VTScada

Software for Monitoring & Control

Developer's Guide

VTScada Developer's Guide

Excerpts from the VTScada Help Files

Contents

Install VTScada and Build an Application

Installation:

You will start by running one of the two installation files: VTScadaSetup-Full.exe if you have purchased the Alarm Notification Option or VTScadaSetup-NoSpeech.exe otherwise. This choice is enforced by the installer, according to what is enabled by your license key.

Options During Installation:

- Choose to download maps as required. Select this unless you have no use for maps, or you intend to run the application on a workstation that will not be connected to the Internet.
- Change the installation folder if required. Some system integrators choose to maintain multiple versions of VTScada by installing each in its own folder.
- Exclude portions of the program including the documentation and the tools for testing and debugging. Choose this only to save space on servers where no development work will be done.
- Select either 32-bit or 64-bit architecture. 64-bit is recommended in most cases unless you are installing on older hardware or operating system.
- Install VTScada as a Windows Service. Do not choose this option until you have created the application that will run on that server and configured it to run automatically. It is also recommended that you first configure the VTScada Internet Server so that you can create client connections to your application using the VTScada Internet Client or Mobile Client.

Build Applications

Most projects will follow a similar process, as described here. The order of some steps can be shuffled and some are optional.

> Note: The first time that you open the Idea Studio, it will display the Overview page, which contains helpful information for new developers. After reading the welcome message, you should do a Select All (Ctrl + A), then Delete, thus clearing the screen for development work to begin.

Preliminary design

Before starting to build in VTScada, you must have a good technical knowledge of the hardware to be monitored, including the how your computer will connect to the devices, a list of the I/O addresses, and some idea of how the user interface should look.

Add new applications

Whether building from scratch or adding a new application based on an existing one, the Add New dialog is the first step.
Add Applications – Options for creating an application
Structure and Rules – The structure of VTScada applications.

Communicate with hardware

You must know the communication protocol and the I/O addresses for the hardware. With that information, you can create tags, the building blocks of every VTScada application.
Hardware – Tags and I/O – About ports, drivers, I/O and polling remote assets.
Working With Tags – The Tag Browser – How to create and modify tags.

Create a display page

Pages are the display screens for operators to use. This step and the next two are generally taken together, one page at a time.
Create a Page (or Widget) – Add new pages to the application.
Define Navigation Between Pages – Links between pages can be created through the menu or with hotboxes.

Draw the user interface

The VTScada Idea Studio is the primary tool for creating design elements on an application page. Many tools are available to help you create a clear and intuitive user interface for operators.

Build the User Interface with the Idea Studio – Starting point for the various design tools.

Formatting Tools – Tools to refine the page appearance.

Draw widgets

Widgets are used to display information being read from your hardware and to provide controls that write to hardware.

Draw Widgets – How to.

Widget Reference – Describes every way that every tag can be drawn.

Design Your Own Widgets – Customize VTScada for your needs.

Configure alarms

Alarms notify operators when values move outside defined tolerances. Some tags include built-in alarms while others require that an Alarm tag be added.

Alarm Configuration – Create alarms that work the way you need.

Customized Alarm Display – Build your own alarm notification system.

Send alarms to remote operators.

The built-in alarm page requires no configuration, but you may want to provide alternate ways to notify operators of alarms.

Spoken Alarms on the Server or Workstation – You can have alarms spoken on the local workstation instead of relying on the default alarm sound.

Alarm Notification System – Alerts by Phone, Email or SMS-Text – If you purchased the Alarm Notification System option, you can send notifications to off-site operators.

Log data

The Historian Tag is used to record a history of tag values and then display that data in a report. The Reports Page and the Historical Data Viewer Page are two tools provided to help you review logged data.

Historian and Logger Configuration – Ensure that you are collecting important information.

Alarm Data Logging – Details about what is logged for alarms.

View logged data

Reports and the Historical Data Viewer are the two most commonly used options for reviewing your logged data.
Historical Data Viewer Configuration
Report Configuration

Secure the application

Operator privileges are granted on a user-by-user basis. This can be managed easily by creating roles (job descriptions) that have the privileges required to do that job, then assigning roles to user accounts. You can extend VTScada's security options by creating your own Application Privileges that apply only to pages or tags within your application.
See: Secure Your Application in the VTScada Admin Guide.

Add email options

Both the Alarm Notification System and the Reports page can use your email server to send information to operators.
See: Email Setup for Alarms and Reports in the VTScada Admin Guide.

Enable the VTScada Internet Server

A browser interface is provided so that operators can work from any workstation on your intranet or, (if connected to the Internet) anywhere in the world.
Internet and Mobile Device Configuration

Distribute the application.

For purposes of load distribution and redundancy, a VTScada application can be run on multiple servers and workstations.
 See: ChangeSets - An Application in One File in the VTScada Admin Guide.
Client/Server Reference

Filter alarms and other information

Control what can be seen, based on which workstation is being used or which operator is logged on.
Filtering Tags, Alarms and Realms

Customize your system

Control the appearance and general functionality

See Application Configuration in the VTScada Admin Guide. – Includes both application properties and system properties.

Create your own tag types for specialized equipment and installations.

Design Your Own Tags and Automated Tag Configuration

Create your own animated widgets.

Design Your Own Widgets

Place your sites on a map.

Sites & Maps

Feature List

For: developers and system integrators who build or update VTScada applications.

The tools and building blocks that are available to help you create applications, grouped by category.

The HMI Interface for Operators

Build the Display	Define Navigation Between Pages
Alarm Configuration	Historical Data Viewer Configuration
Secure Your Application (Admin Guide)	About Operator Notes (Operator's Guide)
Sites & Maps	Report Configuration
About Page Notes (Operator's Guide)	

Secure Universal Access

Connect Using the VTSX Program (Operator's Guide)	Connect Using a Mobile Device (Operator's Guide)
Alarm Notification System – Alerts by Phone, Email or SMS–Text	

Intuitive SCADA Software Development

Build the User Interface with the Idea Studio	VTScada Graphics Library
ChangeSets – An Application in One File (Admin Guide)	Version Control
Working With Tags – The Tag Browser	Automated Tag Configuration
Scripting and Automation (Programmer's Guide)	Debugging and Analysis (Programmer's Guide)
Designate Primary and Backup Servers	Backups (Admin Guide)
Monitor... (Admin Guide)	Historian and Logger Configuration
Client / Server Configuration	Filtering Tags, Alarms and Realms

Station and Site Tags	Expressions in Tags and in Widget Properties (Programmer's Guide)
VTScada as a Windows® Service (Admin Guide)	

Advanced Connectivity

SQL Queries of VTScada Data: The ODBC Server	OPC Configuration in VTScada
Modem Manager Service (Programmer's Guide)	VTScada I/O Device Driver Library
Web Services and XML (Programmer's Guide)	Polling Driver Tags

Add Applications

Create applications and add them to the VAM using the Add Application button – the plus sign at the lower left corner of the VAM.
You will use the same button whether you are creating a new application or loading an existing one.
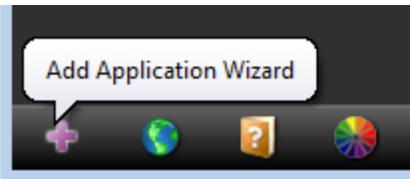
Note: The Add New dialog will open automatically as part of the VTScada installation process.

Note: If you want the Idea Studio to open automatically when a user with configuration privileges logs on, set the application property AutoOpenIdeaStudio to true (1).

Note: You can remove an application from the VAM without also deleting it from the computer. Click the application name to expand its bar, then click its trash can button. You will be prompted for whether you want to delete or retain the files.

Use the following steps to begin the process of adding an application. The wizard will provide a series of choices, ending with a Finish button. At any time before clicking Finish, you can return to previous screens and chose different options. The path through the wizard varies depending on the choices you make. Each path is described in one of the following instruction sets.

1. Open the VAM window.
2. Click the Add New button.
    The Add Application wizard opens.
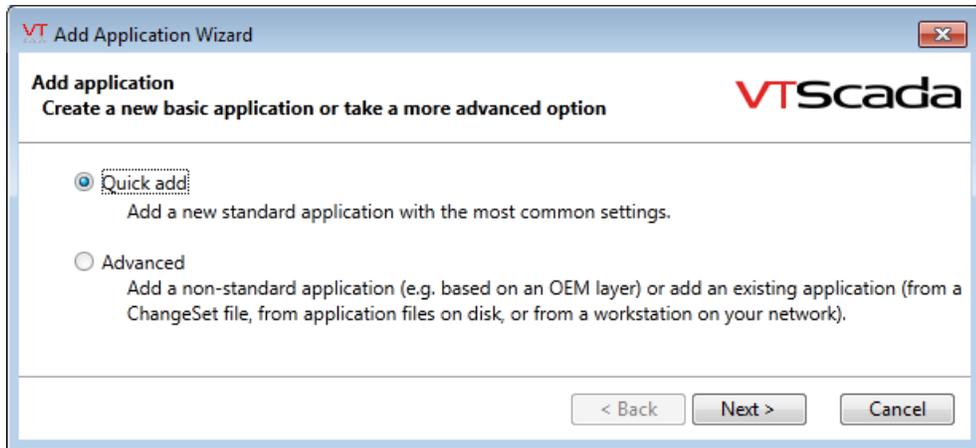
3. Chose between *Quick add* and *Advanced*

- Quick add will create a new standard application (or a ScadaAce application if you have purchased that option,) pausing only to ask for a name to be given to the new application.

- The Advanced option will provide a range of choices for creating the new application.

4. Click Next to open the second screen of the wizard.

- If you chose *Quick add*, the second screen prompts for a name. If you prefer to call the application something other than "New Application", type the name in the data entry box and click Next, then click Finish on the following screen. You will have the option of starting the application immediately. Note that the editor window will open automatically in new applications, allowing you to start development work immediately.

- If you chose *Advanced*, you will have the following choice of paths to follow:
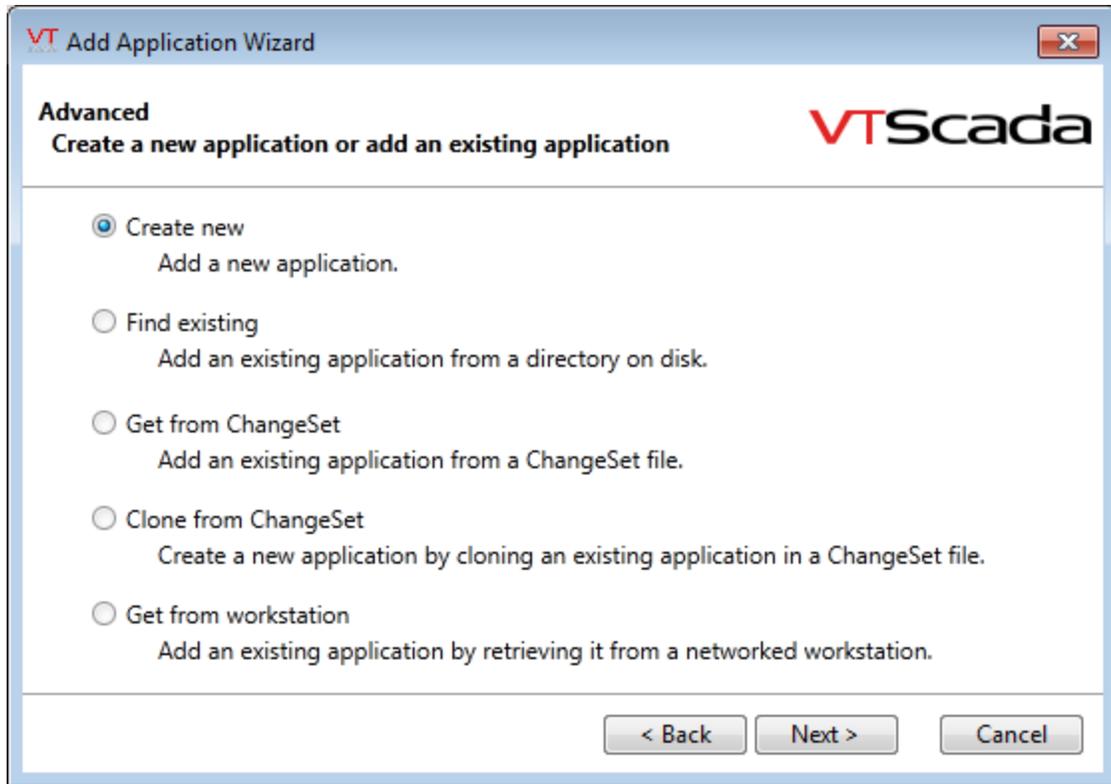
Create New

(Advanced option #1 of the Add Application dialog)

After opening the *Add Application Wizard*, choosing *Advanced*, then *Create new*, you will arrive at the *Create new* page.
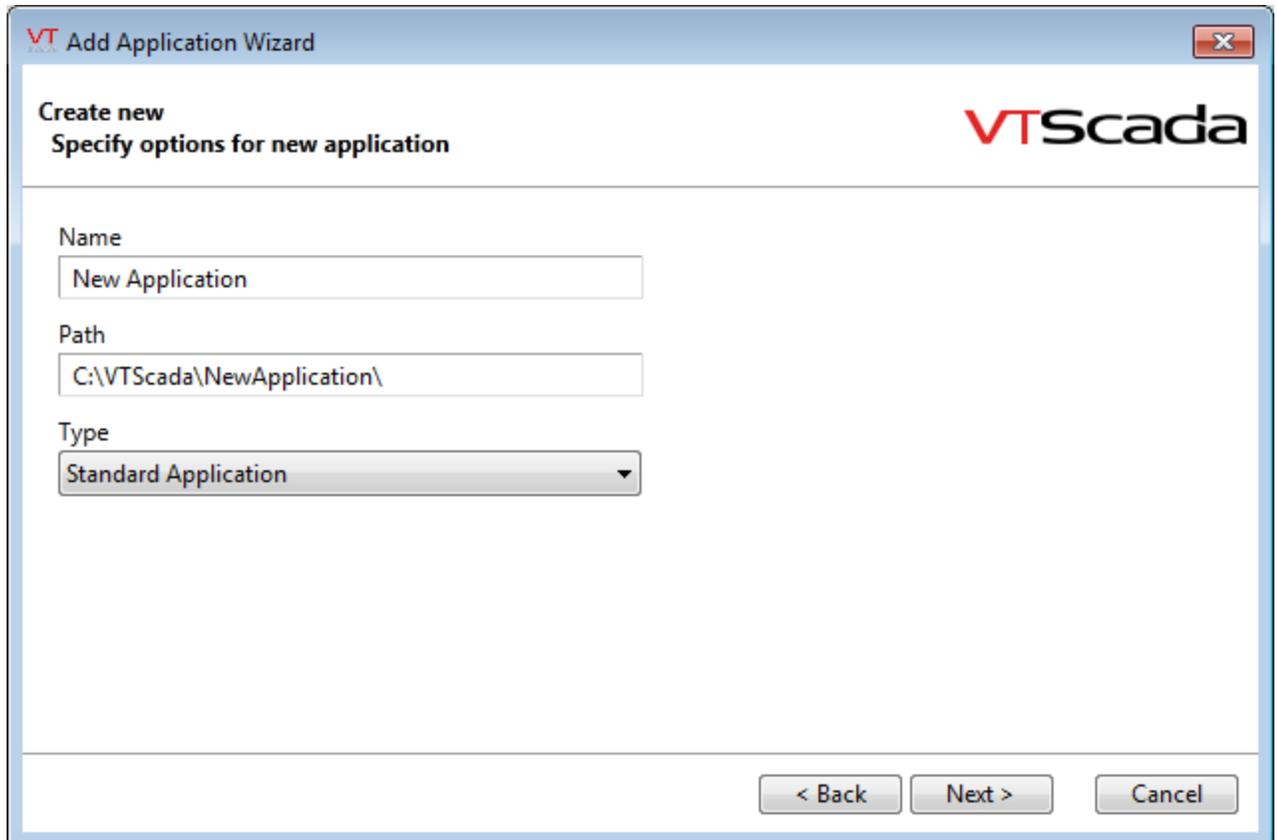
- A default name, "New Application" is provided, but you are advised to replace this with a more relevant name.
- The path will include the VTScada installation folder and the name of the application, with spaces and other symbols removed. You may choose or create a different path, but it is seldom useful to do so.
- The *Type* selects the OEM layer that the new application will be built upon. OEM layers are most often used to provide industry-specific features. See Structure and Rules in the Related Information section for details.

The *Next* button will take you to a confirmation screen where you can review your choices before clicking Finish to create the application.

Find Existing

(Advanced option #2 of the Add Application dialog)

After opening the *Add Application Wizard*, choosing *Advanced*, then *Find existing*, you will arrive at the *Find existing* page.
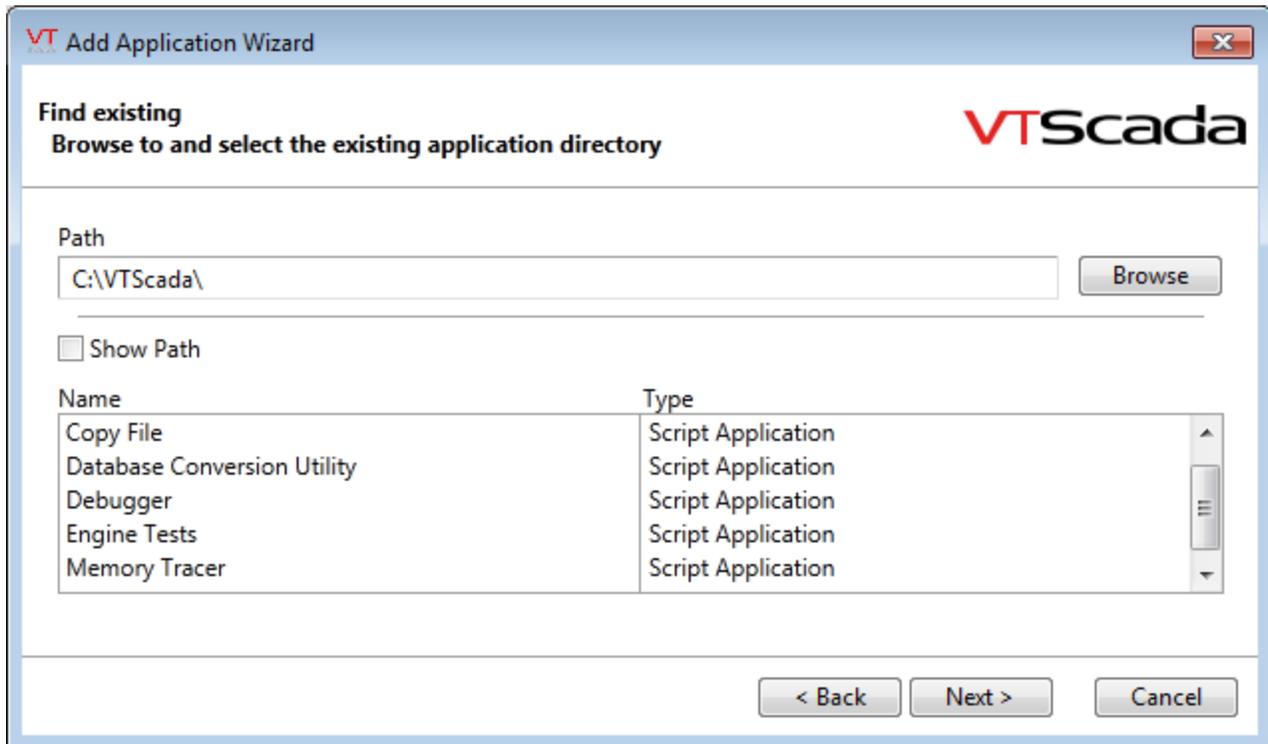
Use this if you have an application on your workstation that you wish to add to the list in the VTScada Application Manager (VAM).

For example, a developer or customer may have given you an application on CD or memory stick. After copying the folders to your computer, you will follow these steps in order to see and run the application in VTScada. You must also follow these steps in order to view and run utility programs that come with VTScada. (These utilities exist to help VTScada programmers optimize their code. Only a few are useful for ordinary application development work.)

1. Browse to the path containing the application's folder.
   The Show Path option is useful for cases where the path name does not match the application name.

2. Select the application to load.

3. Click Next.
   A confirmation page opens, where you can review your choices before clicking Finish.

Troubleshooting:

- The folder exists, and contains an application, but the Found Applications dialog will not display it.
  The application is already listed in the VAM, and cannot be added a second time. If you are attempting to make a copy, create a ChangeSet of the existing application and then create a cloned application using that.

Get From ChangeSet

(Advanced option #3 of the Add Application dialog)

Note: The preferred way to copy a stand-alone application from one computer to another is to package it into a **ChangeSet**[1] file. Applications that will run on multiple servers are best distributed using the Get From Workstation option.

After opening the *Add Application Wizard*, choosing *Advanced*, then *Get from ChangeSet*, you will arrive at the *Get from ChangeSet* page.

---

[1]An application's development history, packaged into a single file for distribution. There are several types of ChangeSet file.

1. Use the Browse button to locate the ChangeSet file.

2. Optionally, change the path (folder), into which the application will be placed.

3. Click Next.
   A confirmation page will open, where you may review your choices before clicking Finish.

The *Name* and *Type* cannot be changed in this screen. You may change the name later using the Application Configuration dialog. The type is the OEM layer that this application depends upon, and cannot be changed at any time.

Note: The OEM layer (*Type*) must exist and be added to the VAM before this application can run.

Troubleshooting:

- A "1" or other number is added after the name or path.
  The application already exists on your computer. You may choose to create a copy (clone) if you wish. The "1" distinguishes the clone from the original.

- If the "Type" field refers to an application that does not exist on this computer, you must add that application before this one will run. Ensure that whoever gave you the ChangeSet file also provides a ChangeSet of the underlying type. See: Reusable Object Layers

- If the application in the ChangeSet file had a server list configured, and those servers cannot be accessed from your workstation, you must remove the server list before starting the application.

- If security was enabled in the application, ensure that you have a valid account name and password before running the application.

Clone From ChangeSet

(Advanced option #4 of the Add Application dialog)
A cloned application, created using a ChangeSet, provides a safe way to experiment with a new development idea. The clone is an exact[1] copy of the original at the time that the ChangeSet file was made, and will include all of the existing communication drivers and I/O, but not logged data. The revision history (repository) will be included from a standard ChangeSet, but not from a snap shot ChangeSet.
By working in a cloned application, you can test, debug and experiment, all without risk of changing the original application.
After opening the *Add Application Wizard*, choosing *Advanced*, then *Clone from ChangeSet*, you will arrive at the *Clone from ChangeSet* page.

1. Use the Browse button to locate the ChangeSet file.

2. Optionally, change the name and the path (folder), into which the application will be placed.

3. Click Next.
   A confirmation page will open, where you may review your choices before clicking Finish.

The type is the OEM layer that this application depends upon, and cannot be changed at any time.

Note: The OEM layer (*Type*) must exist and be added to the VAM before this application can run.

Troubleshooting:

- If the "Type" field referred to an application that does not exist on this computer, you must add that application before this one will run. Ensure that whoever gave you the ChangeSet file also provides a ChangeSet of the underlying type. See: Reusable Object Layers

- If the application in the ChangeSet file had a server list configured, and those servers cannot be accessed from your workstation, you must remove the server list before starting the application.
- If security was enabled in the application, ensure that you have been given a valid account name and password from the original application before starting this instance.

Get From Workstation

(Advanced option #5 of the Add Application dialog)
If you have VTScada licenses for two or more computers, and at least one license includes server capabilities, then use this option to install the application on the other computers on your network.
The Get From Server option is preferred for distributing an application between computers that are connected on a network. By using this method, you ensure that the most up-to-date version of the application is installed, and that future updates are distributed automatically. (Assuming a properly-configured server list.)
If the application is based on an OEM layer, and that OEM layer is present on the remote workstation, it will also be retrieved at the same time as the application.

Note: If the application is to run on both workstations (Client/Server or networked application) then ensure that a server list has been created before adding the application to the second workstation..

After opening the *Add Application Wizard*, choosing *Advanced*, then *Get from Workstation*, you will arrive at the *Get from Workstation* page.

1. Enter the machine name or IP address of the workstation containing the application.
   On a network with a properly configured domain name server, all local machines that have VTS installed will be available for selection in the drop-down list.

2. Click Next.
   Applications available on the selected workstation will be shown. An asterisk will mark any that already exist on your workstation.

3. Select the application you want.

4. Click Next.

   A confirmation page will open, where you may review your choices before clicking Finish.

After clicking Finish, the application will be copied to your workstation and added to your list of available applications in the VAM.

Troubleshooting:

- No servers are shown in the Choose Server dialog.

  This may indicate that your network domain name server (DNS) is not properly configured.

  In most cases, you can ignore this problem and type in the name of the computer where the VTS configuration server is running.

  It may be necessary to add the Setup.INI property, RPCConnectStrategy, and set it to a valid value.

- The Find Servers dialog reports that the requested computer is unavailable.

  This may be preceded by a "Contacting Workstation…" message.

  The specified workstation is not available, or VTS is not running on that computer.

- The application you expected to find is not listed.

  Applications that exist on both computers are not shown. If it is not listed in your VAM, try the Find Existing option.

Next Steps:

...Hardware – Tags and I/O – Begin configuration by defining the I/O tags.

Related Information:

...See: "ChangeSets – An Application in One File" in the VTScada Admin Guide

...Client / Server Configuration – Instructions for and information about configuring a server list.

...Structure and Rules – Build it correctly the first time.

...Reusable Object Layers – OEM layers

Related Tasks:

...Starting, Stopping, and Visibility – Automation and security.

Remove an Application

At any time, you can choose to remove an application from the VAM's list of those available to run. The choices you make while doing so will determine whether this removes only the application's name from the list or if it deletes the application entirely from your computer.
To remove an application:

1. In the VAM, click once the application to remove.
2. Click on the Remove button (trash can image).
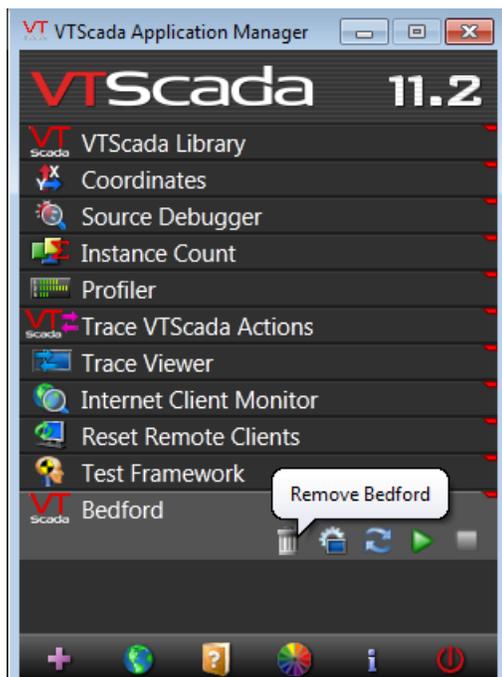
A prompt will ask you to confirm that you want to remove the application.
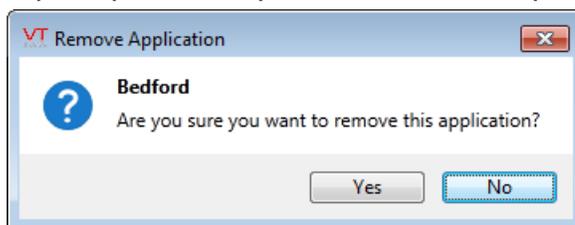
3. Click "Yes"
    You will be asked whether to keep the application's directory and files.
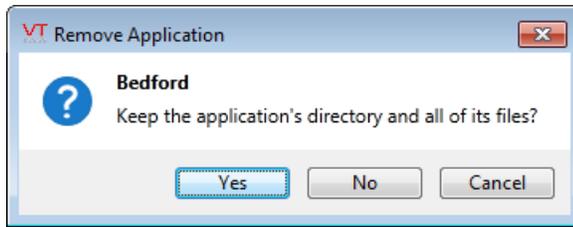
4. Click Yes to keep the directories and files or No to delete them.
   The application will be removed from the VAM.

Note: There is no further confirmation dialog after this – choosing No will remove the application and all of its files from your computer.

OEM layers that have dependent applications may not be removed, but they can be hidden. The process is the same except for the dialog that will be shown.

Structure and Rules

All VTScada applications are built on other applications. At the lowest level is the VTScada library, which provides the development environment, tag and widget definitions, and all the user-interface elements that make up an application.

Script vs. Standard Applications

There are two fundamental types of VTScada application: Script and Standard.

Script applications are written in a text editor using the VTScada scripting language. Script applications are not based upon the VTScada Library Layer and therefore do not have access to the Display Manager, Idea Studio, built-in pages such as Reports and the HDV, the Alarm Notification System or any of the other tools and features provided in the VTScada Layer.

Look in the VTScada Application Manager for examples of script applications. Several useful script application utilities are included with every copy of the program. These include debugging and monitoring tools

such as the Source Debugger, the VTScada Internet Monitor, etc. Script applications are seldom used for SCADA purposes.

Standard applications are based on the VTScada Library layer. These inherit the features and functionality built into the VTScada layer, including:

- The display manager
- Page navigation including menu, page change buttons & page history buttons.
- Tag browser and tag templates (Analog Status, Modem, etc).
- Alarm management.
- Menu item tags
- VTScada Idea Studio (configure pages and all the widgets and objects within your pages)
- Shapes, widgets and images.
- Alignment and spacing tools (align objects left, right, top or bottom. Space objects horizontally or vertically)
- Default Application Pages including…
  - Alarm page (view different alarm categories, manage alarms and alarm sounds)
  - Historical Data Viewer page (plot tag data on a graph)
  - Reports page (generate a variety of report types in varying output formats from a selection of tag data)
  - Operator Notes page (create and view time-stamped notes)

Local vs. Remote Applications

A further distinction between VTScada application types is whether they are "local" or "remote". If local, the application runs on only a single workstation. If remote, the application is designed to run simultaneously on multiple servers and workstations, sharing data and tasks between them and automatically continuing without interruption should a server fail. The application itself is the same whether local or remote – the difference is in the presence or absence of a server list.

Related Information:

...Reusable Object Layers – How new applications gain the features of the applications they are built on.

...Naming Rules – Grouped by the type of thing being named.

...Application Files and Folders – Reference for the parts that make up an application.

Naming Rules

The rules for what is a valid or invalid name will vary according to the type of thing you are naming.

Naming Rules for Applications

In general, applications are stored in a path that matches the name. The rules for application path names are more strict than for application names since the path must be a valid Windows folder name. Application names may include alphanumeric characters, and can include the punctuation key symbols (i.e. decimal (.), and comma (,)), and symbols created using the number keys (for example, the number hatch (#), and underscore characters (_)).

If the application path name is based on the application's name, VTScada automatically removes any invalid characters and spaces. If you attempt to name an application "." (period), then VTS will append a "1" to the path name since "." already exists as the name of the current windows folder. The characters in the following table may not be used in application path names. If you choose to enter a custom application path, you cannot use these invalid characters or spaces.

Invalid Character	Example
Colon	:
Less Than	<
Greater Than	>
Pipe	|
Backslash (other than as a divider between folder names)	\

Naming Rules for Pages

Valid characters that may be used in page file names consist of any combination of:

- Alphanumeric characters (i.e. A through Z, and 0 through 9)
- The underscore character (_)
- The decimal character (.)

Page names can include other symbols (e.g. dollar sign ($), number hatch (#), equals (=) characters), spaces, or strictly numeric characters, but not in page file names. Because these symbols are acceptable in page titles, and the page file name is automatically based on the page title, VTScada strips any invalid characters when generating the page file name.

Naming Rules for Tags

- Tag Names Must Be No Longer than 64 Characters in Length
  This rule refers only to the tag's immediate name, not its full name, which may include a lengthy list of parent tags.
- Full Tag Names Must Be No Longer than 253 Characters in Length
  This rule refers to the tag's full name, which may include many parent tags.
- Tag Names Must Not Duplicate Other Named Items.
  You cannot name a tag "TagBrowser" or "Test", since those names are used elsewhere in VTScada.
- A tag may not be named "Value" or "Area" since those are the names of tag properties. Tags, pages, and widgets may not share a name. Tag names may not duplicate Type names – you cannot name a tag "Calculation" or "Analog Status".
  - In general, it is safest to avoid using common English words unless you add a prefix or suffix.
- Valid tag name characters include:
  - Any combination of alphanumeric characters (i.e. A through Z, and 0 through 9)
  - Spaces, except at the beginning or end of the name.
  - Period (.)
  - Symbols other than those listed below (e.g. you may use the number hatch (#), and underscore (_) characters in tag names)

- Invalid tag name characters consist of:
  - Backslash (\ used to separate parent names)
  - Forward slash (/)
  - Colon (:)
  - Asterisk (*)
  - Question mark (?)
  - Double quotes (")
  - Less than (<)
  - Greater than (>)
  - Vertical bar (|)
  - Leading spaces
  - Trailing spaces
- Tag names cannot be purely numeric.
  "57" is not a valid tag name but "Station57" is.
  An interesting side effect of this rule is that you cannot name a tag "E1" or similar, since this is a valid numeric expression: ("E1" == "0E1" == "zero exponent one").

In the event that you have entered an invalid tag name, VTScada will notify you with a warning dialog, and will clear the tag properties folder's Name field for the entry of a valid tag name.

Application Files and Folders

Files within an application are divided between "user files" and "working files". This chapter discusses user files. The set of working files in the .sync folder must not be edited. Any attempt to do so will damage or destroy your application.

Most user files may be edited using a text editor, but be aware that edited files must be imported into an application's working set before those changes will go into effect. The Import/Export Files dialog is not available in runtime-only versions of VTScada. Also, access to the Import/Export Files dialog is controlled by the Security Manager. Only users with the Edit Files privilege may import an edited file.

As you work within VTScada, it will automatically update the user files to reflect changes to the working set.

Note: The working set of files and the change history repository are stored within the .sync folder. Do not attempt to edit any file in that folder structure. Any attempt to do so will damage your application.

Folders:

Pages

Contains the source code for all of the application pages. You may edit these files and import your changes to modify an application.

PlatformInfo

Contains information about your VTScada installation, the application and the computer you are running the application on. These details may be viewed by opening the VTScada Application Manager (VAM), clicking on Properties, then on Information.
 Any user changes to the files in the PlatformInfo folder will be ignored by VTScada.

Resources

Contains logo and button images used by VTScada.

Bitmaps

If you are adding your own custom images for display in an application page, you will create this folder and store the images here.

Tags

If the setting, ExportTagFilesToUser is set, this folder will contain an up-to-date copy of all the tag files. This copy cannot

be imported, but serves as a reference and can be used in recovery should the .sync folder be damaged.

Data

Contains data output files generated by the application including log files, retained data files and network values. Copy this folder to make backups of your application's data.

Widgets

Contains user copies of the source code for all user-created widgets.

Files:

Accounts.Dynamic

Created only after security has been enabled. Contains encrypted account information. User edits to this file will be ignored.

Application.version

Stores information about the current revision of the application. User edits to this file will be ignored.

AppRoot.SRC

The main source file for an application. Lists top-level variables and modules in use by the application. Edits may be imported. This file was named "AppMod.SRC" in earlier versions of VTScada.

Default.ROS

Contains the default call-out roster for the VTScada Alarm Notification System.

GroupName.SRC

The source code for any user-defined widgets (groups) in the application will be stored in these files, where the name of the widget is used instead of "GroupName". User edits to these files may be imported.

Lexicon.VLX

If you customize the speech lexicon within an application, your edits will be stored in Lexicon.VLX.

PageMenu.TXT

Stores the application's page menu. This file replaces Menu.TXT and Menu2.TXT from earlier versions of VTScada. Edits may be imported.

Servers.RPC

Contains the list of configuration servers. This text file is in XML format. User edits may be imported, but must be made using correctly formed XML.

Settings.Dynamic

Contains application properties (formerly, configuration variables) whose value may be changed without re-starting the application. This file follows the same structure as the obsolete Config.INI

Settings.Startup

Contains application properties (formerly, configuration variables) whose value can only be changed by re-starting the application. This file follows the same structure as the obsolete Config.INI

Related Information:
See: "File Management Tools" in the VTScada Admin Guide

Reusable Object Layers

While VTScada's development tools are useful for a wide variety of industries, you may also want to create additional reusable objects. These objects might be device drivers, reports, tag types, widgets, or other software modules.

With the VTScada layering system, you can create these tools in one application then re-use them in many others. This feature is especially useful for system integrators who develop applications for several clients in a particular industry.

How it works:

Your application, like every application, is built on top of the VTScada System Layer application. All the pages you see in a new application, all of the Alarm Priority tags, Font tags, MenuItem tags, all the built in reports, and more, are defined in the VTScada layer and inherited by your application. You can change some of those tools, such as choosing a different font for one of the Font tags. Other things, such as the Operator Notes page, you use, but you cannot change.
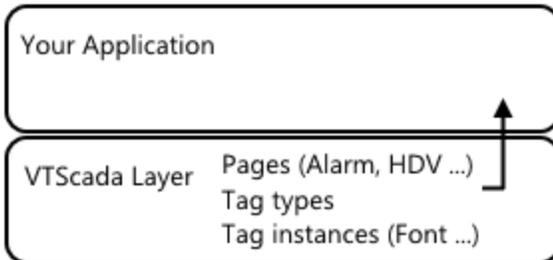
In your application, you build a new tag type called The Gizmo for equipment unique to your industry. You also create a widget that represents all the I/O and alarms for The Gizmo. Further, you have created a report so that operators can view a daily summary of how The Gizmo performed yesterday.
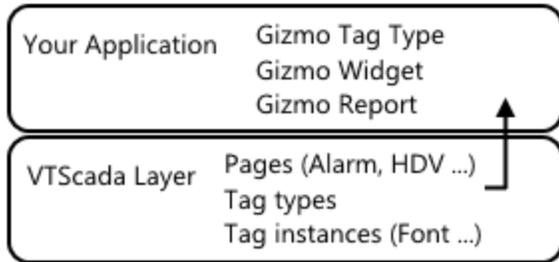
You are using tools inherited from the VTScada layer to build new features in your application.

Two plants use The Gizmo brand hardware for their operations. Your application will be distributed to both plants to serve as the OEM layer for their applications. The applications at Plant A and Plant B also inherit all the features of the underlying VTScada layer, so they also have all the standard tags, widgets, reports, etc.
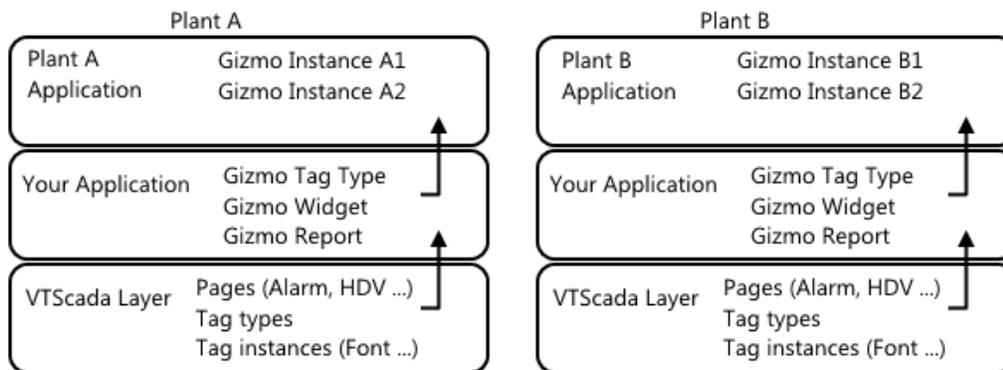
If the Gizmo manufacturing company adds a new feature, you will modify your application to incorporate that into your tag type, widget and report. You then send a Snapshot ChangeSet to both Plant A and Plant B, updating the OEM layer at each location. They can now install and use the latest Gizmo, without needing to do a thing to their applications. The changes were incorporated when they applied the ChangeSet to the OEM layer.

Any VTScada application can be used as the base layer ("type" or "OEM layer") for another. When creating every new VTScada application, a required step is to select its type. (The "Quick Add" option in the wizard simply sets the type to Standard Application.) If, in the Add Application

wizard, you select the Advanced option, and then New, you are given an opportunity to choose the layer that the new application will be built upon and inherit features from. It is possible to create a chain of many such layers, each inheriting the features of all the layers below it.

Note: There is an exception to the rule: Script applications can not be used as OEM layers.
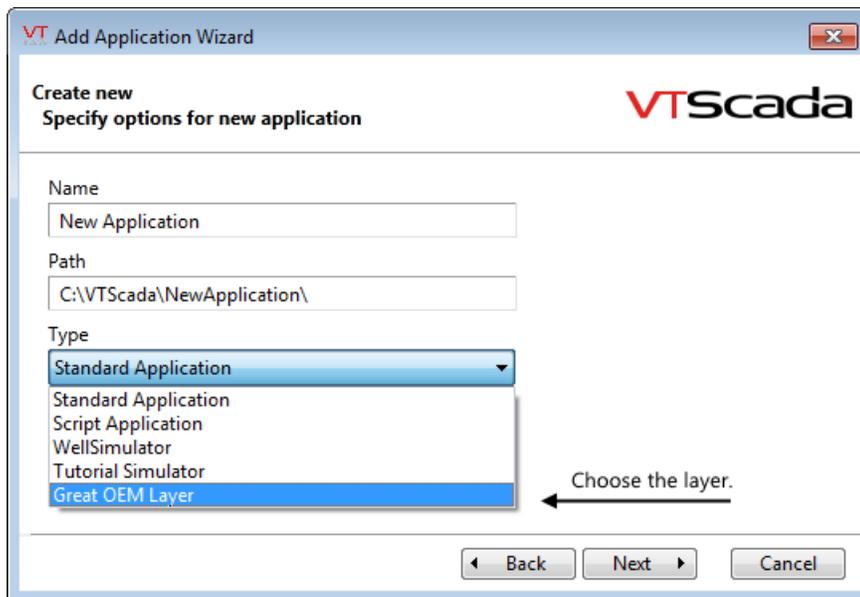
The primary benefits that you gain from the VTScada layering system are re-usability and extensibility:

Re-usability: Inheritance from lower application layers to upper layers means that the custom tools that you create for an OEM layer can be used in other applications. You do not need to recreate those tools or features for every application.

Extensibility: New functions can be added to existing objects such as tags and drivers without changing the structure of applications built on the OEM layer. For example, if you extend the feature set of a custom tag in an OEM layer, then applications built on that OEM layer will immediately have access to the new feature set in your custom tag. Note that certain changes may require a restart before taking effect.

Other benefits from the use of OEM layers include:

- Provides a clear distinction of who is responsible to maintain what portion of an application.

- Ensures consistency across applications.
- Separates the fundamental tools (tags, widgets and other code modules) from the application using those tools. Where application-specific customizations are required, it may be possible to apply a local override to the OEM definition, depending on what you are changing and whether source code is provide with the OEM layer.

OEM layers should be used by:
- A system integrator who deploys SCADA/HMI applications with similar general functionality such as water wells or gas plants.
- A organization that uses a standard set of SCADA/HMI functionality across two or more facilities
- An original equipment manufacturer that deploys a common user interface with their equipment, where the user interface needs minimal or no unique configuration for each equipment installation.
- An organization that needs a clear line between the parts of an application that can be edited and the parts that are under more strict change-control.

Best Practices:
- Whenever the application design calls for two or more of the same complex tag type, widget, or other object, consider creating that in the OEM layer.
- Distribute OEM layers as Snapshot ChangeSets that do not include source files, unless you want others to be able to view and edit your source code.
- Consider enabling security in your OEM layer.
- Consider setting the flag, DoNotStart, in your OEM layer, just before creating the ChangeSet for distribution.
- You will inevitably create instances of tags, widgets and pages in the OEM layer when designing and testing new objects. Before distributing the OEM layer, delete everything that you do not want to be part of applications built on that layer. For example, a new VTScada application has a TCP/IP Port tag type, but it does not come with instances of TCP/IP Port tags.
- If distributing a Snapshot ChangeSet without source code, and if you want to leave the default Overview page intact for applications to edit, then delete the OEM layer's copy of Overview. The default VTScada version will take its place

and be available to the dependent applications. After deleting your copy of Overview, you will need to edit the page menu to put the VTScada version back in.

- In an installation with primary and backup servers, ensure that Application Configuration option, "Synchronize the configuration of OEM layers via derived applications" is selected in the top-level application (found in the "Other" tab of the Edit Properties page of the Application Configuration dialog) so that updates to the OEM layer will be distributed using the same server list.

Related Information:

...Inheritance Across Layers

...Propagation of Changes Through Layers

...Distribute OEM Layer Updates

Inheritance Across Layers

New applications inherit object definitions and application properties from the layers that they are based on. They normally do not inherit *instances* of objects. So, for example, if your OEM layer has a custom page, your new application will be able to see those pages in the OEM layer and can add them to the menu, but will not have its own copy of the source code for that page. The OEM page is visible in the application and can be used there, but (generally) the dependent application can not change that page. If the OEM layer were deleted from the hard drive, the dependent application could no longer view the page (and likely, would not even start).

The same rule holds true for image files, device drivers, wizards and any other custom code. All of these objects in the OEM layer are visible from the new layer, but their files are not copied from the OEM layer to the new application. The new application remains dependent upon its OEM layers to supply various objects rather than becoming a complete, self-contained unit.

For an example, look at the list of application properties in a dependent application: properties with values set in this application are shown in

black and can be changed. Properties whose values were set in the OEM layer are visible, and affect the dependent application, but they are not defined locally. Those properties can be changed, but only by creating a local copy, which will override the value from the OEM layer.

Modify the properties of your application

To add an application property, click the "Insert" button. To delete an application property, select the property to delete and click the "Delete" button. To copy a property (for example, to override a setting for a particular workstation), select the property to copy and click the "Copy" button. To modify an application property, select the property and modify the property fields. You can sort by clicking on the column headings.

The changes you make are not applied until you click the "Apply" button.

| * | | | | | | ▽ ☐ Hide OEM Properties |

Property Name	Section	Value	Workstation	Restart	OEM/	Comment
OperatorNotesConverted	Application	1	-- default --			TRUE indicates that the Operator Notes
Page	System	PageMenuPage(Ir	-- default --			Initial page to show when application sta
RepeatMenuTime	System	5	-- default --			The interval between speech utterances
RosterDelay	System	0	-- default --			
SQLQueryHideLegacyTables	System	1	-- default --			If TRUE prevents v11.0- style tag tables f
#DataLines	System	100	-- default --	✓	✓	Number of communications display line:
5PointStarLabel	Labels	5-Point Star	-- default --		✓	
ABCommMessagesLabel	Labels	AB Communicatic	-- default --		✓	
ABDriverDisableCommStats	System	0	-- default --	✓	✓	
ABDriverStatsLabel	Labels	AB Driver Statistic	-- default --		✓	

Use Care: If the OEM layer includes the source code of the page or widget, then upon opening the page or widget in the Idea Studio, a local copy of the source code file will be created immediately. Changes made in the current application will not affect the OEM layer. *Changes made to that object in the OEM layer will no longer affect the dependent layer.*

> Note: Do not include source code files with your OEM layer unless you intend to make those objects editable in applications built on your layer.

Tag instances are treated in a different manner. Source code is not copied to the child application, but it is possible to override configuration properties. If you have sufficient security privileges, and if the CanRedefineOEMTypes property has been set to 1 (after reading the associated warnings) then you can also redefine tag type definitions in the dependent layer.

Propagation of Changes Through Layers – Describes how and when changes in the OEM layer will affect dependent applications.

Propagation of Changes Through Layers

You will need to deploy changes in the OEM layer if the AutoDeploy option has been disabled.

If the OEM layer has been installed at a site other than where development is being done, then updates to that layer should be made using Snapshot ChangeSets. If the dependent application (henceforth, "application") is modified using new features from the OEM layer, then it is essential that at the client site, the OEM layer be updated before the dependent application.

A restart of the application will be required if changes have been made any of the *.Startup file files or to source code files, excluding tags and pages. Any change that requires a restart of the OEM layer, including all properties marked "Restart required", will also require a restart of the application.

Updates from the OEM layer to the application will be blocked by overrides in the application.

Tag instance parameters

Every tag instance in the OEM layer will be visible in the application and can be used there.

No restart required for updates. Changes made to tag parameters in the OEM layer will be reflected immediately in the application excepting those that have been overridden.

Application overrides. Any property except the name can be overridden in the application.

Tag structure (child tags of defined types).

Every user-defined type in the OEM layer can be used to create

new tag instances in the application.

No restart required for updates. Changes to the structure in the OEM layer will be reflected immediately in the application.

Application overrides. Possible, but discouraged. It is better to define a new type using the OEM structure as a component.

To redefine an OEM type you must have the Manage Types security privilege and the application property, CanRedefineOEMTypes, must be set to 1. (Review the warnings related to that property before proceeding.) New child tags can then be added and the Redefine Type command can be run in the application.

Updates to the type that are made in the OEM layer will continue to be seen in the application with one exception: Adding a child tag to the type definition in the application, then adding a child tag with the same name to that type in the OEM layer will cause an error. The child tag must be removed or renamed in the application.

Properties list of custom tag types

No restart required for updates. Changes to the structure in the OEM layer will be reflected immediately in the application.

Application overrides. Not possible under any circumstance.

Menus (page navigation, palettes)

The navigational page menu and the contents of all three palettes are stored as instances of MenuItem tags. The same rule applies as for other tag instances.

Pages and Widgets

Every page and widget in the OEM layer is visible and interactive in the application.

No restart required for updates. (Restart may be required in older versions of VTScada.)

Application overrides. If the OEM layer includes source code, then the act of opening a page in the application's Idea Studio creates a copy. Until that copy is deleted using the Idea Studio in the application, it becomes the version in use. No updates from the OEM layer will affect this copy.

Application properties

Every property in the OEM layer is visible in and applies to the application.

No restart required for updates, unless the property is marked "restart required". Changes to the property in the OEM layer will be reflected immediately in the application.

Application overrides. A copy can be made of any property. The properties list will still show the OEM version of the property in gray, including its current value, but the application's copy of the property will take precedence.

Code modules including reports

Every report and other module defined in the OEM layer can be used in the application.

Restart required for updates. (May vary.) Most module changes in the OEM layer will require a restart, therefore the application will also require a restart.

Application overrides. Possible by writing code. You must copy the module file (or create a new one of the same name) and declare it in the application's AppRoot.SRC file.

Note: If the dependent application contains its own copy of a source code file or an application property, then it will ignore any changes made to those items in the OEM layer.

If the OEM layer is distributed without source code as a Snapshot

ChangeSet, then the dependent application cannot obtain a copy of any of the source code. It can still set its own values for application properties. Changes to the installed OEM layer at the production site can be made only by applying a ChangeSet obtained from the development workstation.

Related Information:

...Inheritance Across Layers

Distribute OEM Layer Updates

If you have modified your OEM layer, then the process to distribute that change depends on how the OEM layer was originally distributed. You may also need to restart the application, depending on the type of change made to the OEM layer.

If your application runs on multiple servers, and is based on an OEM layer, then that OEM layer must also be present on each server. It can be difficult to maintain separate server lists for both the primary application and its OEM layer application, especially if the OEM layer is used by more than one application. Fortunately, one server list can work for both the application and the OEM layer at any site. All that is necessary is to ensure that the SyncOEMLayers property is set TRUE, as it is by default. To verify, open the Application Configuration dialog of your application (the top layer application, not the OEM layer) and look for the check beside "Synchronize the configuration of OEM layers via derived applications" in the OEM Layer section of the Other properties tab.

With SyncOEMLayers set to TRUE (1), updates in the OEM layer's configuration will be distributed automatically using the application's server list as soon as those changes are deployed in the OEM layer.

Note: Certain layers supplied by Trihedral will not synchronize automatically under any circumstance. Instructions for deployment are provided with all updates to these layers.

In rare cases, it is possible that you might not want this OEM layer to synchronize automatically. For example, you want to apply ChangeSets

individually on each server, perhaps to test changes as you roll them out. Another possibility is that the update will require a full restart, and you want to control when that happens on each server. For these cases, you should turn this setting off before applying the ChangeSet to the OEM layer on each server.

Snapshot ChangeSets without source code are run-only applications. This is the most common way that OEM layers are distributed from system integrators to clients. Updates to the OEM layer must made be at a workstation where the OEM layer's source code resides, then distributed to clients via a ChangeSet. If you are also distributing updates to an application and those updates depend on new features in the OEM layer, then it is essential that the OEM layer be distributed first, followed by the dependent application.

Remember that ChangeSet updates can be applied only to applications having the same globally unique identifier.

> Note: Remember that, after an application has been created, its name is only a display property. You can add a version number to the OEM layer's name, and update that number each time you are ready to distribute a new set of features. Some integrators do this to help keep track of which release is installed at a site.
> The name property must be updated manually.

Troubleshooting:

- Changes made to a page or widget in the OEM layer are not showing up in the dependent application.
  The OEM layer was distributed with source code. The page or widget was opened in the Idea Studio of the dependent application, causing a local copy to be made.
  Use the Idea Studio to delete the local copy in the dependent application. If it is necessary to have local variations to a page, then the OEM page should be made into a widget that can be added to a local page.
- The OEM layer has no link to the Idea Studio.

This is by design. The OEM layer was distributed to this workstation (or server in your network) as a Snapshot ChangeSet without code. Pages and widgets within the OEM layer can be edited only on a workstation where the full version, including source code, is kept.

Related Information:

...See: "ChangeSets - An Application in One File" in the VTScada Admin Guide.

Starting, Stopping, and Visibility

If the VTScada application or the VTScada program is stopped (either by an operator or due to a power failure), some operators may be unable to restart the application due to unfamiliarity with VTScada or with computers in general.
In other instances, you or your clients might prefer that the workstations running the VTScada software be dedicated to that purpose alone.
There are some sites where many applications will run simultaneously on one server.
If any of these apply, you may wish to consider some or all of the following options:

- Configure the application to open in full screen mode (that is with no Windows title bar, close button, minimize or maximize buttons).
- Deny the Application Stop system privilege to operators (so that operators cannot close the Display Manager using the Windows close button if the application is not in full screen mode).
- Configure the application so that it can be hidden while running.
- Configure the VTScada program to auto-start.
- Configure the VTScada application to auto-start.
- Configure the VTScada Application Manager to be hidden to all security accounts that do not possess the View Application Manager privilege.

Related Information:

...Hide the VAM

...Hide a Running Application

...Configuring VTScada to Auto-Start

...Automatically Start an Application

...Shutdown on Low Power

...Starting and Stopping Instructions – See the VTScada Operator's Guide

Hide the VAM

There are two ways to hide the VTScada Application Manager (VAM):

- In your application, you can open the Other tab of the Edit Properties page of the Application Properties dialog and select "Hide VAM from users without Application Manager View privilege" .
- You can set the property HideWAM to 1 in the Setup.ini configuration file. When TRUE, the VAM will be hidden from the view. Hiding the VAM prevents operators from having access to VAM's commands and debugging tools.

The first method is recommended since the VAM will be hidden only when your application is running, and this method does not require that you edit configuration files.

> Note: For VTScada programmers: HideWAM is an older name for HideVAM. In Setup.INI, HideWAM may still be used, but if you are creating code that refers to this property, always use the newer, HideVAM. The setting, HideWAM, is checked only on startup.

If the application is set to run full-screen without a Windows™ title bar, then there will be no way for operators to stop it. If there is a title bar with an X in the upper corner, then stopping the application will also cause VTScada to stop. The VAM will not continue to run, hidden from view, when all applications have stopped.

The security privilege, View Application Manager, allows selected users to see and use the VAM while it remains hidden to all others. These users may stop the application without also causing VTScada to exit. (Assuming that they have the Stop Application privilege.)

To configure the VAM to be hidden using the Application Configuration dialog:

Preparation: Ensure that security has been enabled in the application and at least one account has the "View Application Manager" privilege.

1. Open the Application Dialog
2. Select the Edit Properties page.
3. Select the Other tab within the Edit Properties page.
4. Scroll down and select the option, "Hide VAM from users without …".
5. Click Apply to save your changes, entering a comment in the dialog that opens to record your reason for making this change.

To configure the VAM to be hidden using Setup.INI:

Preparation: Before making this configuration change, it is essential that at least one application be configured to start automatically, since without access to the VAM, you would have no way of starting any application. As a safety measure, VTScada will ignore the HideWAM setting unless at least one application has been configured to auto-start. Ensure that security has been enabled within that application, and that the privilege View Application Manager is granted to at least one account.

1. Ensure that at least one application has its Auto-Start option set.
2. Open the Setup.INI file, located directly within the VTScada installation directory, using the text editor of your choice.
3. The property HideWAM should already be present. If not, add it (or better: HideVAM) within the [OEM] section.
4. Set the property equal to 1.

```
[OEM]
HideVAM = 1
```

5. Save and close the Setup.INI configuration file.
6. Re-start VTScada.
    The VAM will automatically be hidden.
7. Grant the system privilege View Application Manager to selected accounts or roles.

Related Information:

...Configuring VTScada to Auto-Start – Ensure that the VTScada program runs automatically when the server restarts.

...Automatically Start an Application – Ensure that your application runs when VTS restarts

...See: "Application Manager View" in the VTScada Admin Guide. Allow Supervisors to see the otherwise hidden VAM.

...See "Hide the VAM from Operators, but not Managers" in the VTScada Programmer's Guide. Create a module to hide the VAM based on conditions of your choice.

Hide a Running Application

An application can be configured to run with no user interface on selected workstations. This might be done on a primary or backup server, where the application will run, but where there is to be little or no need for user-interaction. When there is a need to see the application, any authorized user can click its Show button in the VAM.

Note: For applications that can be hidden, the controls to Hide or Show the application are available only to user accounts that have the Configure privilege.

To allow an application to be hidden:

1. Open the Application Configuration dialog.
2. Select the Edit Properties page.
3. Open the Other tab of the Edit Properties page.
4. Select the option to Enable the Toggle Visibility icon.

VTScada Application Manager
_____

These settings apply to all applications

☐ Hide VAM from users without "Application Manager View" privilege
☐ Don't show splash screen at startup.
☑ Enable the "Toggle Visibility" icon on the VTScada application list
   ☑ When hiding an application log the current user off

The option to log the current user off automatically is selected by default, and is recommended for most situations. If not selected, then the current user will remain logged on until the automatic logoff time is reached. Until then, anyone may click the application's Show button in the VAM and operate the application as the logged on user.

To hide an application:

Note: An application must be configured so that it *can* be hidden before it *may* be hidden.

There are two ways to hide a running application:

1. From the VAM, click the Toggle Visibility icon, as shown:

   If you use this option, you will not be prompted to confirm the action.

2. Click the X at the upper right corner of the main VTScada screen (provided that window has been configured to include a border and the Windows® controls.)

   If you use this option, you will be prompted to confirm the action.
   If you have the Application Stop privilege, the dialog will ask whether you want to hide the application or close it.

Related Information:

...DispMgrHidden – Set only in combination with a workstation name. If EnableShowHide is TRUE, the application will be hidden from view while running on the specified workstation.

...EnableShowHide – When set to TRUE(1), all application entries in the VAM will have a Toggle Visibility icon.

...LogOffUponHide – When LogOffUponHide is TRUE (1 – the default), users will be logged off automatically when the application is hidden.

Configuring VTScada to Auto-Start

Configuring VTScada to start automatically further simplifies its use for operators. When you configure both VTScada and your VTScada application to auto-start, an operator need only turn on a workstation to access the application. This feature can also ensure that workstations are dedicated to running the VTScada software and cannot be used for any other purpose.

This operation can be performed using standard Windows® operating system practices. You can choose to configure VTScada to auto-start for some users, but not for others; or you might wish to configure VTScada to auto-start for all users. Instructions for both procedures follow.

Note: To create a shortcut, as mentioned in both sets of steps, use Windows Explorer to locate the VTSLauncher.exe file, right-click, and select "Create Shortcut".

To configure VTScada to auto-start for all users on a workstation:

1. Create a shortcut to the VTSLauncher.exe program (normally located in C:\VTScada\).
2. Right-click on the shortcut and select Copy from the menu that opens.
3. Press the key combination, Windows key + R.
4. Type, "Shell:common startup".
5. Right-click in the folder that opens and paste the shortcut to this location.

VTScada will start automatically on the next occasion any user logs on to the workstation.

To configure VTScada to auto-start for the current user on a workstation:

1. Create a shortcut to the VTSLauncher.exe program (normally located in C:\VTScada\).
2. Right-click on the shortcut and select Copy from the menu that opens.

3. Left-click the Windows Start button.

4. Click on All Programs, to expand the list.

5. Find the Startup folder.

6. Right-click to past the shortcut into the Startup folder.

VTScada will start automatically the next time that you log on to the workstation.

Automatically Start an Application

Often, the sole purpose of a workstation is to run a VTScada application. It therefore makes sense for VTScada and your application to start automatically after a power failure or other interruption.

If you have configured the VTScada Application Manager (VAM) to be hidden using the System.INI property, HideWAM, it is vital that your application start automatically since you won't be able to see or use any of the controls in the VAM.

1. Open the Application Configuration dialog for the application.

2. Select the Information option from the menu.

3. Click on the Automatically Start option so that it is checked.
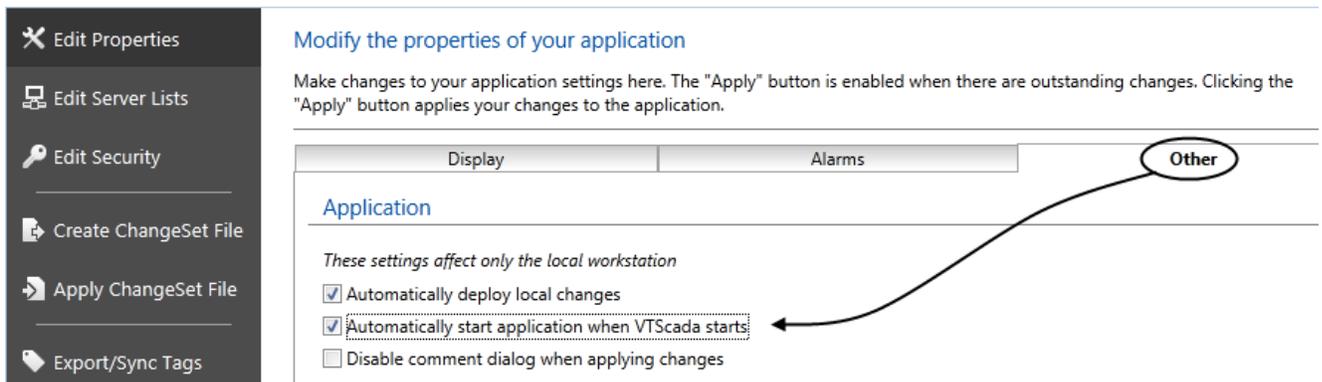
4. Click on Apply.

You might also add the VTScada shortcut to your computer's Startup folder, thus ensuring that VTScada starts automatically when the computer boots. If doing so, ensure that VTScada is run with sufficient privileges so that it is not blocked at startup by your Windows™ operating system.

Related Information:

...See: "Open the Application Configuration Dialog" in the VTScada Admin Guide

Shutdown on Low Power

VTScada is configured to perform an orderly shutdown when it is able to detect that the workstation is running on battery power (usually an uninterruptible power supply – UPS), and that battery is nearing the end of its charge.
You can control this feature using the following properties within the configuration file Setup.INI in the VTScada installation folder. These allow you to enable or disable the automatic shutdown feature, and when enabled, set a threshold for three different battery indicators, any of which will trigger an automatic shutdown.
Warning: Use these settings with care. Some VTScada users have found that in a location with both intermittent main power and faulty status reporting from the UPS, unexpected shutdowns have occurred.

Related Information:
Refer to the VTScada Admin Guide – System Properties for the following:

...OrderlyShutdown – Enables or disables the feature.

...ShutdownOnLowBattery – On or off. When on, and the battery status property indicates "low battery," a shutdown will occur.

...LowBatteryPercent – Threshold value. When the remaining battery life, reported as a percentage value, crosses below this value, a shutdown will occur.

...LowBatteryTime – Threshold value. When the remaining battery life, reported as a time remaining value, crosses below this value, a shutdown will occur.

Hardware – Tags and I/O

You must define how your application will communicate with the hardware that is being monitored. VTScada uses a modular approach, dividing the communication components into three: Ports, Drivers and I/O tags.
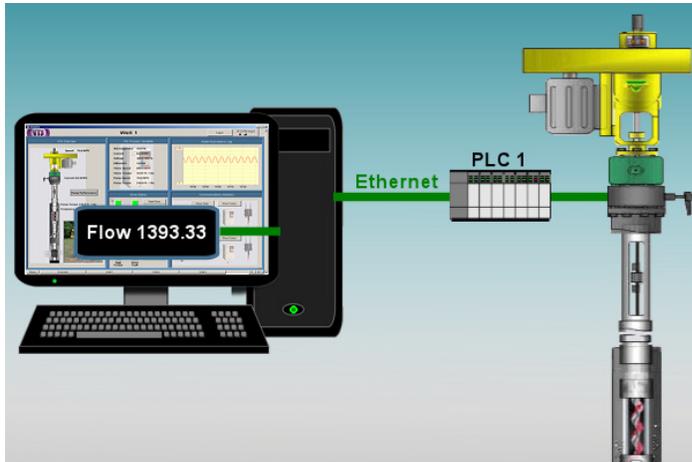
Related Information:
Communication Chain Design Examples

Ports. TCP, UDP or Serial, these define the line of communication between the computer where VTScada is, and the hardware being monitored.

Define a Port
Driver tags need to know where to find the hardware they will communicate with. Rather than build this information into the driver tag, VTScada provides Port tags, allowing you to define the communication details once, then use that configuration for several drivers. If the configuration changes later, then you need only update one tag.
Port tags tell the VTScada drivers about the nature of the physical connection (serial port, TCP/IP over Ethernet, or UDP) and the required configuration details such as IP address, port number, transmission speed, whether a modem is attached, etc.

To create a Port tag:

1. Open the Tag Browser.

2. In the Tag Browser, click "New". You could also right-click on the backslash in the tree view window and select "New" from the context menu that opens.
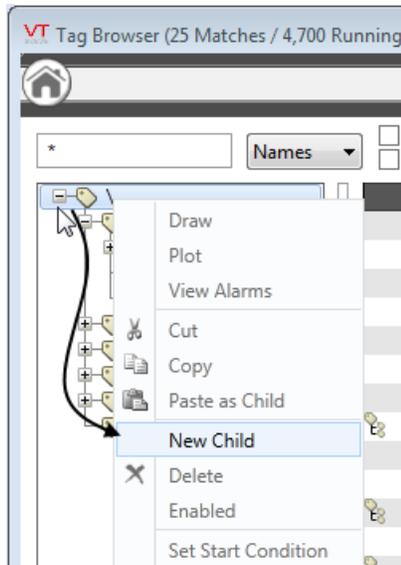
3. In the Select Type dialog, choose the appropriate type of port for your application: Serial, TCP/IP or UDP.
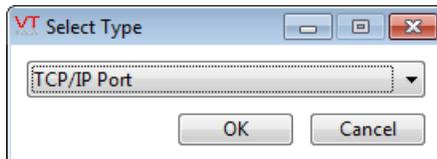
4. In the Tag Properties dialog, provide an appropriate name, area and description for the new tag.

5. Open the Connection tab and define the parameters to match your hardware.

Related Information:

...Port Tags – Port tag technical reference.

...Working With Tags – The Tag Browser – Tag browser description, including filtering options.

...Comm Indicator Widget – Draw the port so that it can be monitored. (A complete list of available widgets is provided with each Port tag description.)

Drivers. Allen Bradley, Modbus, DNP3 and many more. These define the communication protocol to be used. Optionally, a Polling driver might be added to control communication frequency. A driver multiplexer might be added to define an alternate line of communication in the event of device failure. Drivers are connected to Ports.

Define a Driver

Device Driver tags provide the communication protocol used by your hardware. They are designed to link port tags with I/O tags.

Before configuring a driver tag, it is best to create a port tag. The following instructions assume that you have completed that step.

To create a Driver tag:

1. Open the Tag Browser.
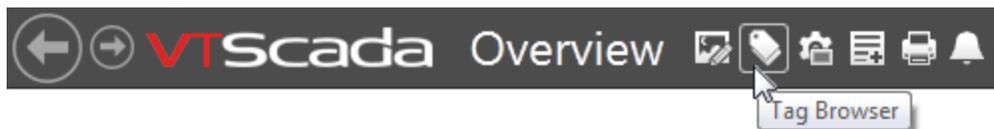
2. [Optional step] Navigate through the tag tree if you want to create the driver as a child of an existing Port tag.

3. Click the button, New.

4. In the Select Type dialog, choose the appropriate type of driver for your application.
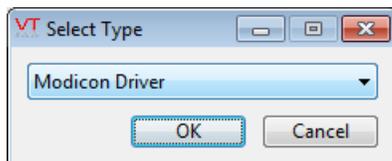
5. In the Tag Properties dialog, provide an appropriate name, area and description for the new tag.

6. In the remaining tabs, provide the configuration details that are appropriate for your device.

By creating the driver as a child of a port, the linkage between the two will be configured automatically. If you chose not to create the driver as

a child of a port tag, you must select a port tag before this driver will be able to communicate with hardware.

Related Information:

...Communication Driver Tags – Communication Driver tag technical reference.

...Communication Driver Log-Enabled Variables – All of the standard drivers have 7 variables that may be logged.

I/O Tags. These are configured to read from or write to specified addresses on the hardware, with scaling applied to translate between the raw machine values and human-readable values. I/O tags are connected to Drivers.

Define an I/O Tag

I/O tags receive values from, or send values to specified addresses on the PLC or RTU. They depend on the configured driver tag to provide the connection details and the protocol for the device.

Addresses must be provided by the person who configured the hardware. This document includes guidelines for the address formats used by each of the drivers provided with VTScada. An assortment of I/O types is provided with VTScada, covering analogs and digitals, inputs and outputs, selector switches, pulse inputs, pump monitoring, and more.

To create an I/O tag:

1. Open the Tag Browser.
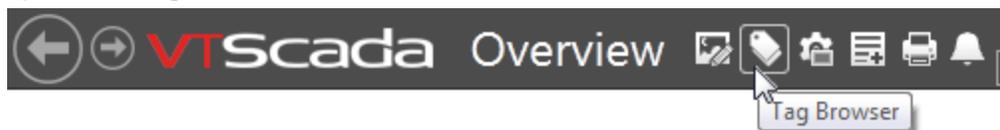
2. Right-click on the driver tag in the tree view window and select "New Child" from the context menu that opens.

3. In the Select Type dialog, choose the appropriate type of I/O for the address you are configuring.
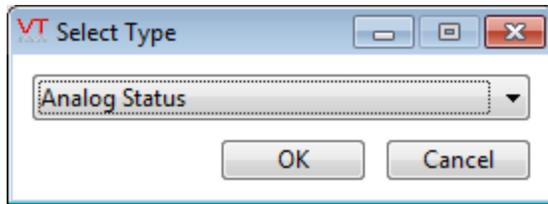
4. In the Tag Properties dialog, provide an appropriate name, area and description for the new tag.

5. In the remaining tabs, provide the configuration details that are appropriate for the type of I/O being configured.

By creating the I/O tag as a child of a driver, that part of the configuration will be filled in for you. If you chose not to create the I/O tag as a child of a driver tag, you must select (or create) a driver tag before this tag will be able to communicate with address specified in the hardware.

Related Information:

...Data Suffixes for Tag I/O Addressing – To define the data type, explicitly .

...Input Tags – Reference of all tags that can read from a device. (Some are also able to write.)

...Output Tags– Reference of all tags that can write to a device.

Next: alarms, data logging and more are connected to I/O tags, and the user interface is built.

Troubleshooting:

- The tag is not visible in the Tag Browser
  Check the filters at the top of the Tag Browser, and the current path shown. The tag is likely excluded from the view.

- No communication
  Work through the tags in the communication chain, verifying that each is configured correctly.
  Ensure that this workstation is able to connect to the device.

Communication Chain Design Examples

A few common arrangements. Many more are possible. Note that the driver tags in each example need not be all the same type.

Scenario 1: VTScada I/O server with wired IP connection to field device transceiver with static I/P.
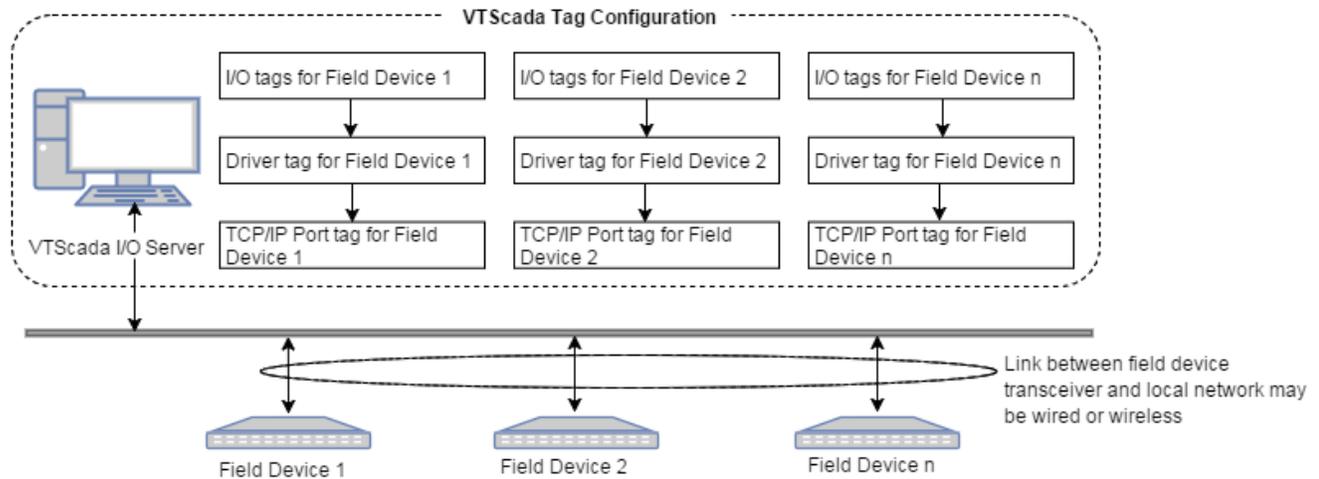
Notes:
1. Scenario shown based on use of typical polling protocol (e.g. Modbus, DF1, ENIP/CIP)

Scenario 2: VTScada I/O server with wireless IP connection to field device transceiver with wireless static IP.
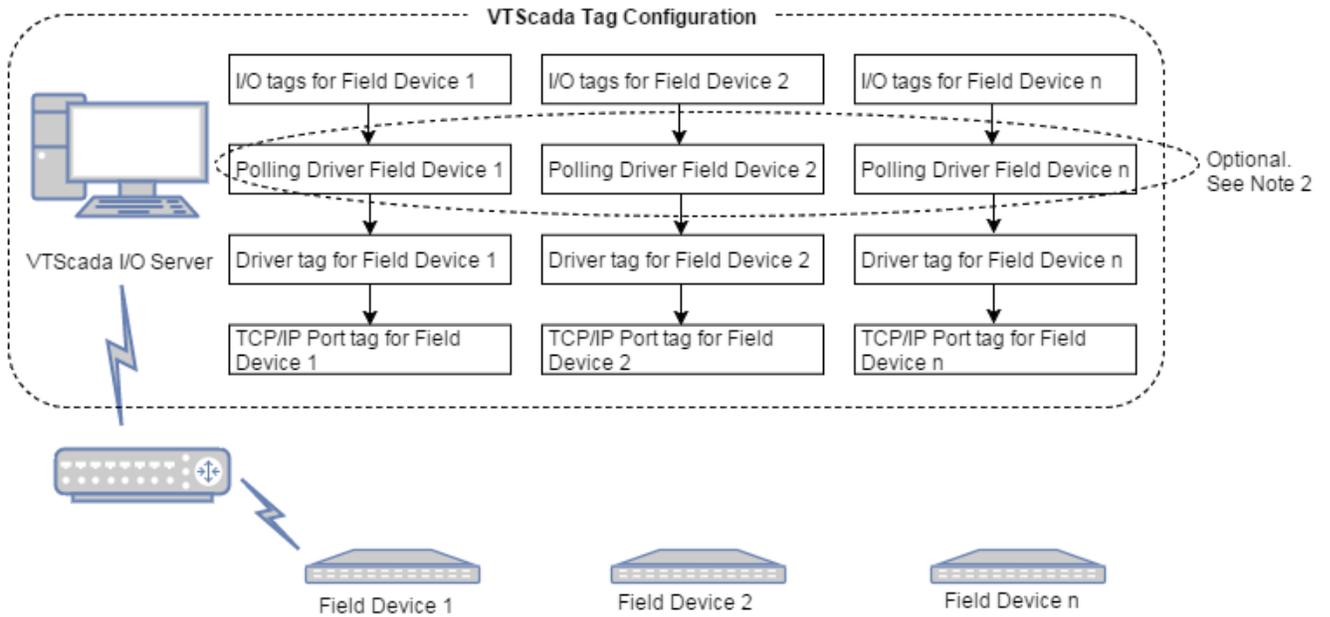
Notes:
1. Scenario shown based on use of typical polling protocol (e.g. Modbus, DF1, ENIP/CIP)
2. Polling group may be excluded except in one of the following scenarios:
   a) Where all data from each field device is to be read at the same time, such as in a single block read. (e.g. To minimize communication costs) In this case, the poll rate in the polling driver overrides the scan rate of the associated I/O tags.
   b) Where the field devices are to be polled in sequence, in which case the polling group ID must be the same for all devices.
   c) Where different protocols (driver tags) are used to communicate with different field devices.

Scenario 3: VTScada I/O server with serial connection to field devices.
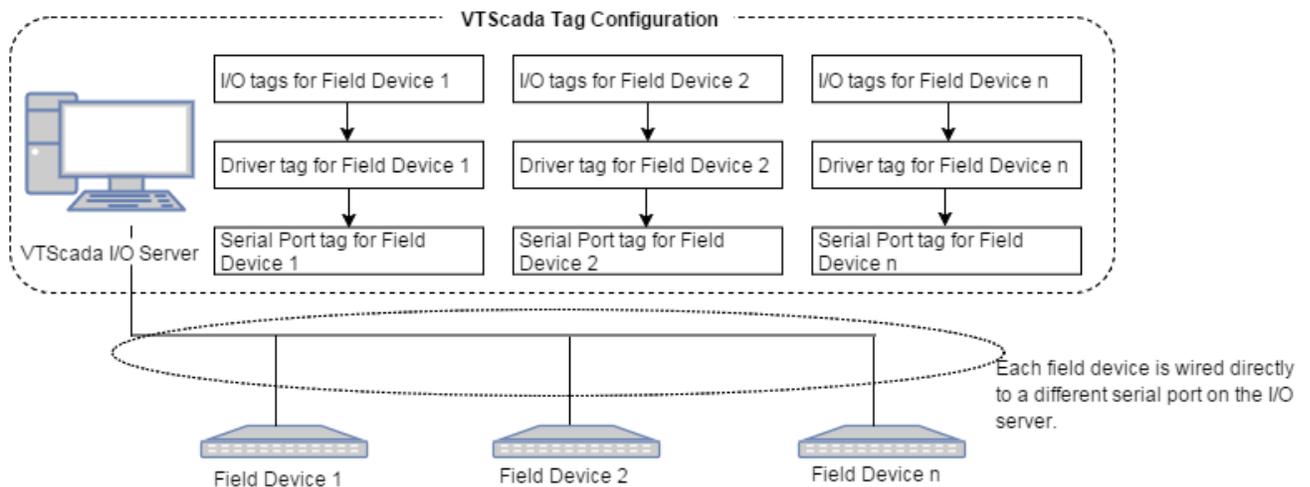
Notes:
1. Scenario shown based on use of typical polling protocol (e.g. Modbus, DF1, ENIP/CIP).
2. If configuring the system for redundant I/O servers, use a serial to Ethernet converter to place each field device on an Ethernet connection.
   If using comport redirector software, continue to use Serial Port tags, otherwise switch the Serial Port tags to TCP/IP tags.

Scenario 4: VTScada I/O server with wireless connection between base station transceiver and field devices.

Notes:
1. Scenario shown based on use of typical polling protocol (e.g. Modbus, DF1, ENIP/CIP)
2. Polling group may be excluded except in one of the following scenarios:
   a) Where all data from each field device is to be read at the same time, such as in a single block read. (e.g. To minimize communication costs) In this case, the poll rate in the polling driver overrides the scan rate of the associated I/O tags.
   b) Where the field devices are to be polled in sequence, in which case the polling group ID must be the same for all devices.
   c) Where different protocols (driver tags) are used to communicate with different field devices.
3. If configuring the system for redundant I/O servers, user a serial to Ethernet converter to place the Base Station on an Ethernet connection. If using comport redirector software, continue to use a Serial Port tag, otherwise switch the Serial Port tag to a TCP/IP Port tag.

Scenario 5: VTScada I/O server with dial-up modem connection to field devices.
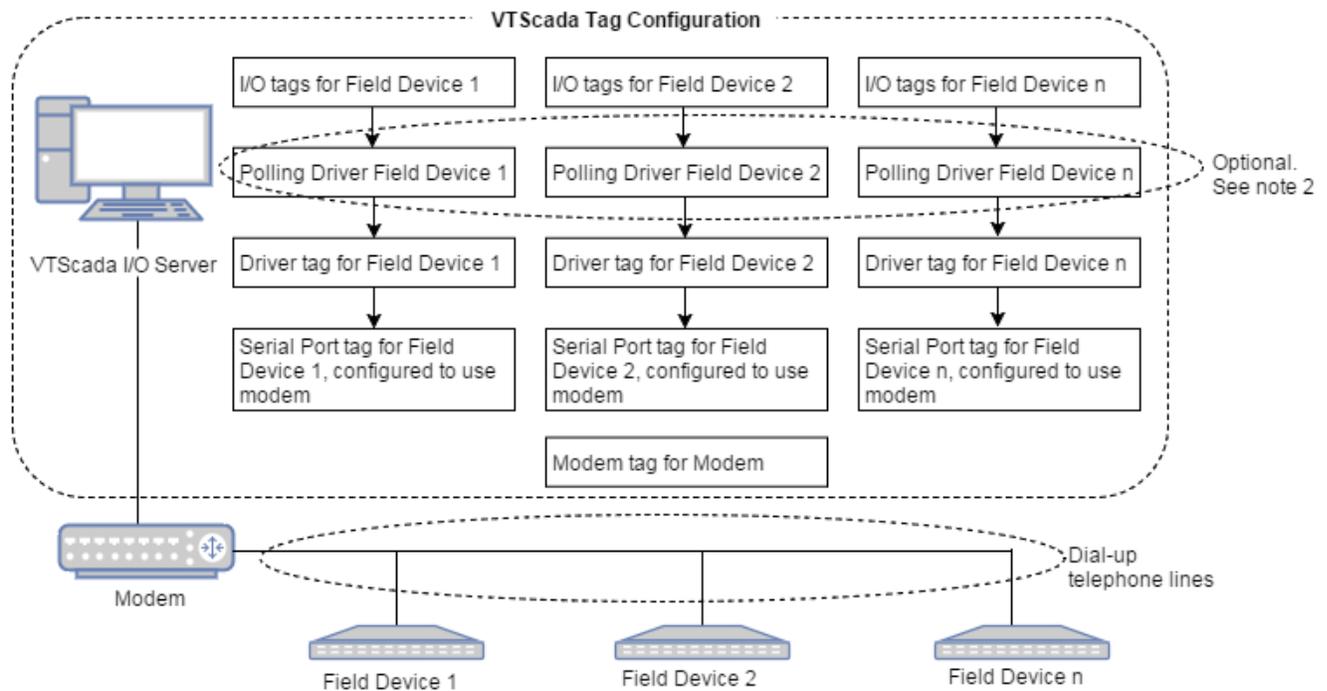
Notes:
1. Scenario shown based on use of typical polling protocol (e.g. Modbus, DF1)
2. Polling group may be excluded except in one of the following scenarios:
    a) Where all data from each field device is to be read at the same time, such as in a single block read. (e.g. To minimize communication costs) In this case, the poll rate in the polling driver overrides the scan rate of the associated I/O tags.
    b) Where the field devices are to be polled in sequence, in which case the polling group ID must be the same for all devices.
    c) Where different protocols (driver tags) are used to communicate with different field devices.
3. If configuring the system for redundant I/O servers, each server must have access to a modem
4. Multi-modem (pooled) configurations supported.

Tag Concepts and Features

Tags are the software version of physical objects and concepts. A tag might represent a TCP/IP port, the device driver for a programmable logic controller, an I/O address on that device, or an alarm setting. All of the characteristics of the device are set using a tag's configuration dialogs and are stored within the instance of the tag.

Most of the tags in an application are used to describe the inputs and outputs of a remote telemetry unit or programmable logic controller. Some tags support those I/O operations by providing alarm set points and history logging. In every application, there will be a few tags that hold

general configuration details such as the fonts to use for labels, the colors to use for displaying alarms, and even the menu structure for navigating pages.

Tags are created and modified using the Tag Browser. They can be drawn on application pages using widgets, to display current values or provide controls for the operator.

Every tag is configured using a tag properties folder– the set of dialog boxes that provide access to each tag instance's properties. Each field in a tag properties folder matches a property in the corresponding type definition. By using a tag's properties folder, you can update its configuration while your application continues to run.

Note that by default, each new tag you create during a development session will have the same configuration as the last that you created of that type, excluding only the name. This can save time when creating a series of similar tags since you won't have to configure the same values over and over. But, if the majority of the fields in each new tag differ from the last, this feature may not seem so convenient. You can disable it by setting the application property, RememberNewTagParameters to 0.

You can also change tag properties by exporting the tag database for external editing, and then import the modified database back into VTScada.

Title Bar: Identifies the tag. If the tag is new, the title bar shows the type of tag being created. If the tag exists, the title bar displays the name.

Tabs: Organize the properties for the tag into logical groups.

X Button: Clears a tag reference field (i.e. a field that takes its data from some other tag.)

VT New Analog Input Properties

| ID | **I/O** | Scaling | Owner | Merit | Display | Alarm | Script | Logger |

I/O Device
*Driver --Missing--

Link Button: Adjusts how the selection links to the tag hierarchy.

Address

Scan Interval
1

Tag Browser Button: Opens the tag browser, so that you can select an existing tag, or create a new tag to be the data source for the matching field.

Engineering Units
%

Manual Data

☐ Enable Output

Fields: Spaces where you can enter text or numeric data to configure the tag.

OK   Cancel

OK Button: Saves the changes to this tag's properties and closes the tag properties folder.

Cancel Button: Discards the changes to this tag's properties and closed the tag properties folder.

A typical properties folder.

Related Information:

...Working With Tags – The Tag Browser – The primary tool for creating and modifying tags.

...Tag Reference – Descriptions of all tags within VTScada.

...Widget Reference – Display tag values and operator controls.

...Import and Export Tags – Work with tags properties outside of VTScada.

...See also: RememberNewTagParameters in the VTScada Admin Guide

Tag Licensing

The version of VTScada that you are running has been purchased with certain licensing restrictions that limit the number of tags that may run at the same time on one computer. The number of tags permitted by your VTScada license can be found in the About VTScada dialog that is opened when you click the About VTScada button on the VAM.
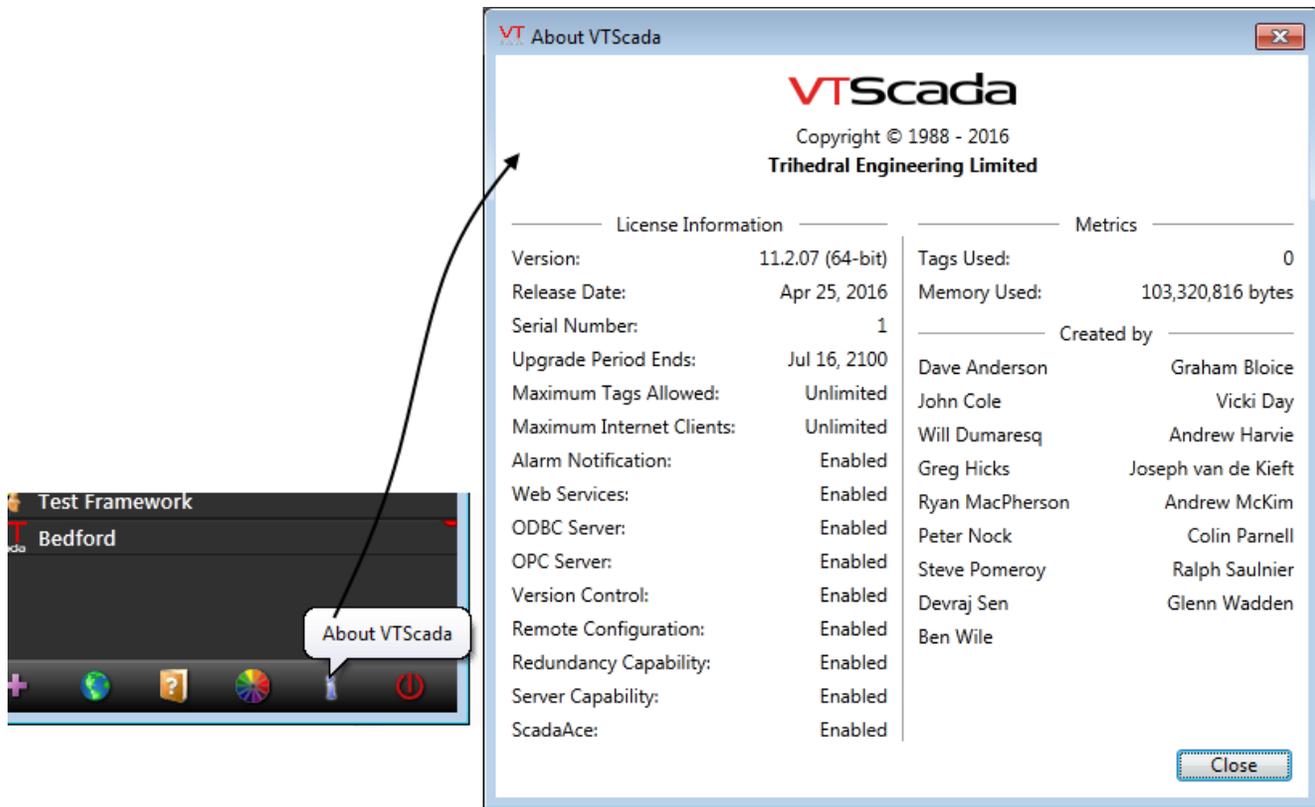
The tag limit only applies to I/O tags in all running applications. If you write code to build custom tags, then those will also be counted.

Tags that count towards your license.	Tags that do not count towards your license. (^{1}) *Some or all of the children of container tags, do count.* (^{2}) *Available only if you purchased the ScadaAce layer.*
Analog Input Analog Output Analog Control Analog Status Digital Input Digital Output Digital Control Digital Status Pulse Input Pump Status Selector Switch	Allen Bradley Driver Alarm Alarm Priority Alarm Status Analog Statistics CIP ENIP Driver *Context*(^{1}) Comm Link Sequencer Counter

String	CalAmp Diagnostic Driver
	Calculation
	Data Flow
	DDE Client
	Digital Statistics
	DNP3 Driver
	Driver MUX
	Font Value
	Function
	Historian
	History Statistics
	IP Network Listener
	Logger
	MDS Driver
	Menu Item
	Modem
	Modbus Compatible Device
	Motorola Gateway
	Motorola RTU Driver
	MPE Duplexer[1]
	MPE Duplexer Pump[1]
	MPE SC[1]
	MPE SC Pump[1]
	MultiSmart[1]
	MultiSmart Pump[1]
	MultiSmart Profile[1]
	Multi-Write
	Network Status
	Notebook
	Omron Driver
	OPC Client Driver
	OPC Server Setup
	Poll Driver
	Rate of Change
	Realm Display Setup
	Report
	Roster

	ScadaAce[1][2]
	Script
	Serial Port
	Siemens S7 Driver
	SMS Appliance
	SNMP Driver
	SQL Logger
	SQL Logger Group
	Style Settings
	TCP IP Port
	Totalizer
	Trigger
	UDP IP Port
	WorkStation Status Driver

Drawing Tags: Widgets

The data from your tags can be represented on-screen in many different ways. Animated gauges, level indicators, numeric displays, etc. can be combined with static elements such as tanks and pipes to create an intuitive operator interface.
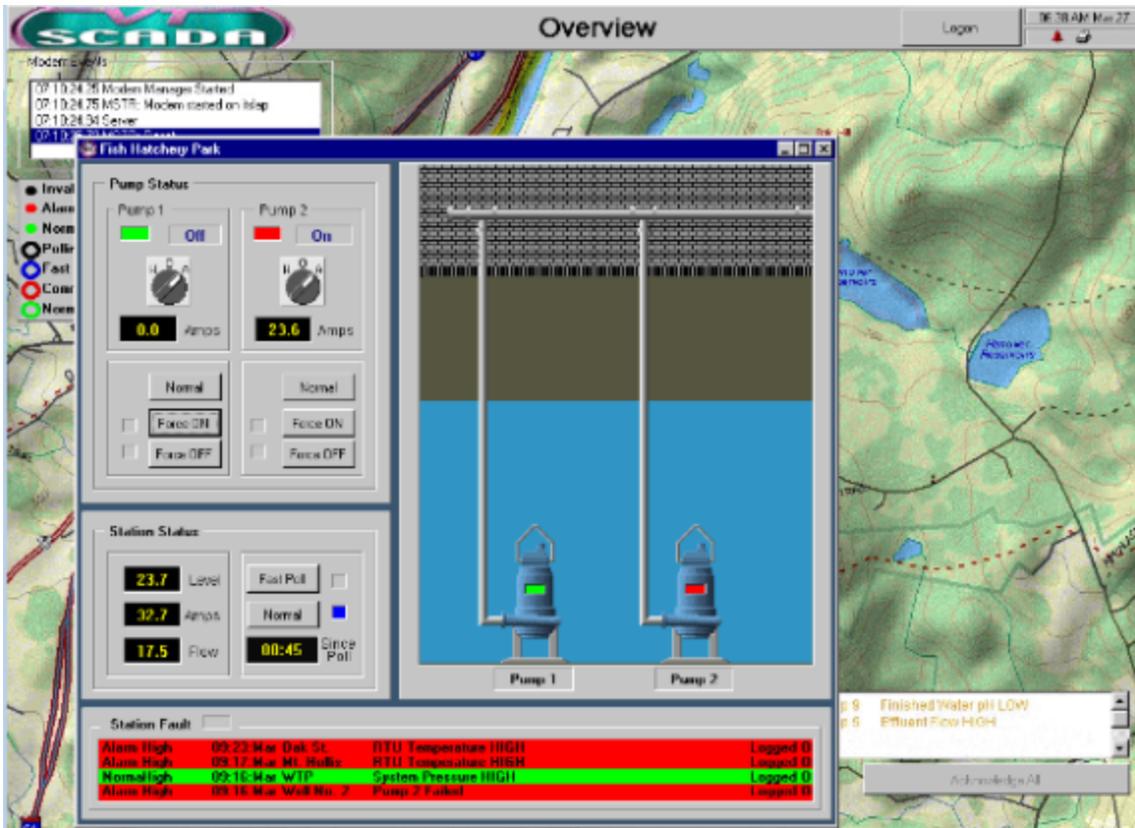
Control elements that resemble those from physical control panels, such as dials, buttons and sliders can be used to make an interface that operators will recognize immediately.

Widgets are added to pages in the Idea Studio. After being added to a page, the widget must be linked to a tag. If you select a tag in the Tag Browser, then click Draw, the Idea Studio will open, and a palette of widgets, filtered for the type of tag you are drawing, will be shown. In general, it's more common to create the tags first, then draw them as widgets, but some prefer to build the user interface first, then create the tags that will be linked to the drawn widgets. VTScada does not restrict your choice of which to do first.

Related Information:

...Panels - View and edit links between widgets and tags (or other data sources.)

...Widget Reference

Constant, Expression or Tag

When configuring a tag, you will find many properties that offer three ways to set the associated value: Constant, Expression or Tag. For example, when configuring an Alarm tag, you will have the option of setting a delay on the alarm. Depending on your choice of constant, expression or tag, the configuration parameter will look like one of the following examples:

> Note: A fourth option is also available for all tag parameter fields: Tag parameter expressions can be used for any and all fields. Tag parameter expressions are typically used for automated tag configuration.

The following is provided as a guide to help you choose the option best suited to your application.

Constant

Constant should be selected for any parameter whose value rarely, if ever, changes. A developer can return to the tag's configuration screen to edit the value, but otherwise 'constant' means 'unchanging'. Select this option for all values that should remain as you configured them.

Expression

Expressions are snippets of the VTScada programming language that you can write to create values that change with changing conditions. For example, you could create an expression that sets this parameter to the average of 3 or more other tag values. Another expression might set a value of 0 when another tag's value is below 50%, then set 100 when the other tag's value rises above 50%.

Select this option when you want the parameter's value to be based on a formula, usually referencing other tag's values. The result will be the same as if you configured a Tag Parameter Expression with the optimization option deselected.

The button opens the expression editor. The button clears the current expression. The word ": Invalid" means either that no expression has been provided, or that the result of a calculation is meaningless, such as dividing by zero.

> Note: This type of expression cannot use functions that are marked as "Script-only".

Tag

Many parameters can take their value directly from another tag. That other tag might be an operator control for setting a value, such as an alarm set point or delay. The selected tag need not be one for which you have provided an operator input – any tag with a value can be used to set the parameter's value.

Select this option when you want to have a parameter's value match another tag's value, such as when you want to provide an operator with control over a value.

The tag browser button will open the Tag Browser, from which you can select or create a tag whose value will be used. The "X" button clears the selected tag.

The link button is used with parent-child tag structures to control how the selected tag fits into the overall structure.

Related Information:

...Automated Tag Configuration – Use parameter expressions in tag configuration fields.

...Relative Tag and Property References – Linking one tag to another. See: VTScada Programmer's Guide

...Creating Expressions – Guide to the expression language, in the VTScada Programmer's Guide

Best Practices for Tag Configuration

The following rules and recommendations will help you keep your tags well organized, even if their numbers run into the hundreds of thousands.

Plan your tag structures carefully

There are considerable advantages to be gained by grouping tags according to their usage in your application. Whether it be by state, zone, station, mechanism or controller, there is always a logical way to organize your tags. In particular, make use of Context tags as a parent to group together the various tags related to each type of equipment.

Tag naming

- Tag Names Must Be No Longer than 64 Characters in Length
  This rule refers only to the tag's immediate name, not its full name, which may include a lengthy list of parent tags.
- Full Tag Names Must Be No Longer than 253 Characters in Length
  This rule refers to the tag's full name, which may include many parent tags.
- Tag Names Must Not Duplicate Other Named Items.
  You cannot name a tag "TagBrowser" or "Test", since those names are used elsewhere in VTScada.
- A tag may not be named "Value" or "Area" since those are the names of tag properties. Tags, pages, and widgets may not share a name. Tag names may not duplicate Type names – you cannot name a tag "Calculation" or "Analog Status".
  - In general, it is safest to avoid using common English words unless you add a prefix or suffix.
- Valid tag name characters include:
  - Any combination of alphanumeric characters (i.e. A through Z, and 0 through 9)
  - Spaces, except at the beginning or end of the name.
  - Period (.)
  - Symbols other than those listed below (e.g. you may use the number hatch (#), and underscore (_) characters in tag names)

- Invalid tag name characters consist of:
  - Backslash (\ used to separate parent names)
  - Forward slash (/)
  - Colon (:)
  - Asterisk (*)
  - Question mark (?)
  - Double quotes (")
  - Less than (<)
  - Greater than (>)
  - Vertical bar (|)
  - Leading spaces
  - Trailing spaces
- Tag names cannot be purely numeric.
  "57" is not a valid tag name but "Station57" is.
  An interesting side effect of this rule is that you cannot name a tag "E1" or similar, since this is a valid numeric expression: ("E1" == "0E1" == "zero exponent one").

In the event that you have entered an invalid tag name, VTScada will notify you with a warning dialog, and will clear the tag properties folder's Name field for the entry of a valid tag name.

Use the Area property to help organize tags

The Area property, although not required, is highly recommended. Configuring your tags with areas will enable you to filter your tags by area in reports and other tag selection dialogs. This will help you to locate tags quickly, especially in larger applications where there are hundreds of tags.

If you are using child tags, note that their area will automatically be the same as the nearest parent with an explicitly configured area. You can override this if a child tag should have a different area value than its parent.

For customers whose VTScada license includes the Alarm Notification System option: Roster tags, which control the list of operators to be con-

tacted in the event of an alarm, are linked to tag areas. That is, all tags sharing an area will use a matching roster.

Areas can be used to reduce the amount of memory required to run an application or to stay below licensing restrictions by allowing you to exclude certain tags from being loaded into your application at startup, on the basis of the area configured for their Area property. This is done using the [AREAS] section of the Settings.Startup configuration file for your application.

Areas can also be associated with user groups to restrict configuration access to part of an application when users are connecting over the Internet from a VTScada Internet Client (VIC).

Tags should include a clear description

In addition to a consistent and meaningful naming convention, tags should be given a clear description that will assist you and your users in determining the purpose of the tag and the equipment process it represents. The tag description appears throughout VTScada in tool tips, and in the Tag Browser, to help users determine the tag's name and its function. The Alarm Notification System will use the identify alarms using the description rather than the name (by default, where a description is available, and subject to how you configure the message template).

The importance of tag descriptions is especially important in systems where the tag names are abbreviated or follow a naming code that does not necessarily lend itself to easy identification.

Related Information:

...Naming Rules - Rules for all named objects in VTScada

Configuration Field Colors

Tag fields in the configuration panel will change color to provide more information about the type of change that has occurred in them. The following colors are used:

Field Color	Meaning
White	Unchanged during current editing session. No over-ride or parameter expression.
Green	The parameter value has been changed from its previous (or default) value in the current editing session. A special case is if the parameter's original value came from code built into the tag type's definition. In that case the color orange will be used...
Orange	A parameter value that was originally defined through parameter expression code has been overridden to use either a specific value or a new expression.
Pale Yellow	Non-inherited value. Similar to an override, but used in the Area field of a child tag if you have provided a new value rather than allowing the tag to use the Area inherited from its parent tag.
Blue	Blue indicates that the value is calculated from a parameter expression.
Red	The field's current value is outside the allowed range of values for this field.

Related Information:

...Icons That Describe Tags - Tag browser symbols.

Configuration Field Right-Click Menus

The context menu that appears when you right-click on a tag's configuration field will vary according to what the field is and how it was created.

Name field
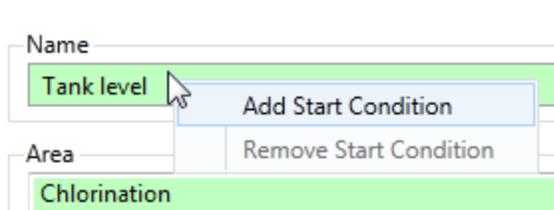
Add Start Condition
Remove Start Condition
Options for the tag name field are limited to start conditions.

Other fields, no existing code to override.

Add (Edit) Parameter Expression

Remove Parameter Expression
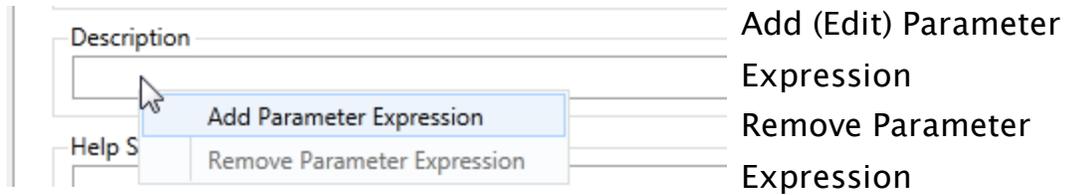
Other fields, with code to override

Inherit from parent

Add Override (Remove Override)

Add (Edit) Parameter Expression
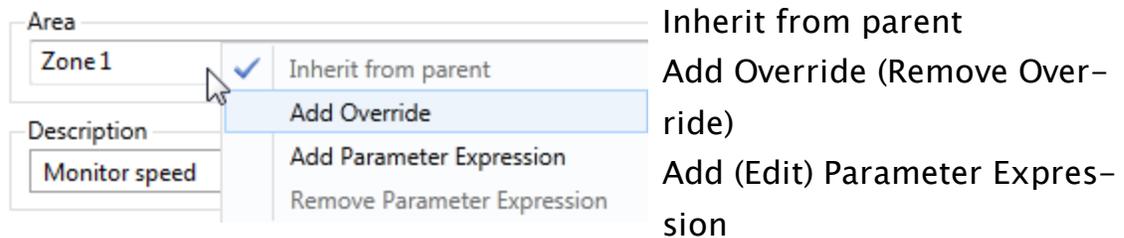
Remove Parameter Expression

If an Override has been added using this menu, the next right-click will show the option, Remove Override. You can select this to allow the field to take its value from the code within the tag.

Calculated fields are used extensively in the children of Station Tags.

Related Information:

...Automated Tag Configuration – General concepts for the use of parameter expressions.

...Create or Remove a Parameter Expression – Detailed instructions for working with expressions.

...Create a Start Condition – A start condition is a special case of parameter expression, used with the Name field.

...Relative Tag References & Parameter Expressions – Rules to find and refer to other tags in your expressions.

...Tag Configuration Overrides – Override an expression for special cases.

Tag Area Filtering and Alarm Area Filtering

Some of the applications you create might include a large numbers of tags. There may be circumstances under which you may want to prevent certain groups of tags from loading at startup to reduce the amount of memory required to run the application, or you might require that only certain groups of tags are loaded on specific workstations. You can use this functionality to ensure that you do not exceed the maximum number of tags permitted for your VTScada application due to licensing restrictions. In such situations, VTScada makes it possible to disable some tags, preventing them from loading and consuming unnecessary amounts of memory. This process is referred to as tag area filtering.
You may wish to filter Alarm tags so that specific alarms load on specific workstations. The typical purpose for this functionality is when you have a remote application installation that has one common configuration for all workstations, but on some machines you wish to see a subset of the alarms (for example, if you wished to restrict alarms to specific geographical areas). This process of alarm area filtering restricts the alarm displays and sounds to only those areas that have been enabled for the relevant machine.
The area that is being filtered is the normal Area property of the tag.

Related Information:

...Filtering Tags, Alarms and Realms – includes Tag Area Filtering, Alarm Area Filtering and Realm Area Filtering.

...Tag Licensing

Table of Type Characteristics

Name	Name in exported tables	Context type	Group memberships	Log-enabled variables
Port Tags				
SerialPort	SerialPort	*Port	Numeric, Ports, Trenders	Value (integer)

TCP/IP Port	TCPIPPort	*Port	Numeric, Ports, Trenders	Value (integer)
UDP/IP Port	UDPIPPort	*Port	Numeric, Ports, Trenders	Value (integer)
Driver Tags				
Allen Bradley Driver	ABDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
CalAmp Diagnostic Driver	DataRadioDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
CIP/ENIP Driver	CIPENIPDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Data Flow Driver	DataFlow	*Driver	Numeric, Container, Digitals, LiftstationDrivers, Trenders, Loggers	*see note 1
DDE Client	DDEDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
DNP3 Driver	DNP3Driver	*Driver	Drivers, Numeric, Trenders, LiftstationDrivers, Loggers	*see note 1
Driver Multiplexer	DriverMUX	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
MDS Diagnostic Driver	MDSDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Modbus Compatible Device	ModiconDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Motorola ACE RTU		*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Omron Host Link Driver	OmronDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1

OPC Client Driver	OPCClientDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Polling Driver	PollDriver	*Driver	Drivers, Container, LiftstationDrivers, Numeric, Trenders, Loggers	*see note 1
Siemens S7 Driver	SiemensS7Driver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
SNMP Driver	SNMPDriver	*Driver	Drivers, Numeric, Trenders, Loggers	*see note 1
Workstation Driver	WorkstationStatusDriver	*Driver	Drivers, Numeric, Trenders	*see note 1
Input and Input/Output Tags				
Analog Input	AnalogInput	*Numeric	Analogs, Numeric, Trenders	Value (floating point)
Analog Status	AnalogStatus	*Numeric	Analogs, Numeric, Trenders	Value (floating point)
Digital Input	DigitalInput	*Numeric	Digitals, Numeric, Trenders	Value (byte)
Digital Status	DigitalStatus	*Numeric	Digitals, Numeric, Trenders	Value (byte)
Pulse Input	PulseInput	*Numeric	Numeric, Analog	Value (integer)
Pump Status	Pump	*Numeric	Digitals, Loggers, Numeric, Trenders	Value (byte)
String	StringIO	none	Trenders	Value (text)
Output Tags				
Analog Control	AnalogControl	*Numeric	Analogs, Numeric, Trenders, Outputs	Value (floating point)
Analog Output	AnalogOutput	*Numeric	Analogs, Numeric, Trenders, Outputs	Value (floating point)
Deadband Control	DeadbandControl	*Numeric	Numeric, Trenders	Value (floating point)

Digital Control	DigitalControl	*Numeric	Digitals, Numeric, Trenders, Outputs	Value (byte)
Digital Output	DigitalOutput	*Numeric	Digitals, Numeric, Trenders, Outputs	Value (byte)
MultiWrite	MultiWrite	*Numeric	Outputs	Value (floating point)
Selector Switch	SelectorSwitch	*Numeric	Numeric, Trenders	Value (short)
Trigger	TriggerTag	*Trigger	Digitals, Numeric, Trenders	Value (byte)
Alarm Tags				
Alarm	AlarmPoint	*Alarm	Digitals, Numeric	Value (logged by VTS)
Alarm Priority	AlarmPriority	*Alarm Priority	Digitals	Value (do not log)
Alarm Status	AlarmStatus	*Alarm Status	Numeric, Trenders	Value (integer)
Modem	Modem	*Modem	Numeric	Value (integer)
Roster	Roster	*Roster	Digitals, Numeric	Value (Boolean)
SMS Appliance	SMSAppliance	(none)	Numeric	Value (integer)
Logging and Reporting Tags				
Historian	HistorianTag	*Historian	Numeric	(none)
Logger	LogPoint	*Logger	Numeric	(none)
Notebook	Notebook	*Notebook	Trenders, Loggers	Value (text)
Report	Report	*Report	(none)	Value (0 or 1)
SQL Logger Group	SQLLoggerGroup	*SQL Logger Group	(none)	(none)
SQL Logger	SQLLogger	*SQL Logger	Numeric	Value

				(matches tag being logged)
Calculation and Inquiry Tags				
Analog Statistics	AnalogStatistics	(none)	Container	Do not log. Selected child tags may be logged.
Calculation	Calculation	*Numeric	Loggers, Numeric, Trenders	Value (floating point)
Counter	CounterTag	*Numeric	Analogs, Numeric, Trenders	Value (integer)
Digital Statistics	DigitalStatistics	(none)	Container	Do not log. Selected child tags may be logged.
Function	Function	*Numeric	Numeric, Trenders	Value (floating point)
History Statistics	HistoryStats	*Numeric	Numeric, Trenders	Value (floating point)
Network Status	NetworkStatus	*Numeric	Digitals, Loggers, Numeric, Trenders	Value (byte)
Rate of Change	RateOfChangeTag	*Numeric	Analog, Numeric, Trenders	Value (floating point)
Totalizer	TotalizerTag	*Numeric	Analog, Loggers, Numeric, Trenders	Value (floating point)
Script	Script	*Script	Numeric, Trenders	Value (floating point)
Station Tags				
Multismart Station	Multismart	*Station	Container, Stations	Value (matches the driver's

				value)
MPE SC Station	MPESC	*Station	Container, Stations	Value (matches the driver's value)
MPE Duplexer Station	MPEDuplexer	*Station	Container, Stations	Value (matches the driver's value)
Configuration Tags				
Comm Link Sequencer	ContextTag	Defined by configuration	Numeric, Trenders	Value (Text)
Context	ContextTag	Defined by configuration	Container	(none)
Font Style	FontValue	*Font Style	(none)	(none)
IP Network Listener	IPNetworkListener	*Port	Numeric	Value (byte)
OPC Server Setup	OPCServerSetup	*Driver	Numeric, Trenders, Loggers	
Realm Display Setup	RealmDisplaySetup	*Realm Display Setup	(none)	(none)
Style Settings	StyleSettings	*Style Settings	(none)	(none)

*Notes:
1: See: Communication Driver Log-Enabled Variables

Common Tag Configuration Properties

Many of the standard VTScada types have similar or identical tabs on their tag properties folders. For example, all tags have an ID tab, which contains properties that help to identify the tag and make it unique from all other tags in the system. All input and output tags have an I/O tab where you can specify the parameters that associate the tag with an I/O device.

The topics in this chapter describe the properties that are commonly seen in VTScada tag configuration folders.

Related Information:

...The ID tab

...I/O Tab

...Alarm Tab and Alarm Setup Tab

...Address Select

...Logger or Historian Tab

...Merit Tab and Quality Tab

...Display Tab

...Owner Tab

...Scaling Tab

...Script Tab

The ID tab

Every tag's configuration panel will have an ID tab that includes the same four elements. All tags of all types must be identified and all will share the same identifying characteristics: name, area and description. The fourth element, Help Search Key, is provided for those who have created their own compiled help files and wish to link specific tags to topics within that file.
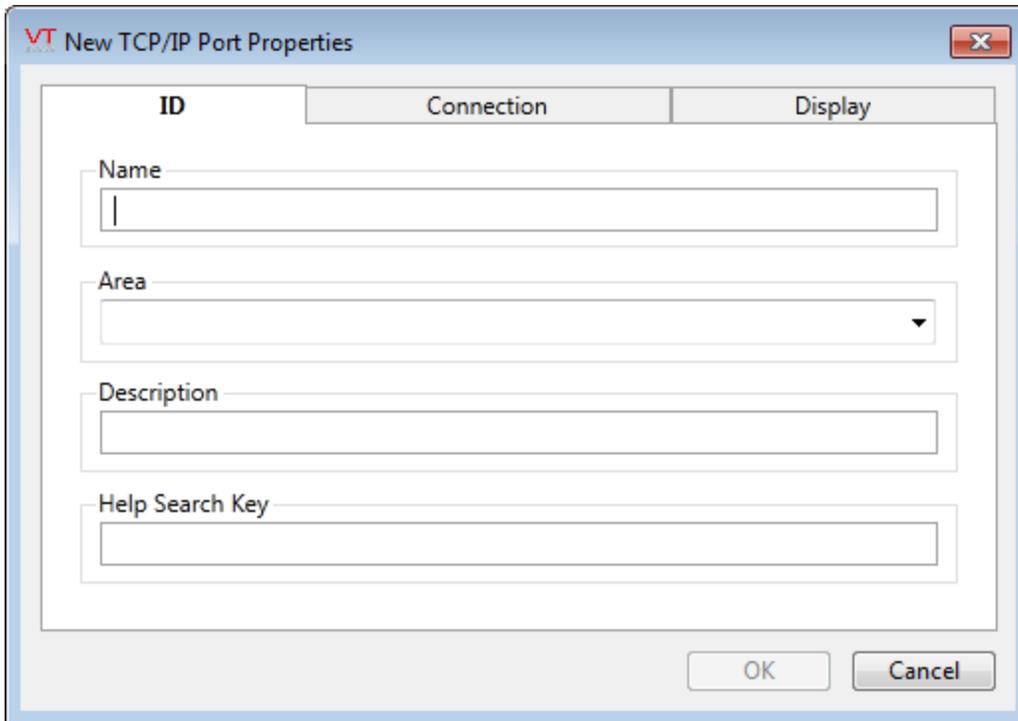
Three types: Report tags, Context tags, and Data Flow RTU driver tags, will include an extra field in their ID tab. See the descriptions of those tags for more information.

Name

There are rules for valid tag names (see related information). If the tag is a child of another tag, the parent names will be displayed in a separate area before the name field. See: Parent-Child Tag Structures.The current tag's name, excluding its parents, is also called the "friendly name".

You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typ-

ing an existing area name is not recommended since a typo or mis-
spelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool
is unnecessary. An area definition will exist as long as any tag uses it and
will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. Use the description field to give each tag a
human-friendly note describing its purpose or location. While not man-
datory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be
selected for a report and also on-screen when the operator holds the
pointer over the tag's widget. The alarm notification system will refer to
tags by their description rather than name unless otherwise configured.
The description field will store up to 65,500 characters, but this will
exceed the practical limits of what can be displayed on-screen.

Help Search Key

Used only by those who have developed their own Compiled Help Module
(CHM) documentation (such as this guide). To see this system in action,
open the ID tab of any tag's configuration panel and press the F1 key.
This guide will immediately open to display information about that
screen because a link exists between the topic in this guide and that
screen.

When a user-defined help search key is defined for a tag, a Help button
will appear between the tag properties folder's OK and Cancel buttons
the next time the tag properties folder is opened. Further, when any of
the tag's widgets is right-clicked, selecting the Help shortcut menu
option will open the assigned help topic.

In addition to creating the help file, defining a numeric topic id and set-
ting that id as the Help Search Key, you must also tell your application
how to find your .CHM file.

Related Information:

...Tag Identification – Valid tag names.

...Create a Start Condition – Don't start tags that are not required.

...See: "Integrating Custom Help Files into VTScada" in the VTScada Admin Guide

I/O Tab

All of the input and output, status and control tags will have an I/O tab in their tag properties folder.

The I/O tab consists of properties used to identify and establish a connection to the communication driver tag being used to communicate with your physical I/O device. This is done by identifying the communication driver tag, the address at which this basic I/O tag is to read or write data, and the rate at which the I/O device should be scanned for data. A manual data option is provided to allow you to set a constant value for the tag, independent of any actual I/O, for configuration and testing purposes.

Scan Interval. In general, it is desirable to limit the number of different scan intervals for a particular I/O Device, to optimize the reading to a minimal number of blocks and improve overall system update performance.

Typical I/O tabs for both Analog Status and Digital Status tags:

Digital input tags (including Digital Input, Digital Status and Pump Status tags) can have single or double bit addresses.

A single-bit digital status reads a value of 0 or a 1 from one address on an I/O device. An example of a single-bit digital status would be a pump's status, where the pump is either off (0), or on (1).

A double-bit digital status reads a value of 0 or 1 from each of two addresses on an I/O device. An example of an equipment process requiring a double-bit digital status might be a valve with 4 states: open, closed, action in progress, or error.

The I/O tab of Analog Input, Analog Output, Digital Input and Digital Output tags will include a field to set the engineering units. For the corresponding Status and Control tags, that parameter is configured on the Scaling tab.

Related Information:

...Data Suffixes for Tag I/O Addressing

...Allen-Bradley Driver I/O Addressing

...CalAmp Driver I/O Addressing

...CIP Driver I/O Addressing

...DNP3 I/O Addressing

...DriverMUX I/O Addressing

...MDS Diagnostic Driver I/O Addressing

...Modbus Compatible Device I/O Addressing

...Modbus Plus PLC Addressing

...Motorola ACE Driver I/O Addressing

...Omron I/O Addressing

...OPC Client Driver I/O Addressing

...Siemens S7 Driver I/O Addressing

...SNMP I/O Addressing

...Workstation Status Driver I/O Addressing

Alarm Tab and Alarm Setup Tab

There are three common versions of the alarm or alarm setup tab. In the Analog Input and Digital Input tags, the alarm tab displays the list of Alarm tags that are attached, and provides a shortcut to add or remove Alarm tags. In Analog Status and Digital Status tags, this same tab has the label "External Alarms" in order to distinguish the attached Alarm tags from the built-in alarms.
Analog Status, Pump Status and Digital Status tags all have an alarm tab or alarm setup tab, which is used to configure the built-in alarm for that type. These three example images show the differences between the versions:

New Analog Input Properties

ID | I/O | Scaling | Owner | Merit | Display | **Alarm** | Script | Logger

Alarm Contributors

Name	Description

Add   Delete

OK   Cancel

New Analog Status Properties

ID | I/O | Scaling | **Alarm** | External Alarms | Quality | Display | Historian

Low Alarm Priority
None

High Alarm Priority
[2] High

Low Alarm Setpoint
0
● Constant ○ Expression ○ Tag

High Alarm Setpoint
100
● Constant ○ Expression ○ Tag

Low Alarm Deadband
0
● Constant ○ Expression ○ Tag

High Alarm Deadband
0
● Constant ○ Expression ○ Tag

Low Alarm Delay (s)
0
● Constant ○ Expression ○ Tag

High Alarm Delay (s)
0
● Constant ○ Expression ○ Tag

Disable Low Alarm
☑ Disable
● Constant ○ Expression ○ Tag

Disable High Alarm
☑ Disable
● Constant ○ Expression ○ Tag

Low Alarm Rearm Time (s)
3600
● Constant ○ Expression ○ Tag

High Alarm Rearm Time (s)
3600
● Constant ○ Expression ○ Tag

☐ Low Alarm Rearm Enable
☐ Low Alarm Popup Enable

☐ High Alarm Rearm Enable
☐ High Alarm Popup Enable

Sound

OK   Cancel
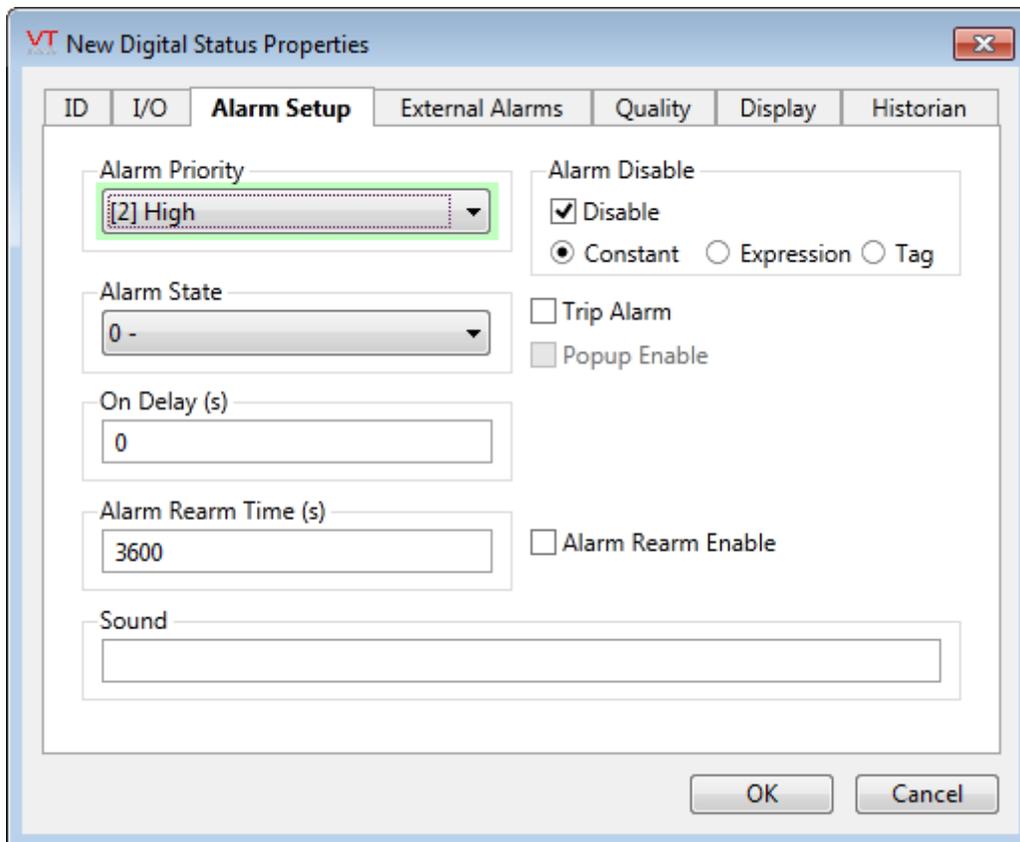
The common feature in all three variations is that once fully configured, the value of the tag will be monitored by the Alarm Manager, and an alarm will be triggered when values reach a given value.

> Note: Alarm tags created using the Add button in the alarm tab will be created as children of that tag. You are not restricted to the built-in alarms.

Complete details for configuring the alarm tab of any particular type are provided with the description of that type.

Address Select

I/O tags that have been configured to use the SNMP driver or the OPC Client driver will have access to an Address Select dialog through a button on the I/O tab of their configuration dialog.
Example of an Analog Control showing the MIB button, visible only after an SNMP driver has been selected as the I/O device.
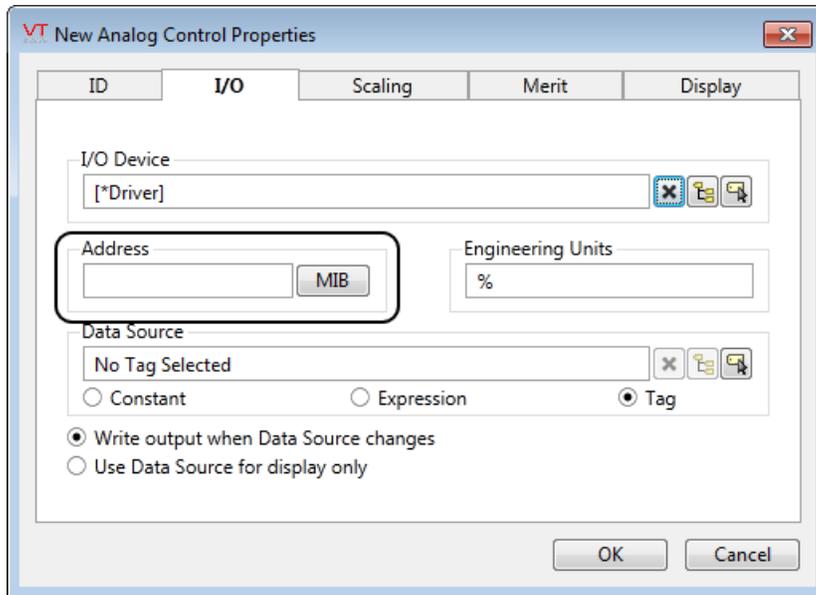
Related Information:

...SNMP I/O Addressing – Describes how the MIB button can help you to build an SNMP address.

...OPC Client Driver I/O Addressing –OPC addresses and how they can be built using the Address Select.

Logger or Historian Tab

How Historians Differ From Loggers
Many newer application have no logger tags at all. Most I/O tags have a built-in connection to the Historian, making a Logger unnecessary

Note: Do not attach a logger to any tag that includes a built-in Historian connection.

Loggers control *when* data is recorded. Historians control *where* the recorded data is stored. Every Logger tag must use a Historian tag in order to save logged data.

A single Historian tag can be used by all the Loggers in an application, but a unique Logger must be used for every tag whose value is to be recorded, provided that the tag in question does not have a built-in connection to a Historian.

Some tags, such as the Analog Input tag and the Digital Input tag, do not have internal logging defined and therefore must have an external Logger tag attached if the data is to be stored. Other tags, such as the Analog Status, do have internal logging and therefore need only a Historian tag attached, to control which database the data is to be logged to.

Typical Logger tab:
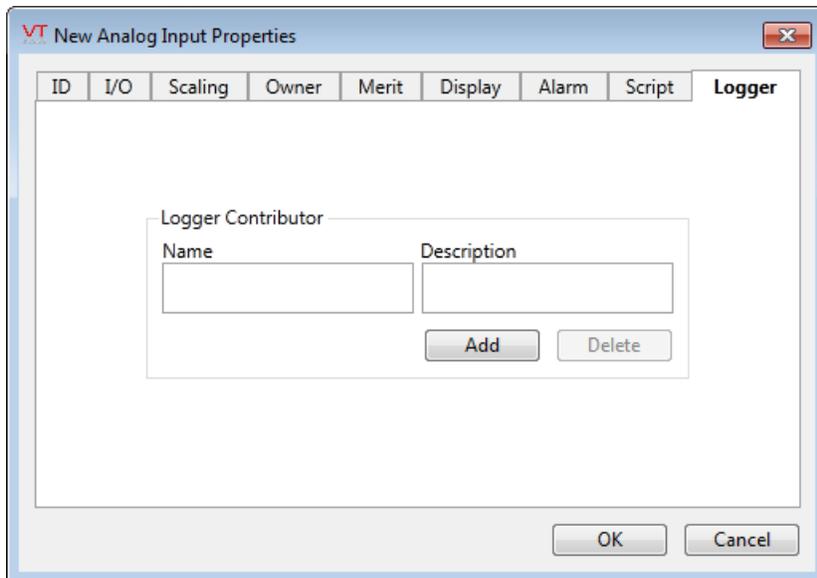
Input and output types will have a Logger tab on their tag properties folder. For tags that do not have a built-in connection to a Historian, you must attached a Logger tag if you want to store a record of the tag's values.

Status and control tags have built-in connections to a Historian tag and therefore do not have a Logger tab since there are no user-configurable settings.

If a Logger tag is being used, only one may be associated with any given tag. If that tag has a built-in connection to a Historian, you must not also attach a Logger tag.

Typical Historian tab:
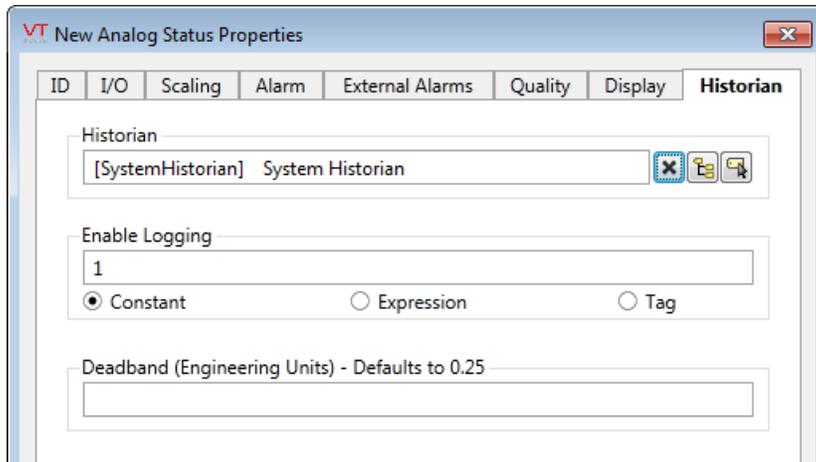
Most tag types that read or calculate values will have a Historian tab in their tag properties folder. This is used to select the particular Historian database that will be used to store the data recorded from the current tag.

Note: Do not change historians after data has been collected. Data will be retrieved only from the attached historian, meaning that for all intents and purposes, you will lose the data that was recorded with the earlier choice.

Related Information:

...Historian Tags

...Logger Tags

...See also: DefaultCalculationDeadbandFractionOfFullScale in the VTScada Admin Guide

Merit Tab and Quality Tab

Many types of tag include a Merit tab or Quality tab on their tag properties folder. "Merit" and "Quality" are two names for what is essentially the same tab. For input and status types, the screen will look like the following examples:

For output and control types, there is no "Quality Based On" field.
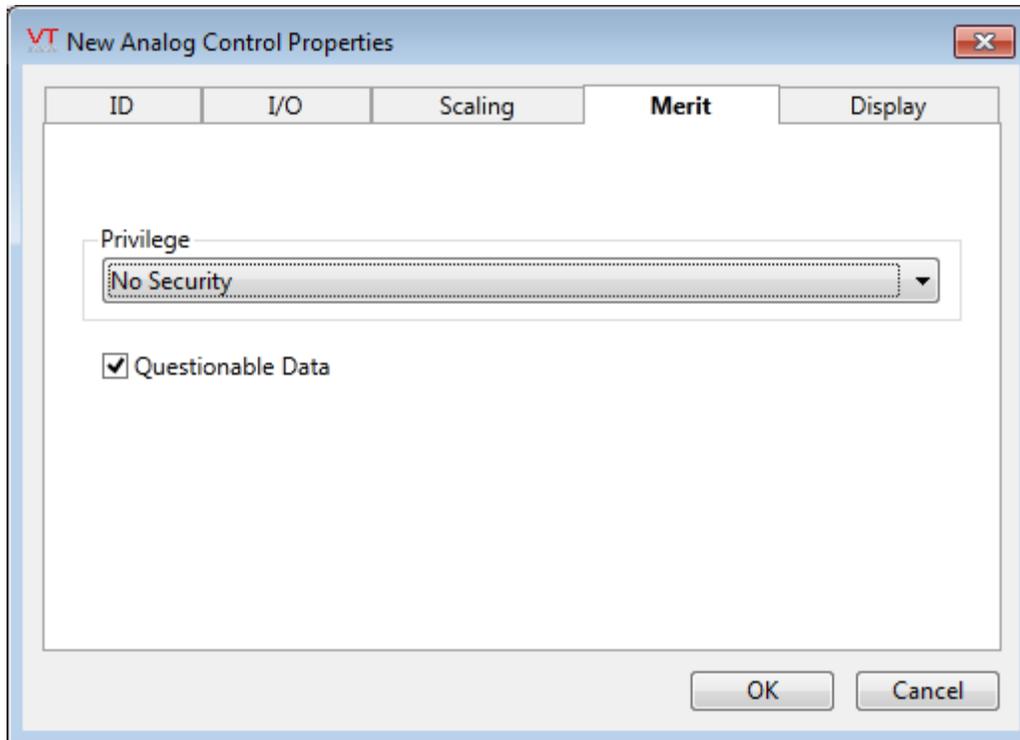
Questionable Data

Use this field to flag the tag's data in the event that you sus-
pect the values it is reporting might not be correct, or when
this tag has initially been created and you wish to ensure that
its data is marked for extra monitoring.
After you have verified the accuracy of each tag's data, you can
deselect its questionable data marker, thereby keeping track of
which tags you have verified and which you haven't.

Quality Based On

The Quality Based On field can be tied to a tag that monitors
driver communication errors, a register in the PLC or RTU, or
other indication of system health. The tag you are configuring
will use that value to set an internal variable named
QualityIssue. You may test this value in an expression:
[TagName]\QualityIssue.

QualityIssue will be a 0 or 1. "Good Quality" is defined as a value of "0 or Invalid" and "Bad Quality" is defined as any valid, non-zero value.

VTScada programmers can create container tags that make use of the quality value of contributors, which are either Analog Inputs or Digital Inputs.

Security Privilege

A Security Privilege may be applied to any tag that can write data. Using the drop-down list, you can select an existing application-specific privilege that will restrict who is allowed to use this tag to write data. Once a value is set, users will need to have that privilege granted to them before they can send a signal to an I/O device.

Application privileges are added using the Administrative Settings security dialog.

Related Information:

...Refer to the VTScada Admin Guide for "Protect Pages and Output Tags" – Create your own privileges for protection.

Display Tab

Content will vary by tag type.

Style

Many widgets can be configured to take their display characteristics from a selected Style Settings tag, rather than requiring that you set color and text for each possible state. If you select a particular Style Settings tag, then all widgets that are style-aware will be displayed using the settings in that tag unless the widget is linked to an alternate Style Settings tag.

Reference:

...Style Settings Tags

Order

For I/O tags that may be displayed as part of a Site page (or, Station Page in versions of VTScada prior to release 10.2), the order of the elements in the display can be controlled by the number on this tab. Use this field if you have planned which tags will be shown on a given Site page, and want to control the order in which they are listed.
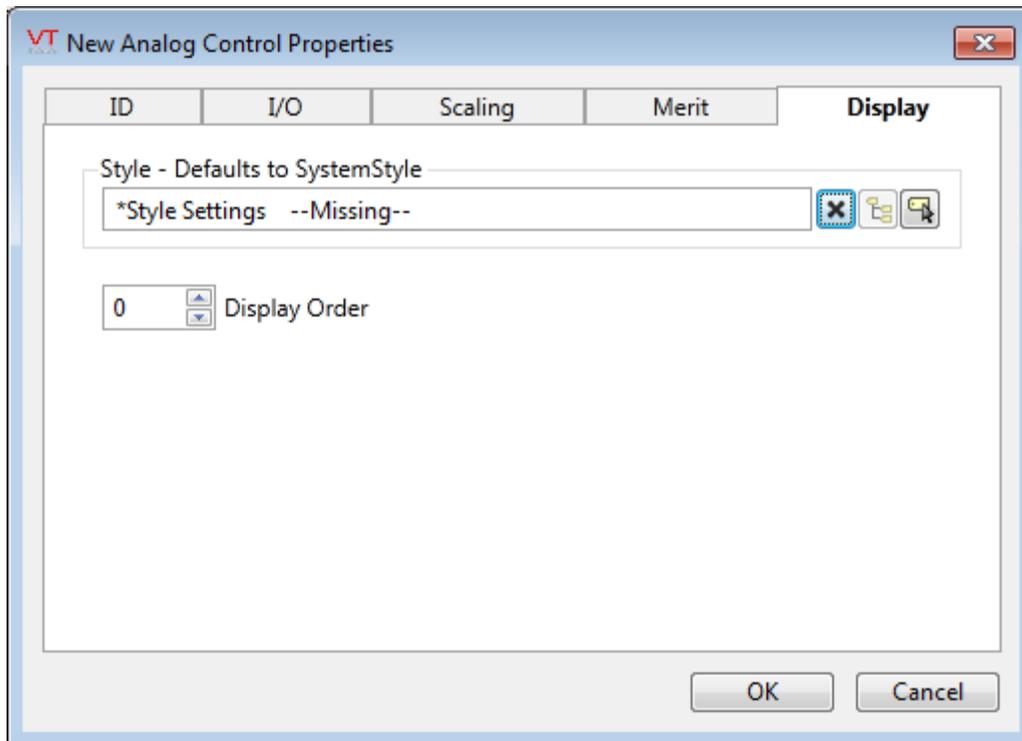
On Text / Off Text

Found only in Digital tags and used only in widgets that are designed to display these values for the matching state of the tag. Set as appropriate for the equipment being controlled or monitored, On/Off, Running/Stopped, Open/Closed, ...

Owner Tab

Input and Output tags, both analog and digital, have an Owner tab on their tag properties folder.

VTScada supports the concept of owner/contributor (also referred to as container/contributor) relationships among tags, where multiple

contributor tags can supply their values to an owner or container tag. An owner tag's value is determined by a mathematic function of the values of its contributor tags. An example of an owner/contributor relationship might be at a power generation station, where the owner tag would calculate the overall power output for the entire station based on the value of its many contributor tags. Within a contributor tag, you can select the owner tag and specify how the contributor's value should be used.

> Note: You must write your own Owner tag. This requires a solid knowledge of VTScada programming techniques. Reference information can be found in Programming Custom Tag Types within the VTScada Programmer's Guide .

Owner

The Owner field can be used to specify a tag to which this contributor should supply its data. Establishing an owner/contributor relationship also enables standard tags to monitor custom tags without prior knowledge of the number of each type required.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. This requires the owner to have a variable (an array) named DataQuality. The contributor will then pass it's quality value to an element of that array.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box enables you (as the programmer of an owner tag) to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box enables you to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Set Owner\Data(n)[...] To Value

When selected, the Set Owner\Data0[...] To Value check box enables you to set the value of this contributor tag as the first

element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Scaling Tab

All analog I/O tags include a Scaling tab on their tag properties folder. All will include the four basic fields that scale raw values to engineering units, but the Analog Status tag also provides display-scaling properties and moves the Engineering Units field to this page.
Sensors and other hardware typically work with a range of discrete values that are much more finely divided than standard engineering units such as degrees. For example, a temperature sensor might report values within a working range of 0 to 4095. Depending on the sensor's calibration, the low end of the scale, 0, might represent 32deg. At the high end of the scale, 4095 might represent 200deg. Given these points, and assuming a linear relationship, one can then calculate the actual temperature indicated by any value output from the sensor.

Note: You can set your own default values for the scaling fields. Set properties AnalogUnscaledMinDefault, AnalogUnscaledMaxDefault, AnalogScaledMinDefault, AnalogScaledMaxDefault as required.

Working the other way, one can also calculate the equipment value required when an output tag is used as set point.

Calculated Tag Value

Scaled Data From Equipment (y-axis)
Unscaled (Raw) Data From I/O Device (x-axis)

High Value, Low Value
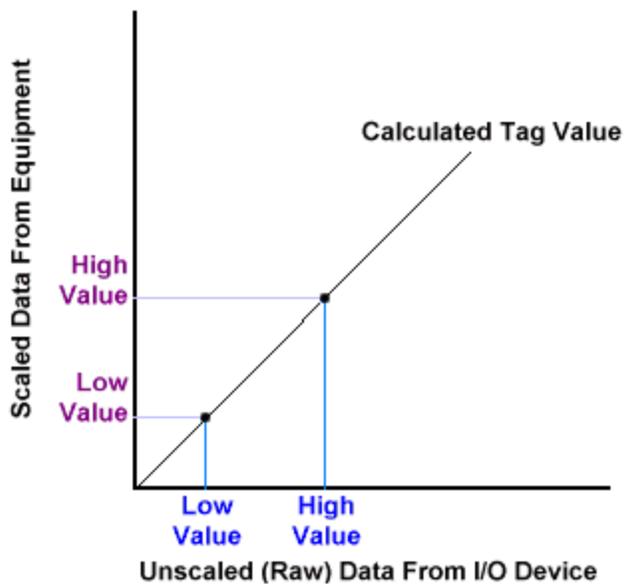
The Engineering Units property is used to define the units represented by the scaled value. For example, "%", "rpm", "psi", or "kPa".

The high and low scaled values fields of the Analog Status tag are used to set a default range for use when viewing the tag's trend graph in the Historical Data Viewer, or with widgets whose displayed scale range can be adjusted. These display scaling fields have no effect on the tag's value.

Related Information:

...Refer to the VTScada Admin Guide for the following properties:

...AnalogUnscaledMinDefault

...AnalogUnscaledMaxDefault

...AnalogScaledMinDefault

...AnalogScaledMaxDefault

Script Tab

Script tags reference the script-code within a user-defined module (program file), and execute that code when the value of the associated tag changes. Script tags can be associated with this tag using the Script tab.

Script Contributors

The Script Contributors list identifies the name(s) and description(s) of the existing Script tag(s) that will execute a script based on the value of this tag (Script tags and their properties are described in Script Tags). The associated Add button and Delete button may be used to modify the Script Contributors list.

Note: Any Script tags identified in the Script Contributors list are not properties in this tag; rather, this tag is a property of the Script tag(s), as identified in the Script tag's "In Tag Scope" property (see Script Tag Properties: Execute Tab).

Add

Opens the Tag Browser so that you can select an existing, or create a new Script tag to monitor the value of this tag. Information on the required configuration can be found in Script Tags topic.

Delete

Use the Delete button to disassociate a selected Script tag from this tag. The Script tag is not deleted by this operation, and can be associated with another tag if you so choose.
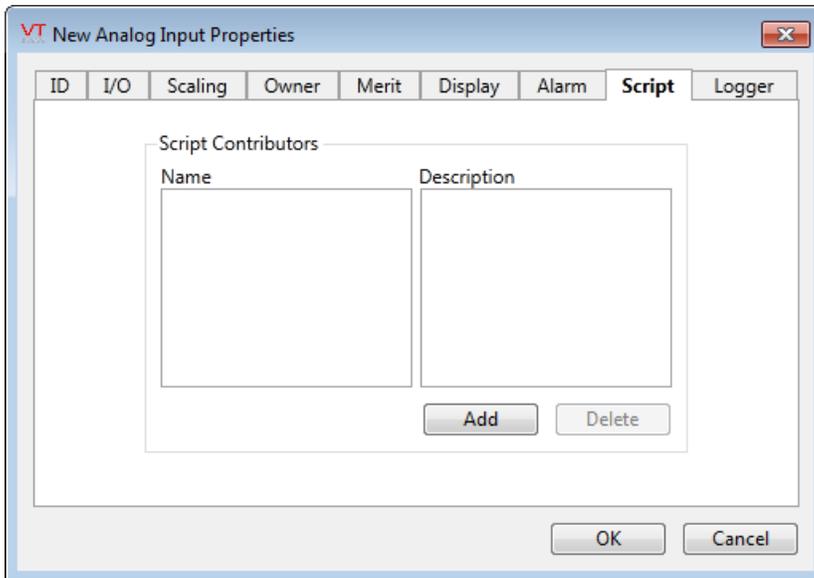
Related Information:

...Script Tags

Working With Tags – The Tag Browser

The Tag Browser is the primary tool for creating, viewing, drawing and managing the tags in your application.

Open the Tag Browser:

From the title bar:

1. Click the Tag Browser button.
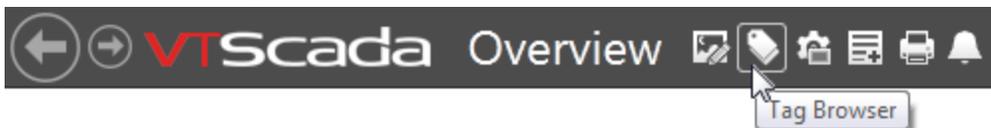
From the Idea Studio:

1. Open the Home ribbon.
2. Click the Browser button.
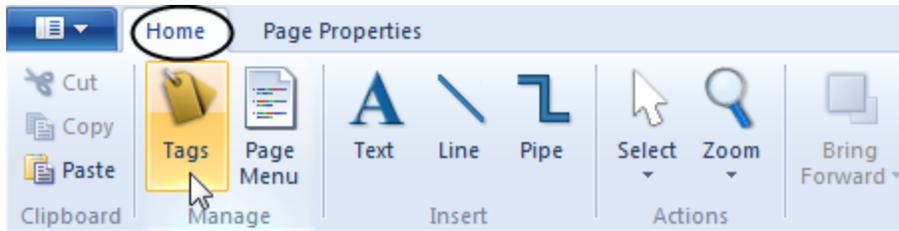
The tag browser is designed to work much like Windows Explorer, where tag structures are similar to the folders and files on your workstation. Every VTScada application comes preconfigured with several tags that are used for basic functionality. These include Font Style tags, Alarm Priority tags, and Notebook tags. Also, palette items including widgets, shapes and images are organized using Menu Tags. These are what you see when you first launch the tag browser. Rest assured that they do not count against your licensed limit. Only I/O tags and tags that you code from scratch are counted towards the license limit.
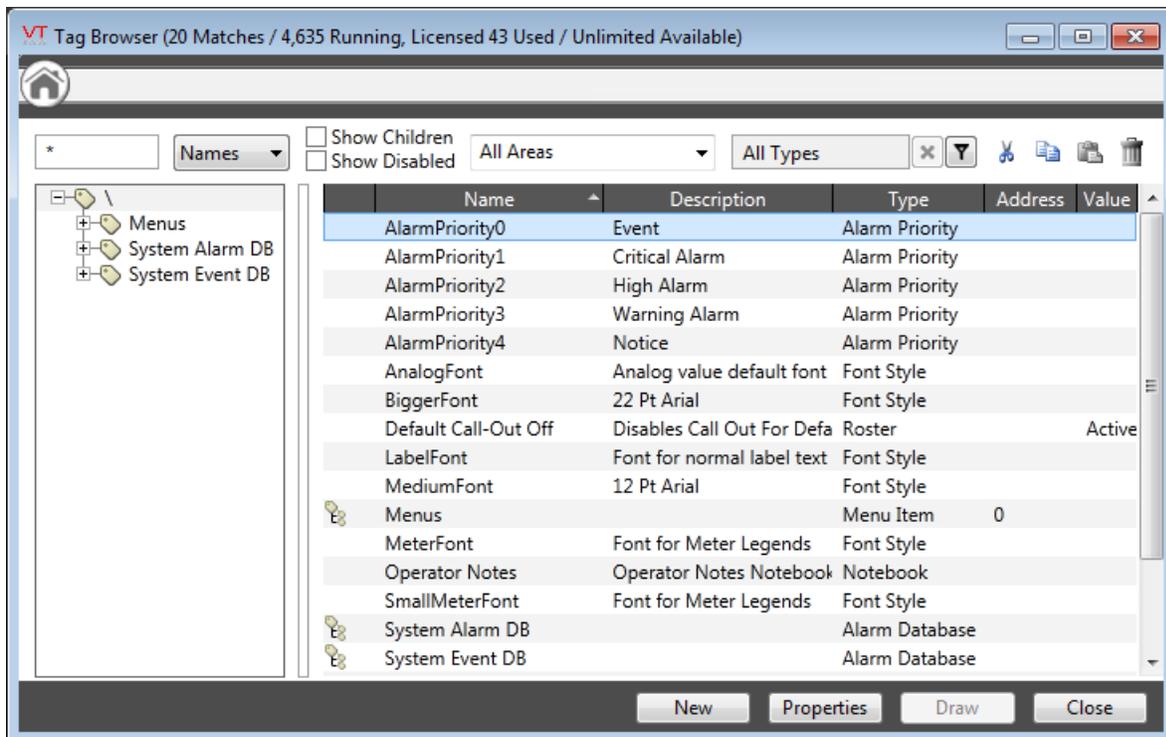
Parts of the Tag Browser:

The Address Bar.

The top bar displays the context of the tag you are viewing, within a tag

structure. It works in much the same way as the address bar in Windows Explorer:
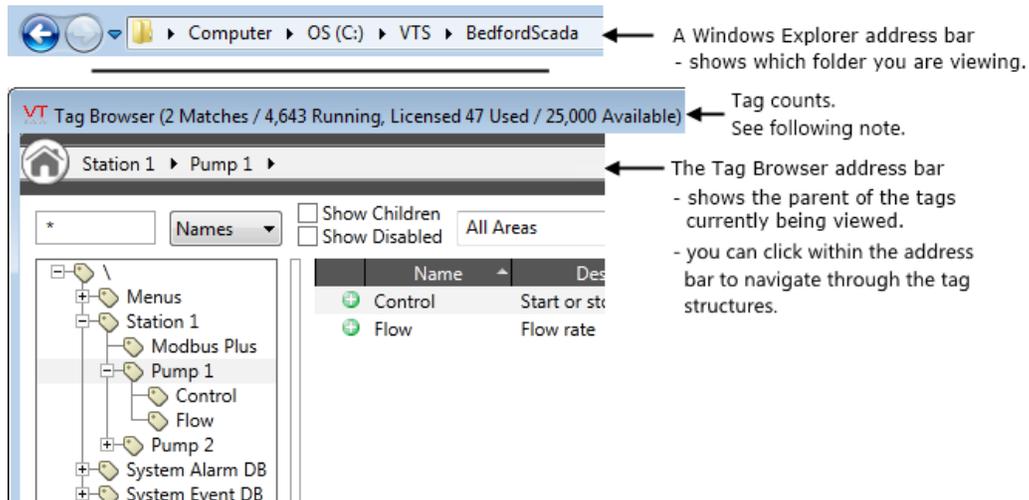
Tag Counts: The number of **matches** refers to the current filter, including the selected node in the tag tree. In other words, the number of tags displayed in the main window.

The number **running** includes menu item tags and is normally a very large number.

The number of **licensed** tags **used** shows how many I/O, calculation and other tags that count against your license limit have been created and are enabled.

Finally, the number **available** shows how many more I/O, calculation and other license-limited tags you may create.

Filter Controls

Filters allow you to view just a specific group of tags. For example, to quickly review the I/O address and the current value of all tags of a certain type (for example, Analog Status tags in a named area), check Show Children and filter the Tag Browser's display.
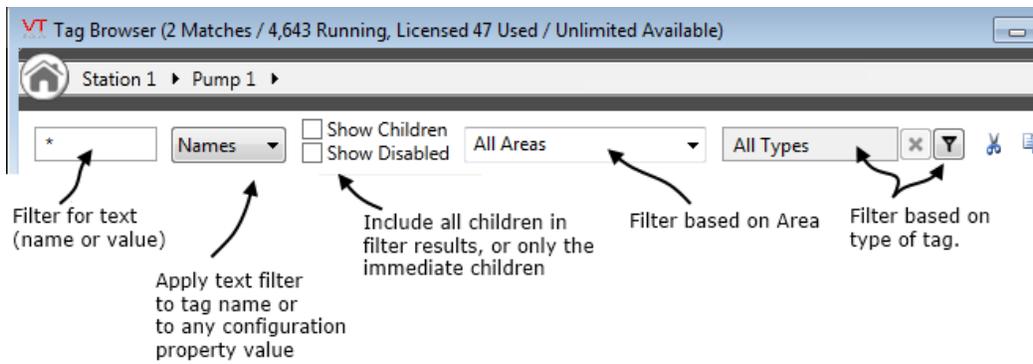
Tree View Window

The window down the left side ("tree view window") will display all of the parent tags in your application. This is similar to the folder view in Windows Explorer. You can close or open the tree view window by double clicking on the vertical bar that separates it from the main tag window.
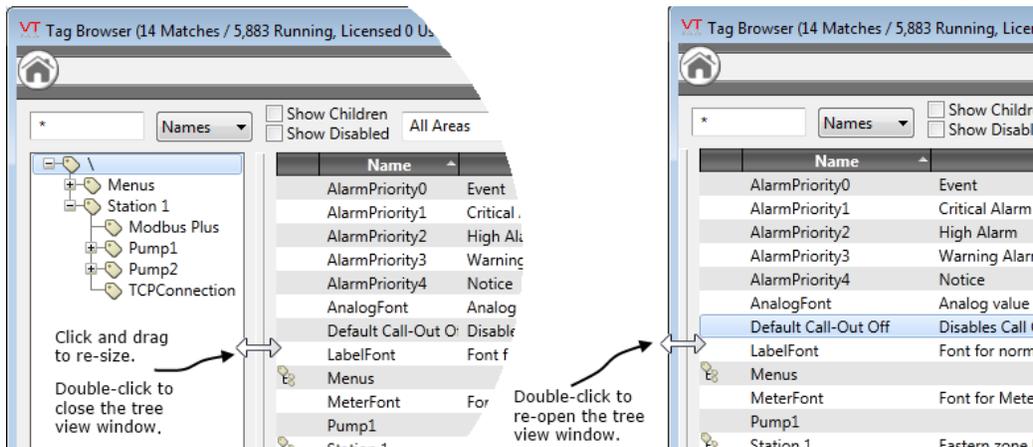
Tag List Window

The main area of the browser, showing all of the tags in your application. The list can be sorted by any of the column headings by clicking on the desired heading. Click a second time to reverse the order.
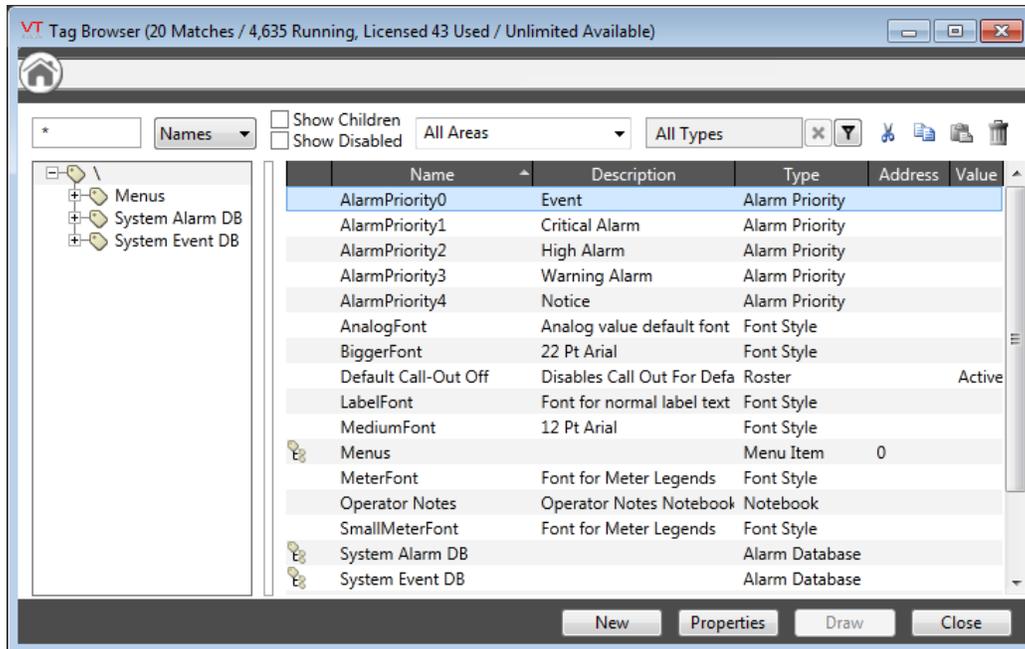
Any given tag may or may not be visible, depending on:

- The filter control in effect.
- The selected parent tag.
- Whether or not the "Show Children" filtering option is selected.
- Whether the tag is disabled, and whether the "Show disabled" filtering option is selected.
- Whether realm-area filtering is in effect.

The Menus Group

The contents of both the page menu and the palettes in the Idea Studio dialog are organized using Menu Item tags. Since the order of these tags is reflected in the organization of menu and palettes, extra control buttons are added to the Tag Browser when you are working within the Menus group.

When creating new tags, your choices will be limited according to the parent you are adding the tag below. Within the menus group, you can create only tags relevant to menus or palettes (drawing tools), such as pages, menu items, folders, and dividers.
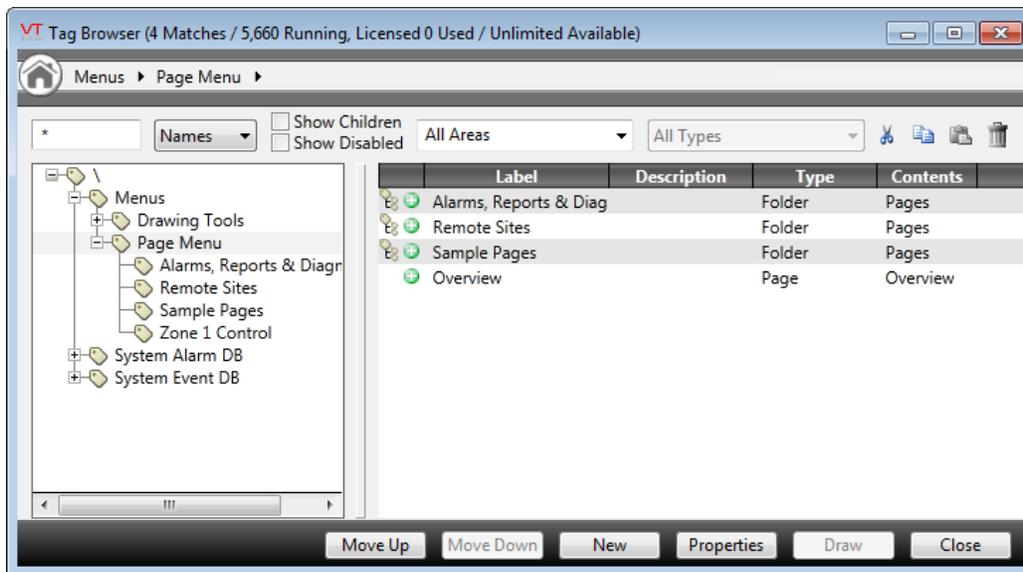
Command Buttons

All the commands that apply to a tag can be accessed by right-clicking on that tag to open its context menu. In addition, a set of copy & paste commands can be found at the top of the tag browser and the most commonly used commands are available in buttons at the bottom of the browser.
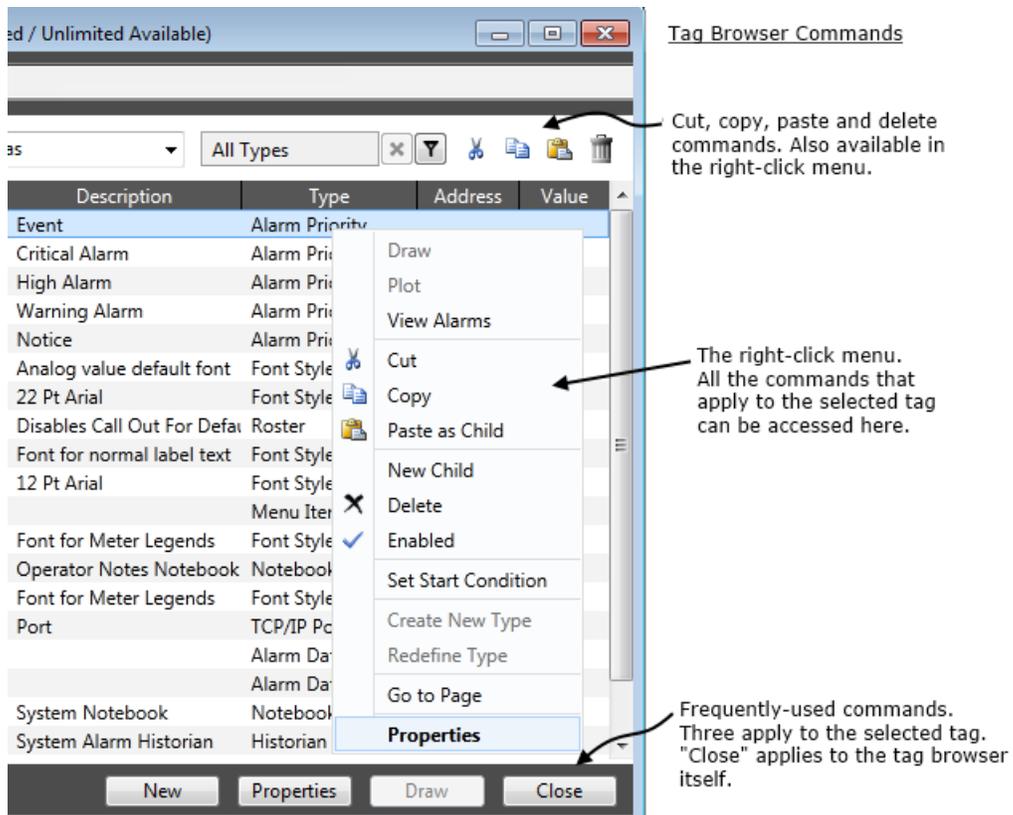
Shortcut: Modifying Tag Properties...

The Tag Browser is not the only path to edit a tag's properties. In a running application, if you can see the widget that's linked to the tag you want to edit, you can right-click on that widget to open a context menu. The contents will vary by tag type, but the Properties option will be available for every tag. Most of the other options in this menu open specific pages of the full properties dialog.
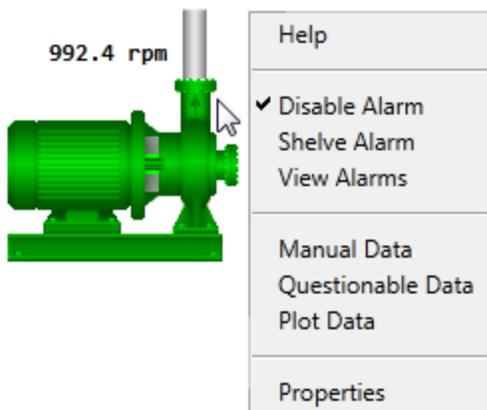
Your security account must have tag configuration privileges in order to proceed. This right-click menu can be disabled by editing the properties of the widget in the Idea Studio.
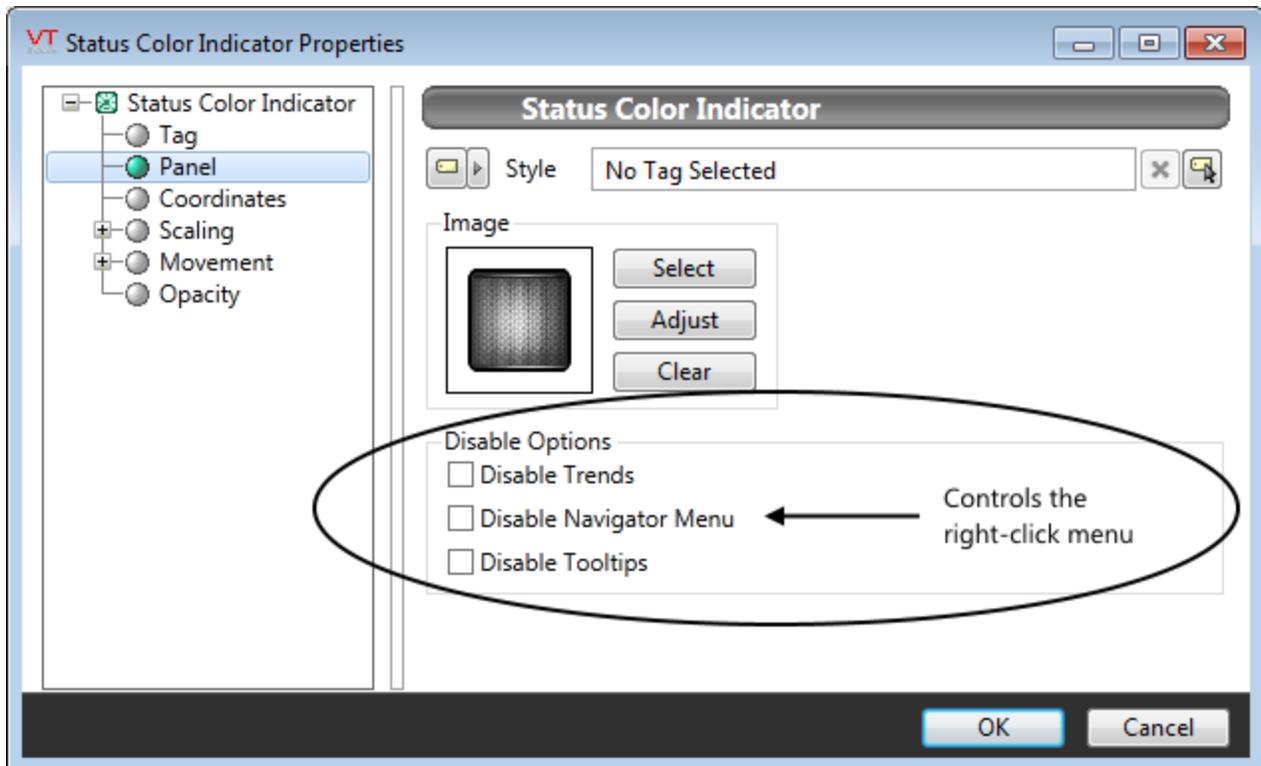
Related Information:

...Tag Concepts and Features – A complete reference for the tags included with VTScada.

...Searching and Filtering the Tag List – Find and use the controls.

...Icons That Describe Tags – Tag browser symbols.

...Tag Licensing – Tags that do or don't count towards your license limit.

Related Tasks:

...Draw From the Tag Browser – Open the Idea Studio & the Widgets palette.

...Go To Page – View the page(s) where there are widgets linked to this tag.

...Add a New Tag or Child Tag – Creating new tags.

...Adding New Menu Items – Menu and palette control.

...Move and Rename Tags – What does and does not update auto-matically.

...Copy an Existing Tag – Save work by duplicating tags.

...Plot Trends From the Tag Browser – Open the Historical Data Viewer for a selected tag.

...Delete a Tag – What to do first.

...Disable Tags – The same effect as deleting a tag, but with the option to re-enable.

...Design Your Own Tags

Searching and Filtering the Tag List

In even the smallest application, the tag list will soon become long enough to require much scrolling. For this reason, it is important to know how to filter the list so that you can find the tags you are interested in.

Note: If you are using parent-child tag structures, only the immediate child tags of the selected parent will be shown unless you check the Show Children option. Regardless of whether you have organized your tags into hierarchical structures, the \ entry is considered the grand-parent of all other tags.

The filters are shown in the following image. You may use any or all of these filters in combination.

If no tags match the filter parameters, or if you have used the address bar or tree window to browse to a tag that has not child tags, then the main browser window will display the text, "There are no tags that match the current selection".
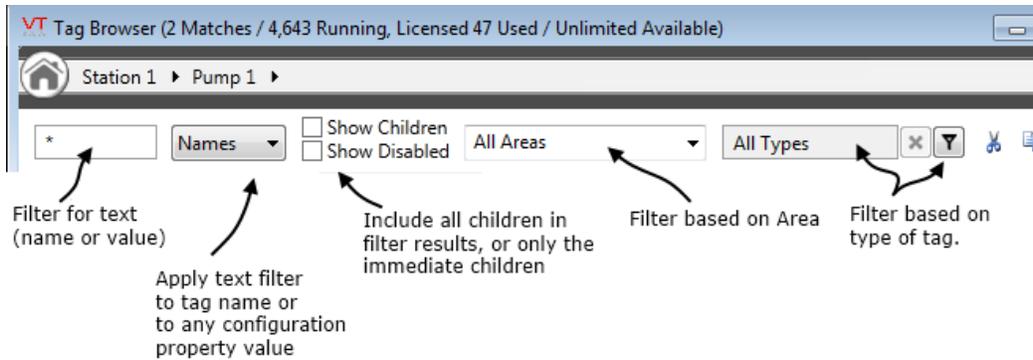
When a filter is limiting the list of tags displayed, these controls are shown against an orange background.

> Note: The filters work in combination. Be sure that each is set correctly to ensure that you locate the tags you want.

Examples of how to filter the tag display:

Filter by location in the tag structure.

When a parent tag is selected, the list is automatically filtered to show only the child tags of that parent. If Show Children is checked, all children and grand children are included, otherwise only the immediate children of the current parent are shown.

Filter by name or other property

The text filter may be applied to names, or to any field within the tag's configuration.

By default, the search field displays an asterisk (*) indicating that all tags should be displayed. A search string can consist of the full name of the tag, or parts of the name, combined with the asterisk (*) wild card character. You can use the asterisk wild card to stand for any combination of characters.

The search field is not case sensitive. Entering "A*" is the same as entering "a*".

For example:

- **A*** Return all tag names that contain an "A" as their first character (e.g. "AnalogFont" or "AlarmPriority0").

- ***A** Return all tag names that contain an "A" as their last character (e.g. "WellA" or "Soda").
- ***A*** Return all tag names that contain one or more instances of the letter "A" (e.g. "AnalogFont" or "LabelFont").
- **A** This is equivalent to *A*.
- ***** Display all tags (assuming no other filter is in effect).

If you change the scope from "Name" to "Full" | *RPM* | Full ▼ | then the search will look at all text properties including description, engineering units, and more.

Show Children

Includes all children and grand-children from the selected parent.

Show Disabled

Includes inactive tags. These are tags whose start condition evaluates to false and tags that have been explicitly disabled. (Disabling a tag sets the start condition explicitly to FALSE.)

Filter by Area

Select one area at a time to view tags that were configured with the matching area property. In the following example, "Larger Region" has been selected in the Areas drop-down list, and the tag list displays only the tags that have been configured with "Larger Region" for their Area property.
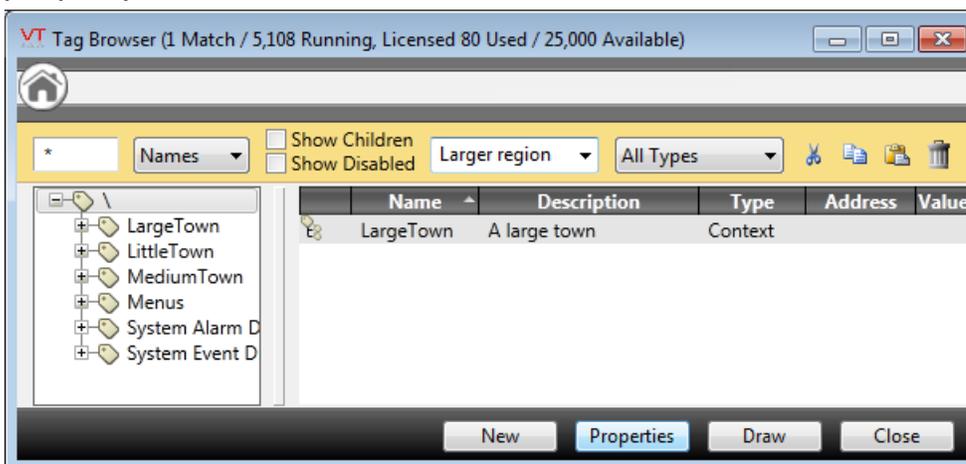
Filter by Type

Select one type at a time to view only the tags of that type. This filter also

affect the New Tag selection by pre-selecting the same type from the list. Use care if you create a tag of a type other than the one being filtered for – it will be created normally, but won't be visible in the browser until you change or remove the type filter.

When you first run a VTScada application, the Types drop-down list is set to "All," indicating that all types of tag should be displayed. You may select a type or a tag group by which to filter the tag list. Types are listed in alphabetical order in the Types drop-down list, followed by tag groups listed in alphabetical order (see Tag Groups).

In the following example, "Alarm Priority" has been selected in the Types drop-down list, and the tag list displays all the Alarm Priority tags for this application.
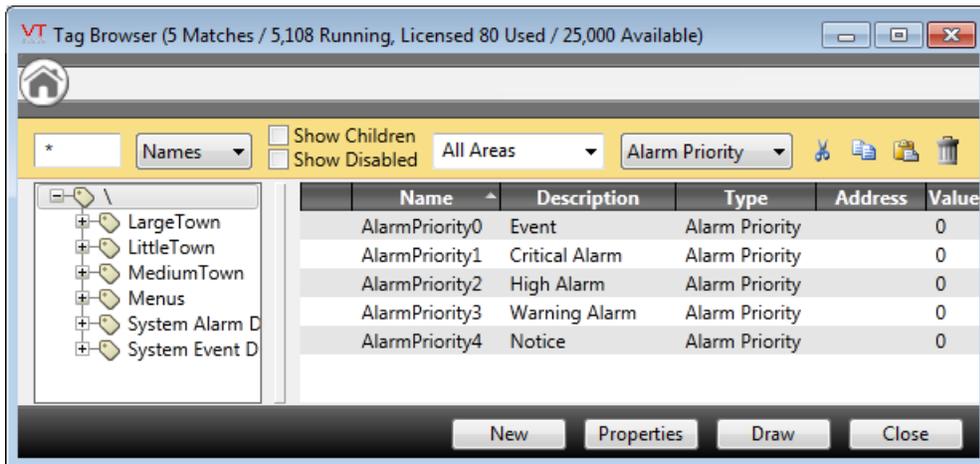
Once you've finished with a filtering option, you should reset the controls back to their default, which is to display all tags. The orange highlight warns you that when a filter is still in effect.

Related Information:

...Filtering Tags, Alarms and Realms

Tag Identification

Every tag has two forms of identification: A short, human-friendly name that you give it, and unique ID value that is automatically generated. Throughout this help guide, the word "name" will be used, rather than "short name" or "friendly name". In the places where the text refers to

the unique ID, the term "unique ID" will be used. If you want to see the unique ID value, hover the cursor over the name field in the tag browser:
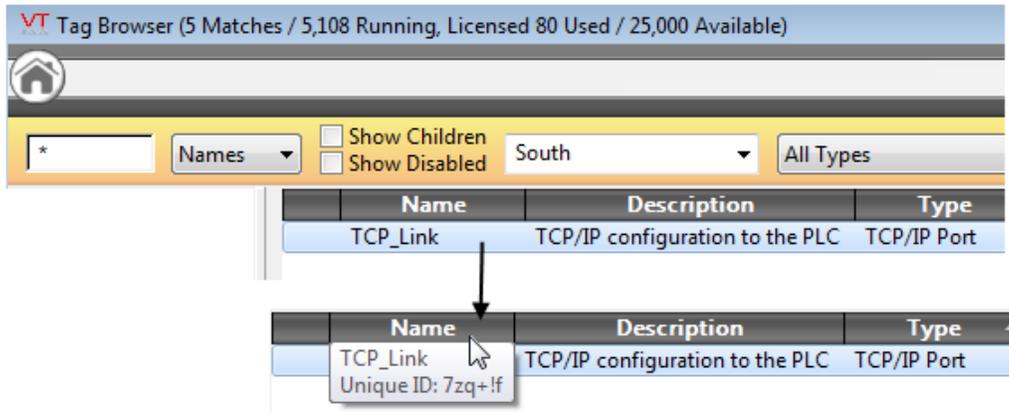
VTScada uses the tag's unique ID for logging, and for animations on a page. The name is used for references between tags and for anything in the user interface that needs to identify a tag.

A tag's unique ID value can not change, but you may change the name at any time. Since logging, alarms, page animations, and more, are all tied to the unique ID value, they are not be affected by any change in the tag's name. This includes moving a tag from one location in the tag hier-archy to another (which is, in effect, simply a change of name).

Note: Any expressions or relationships between tags will need to be updated after a change of tag name since these are based on the name parameter rather than the unique ID.

When tags are organized into a hierarchy, the name includes the parent tags. For example, if Pump1 and Pump2 both use an Analog Status tag to display the flow rate, you are free to call both of those Analog Status tags, "Flow". VTScada will know that they aren't the same tag because a) the unique ID values will be different and b) the full names will differ: "Pump1\Flow" versus "Pump2\Flow".
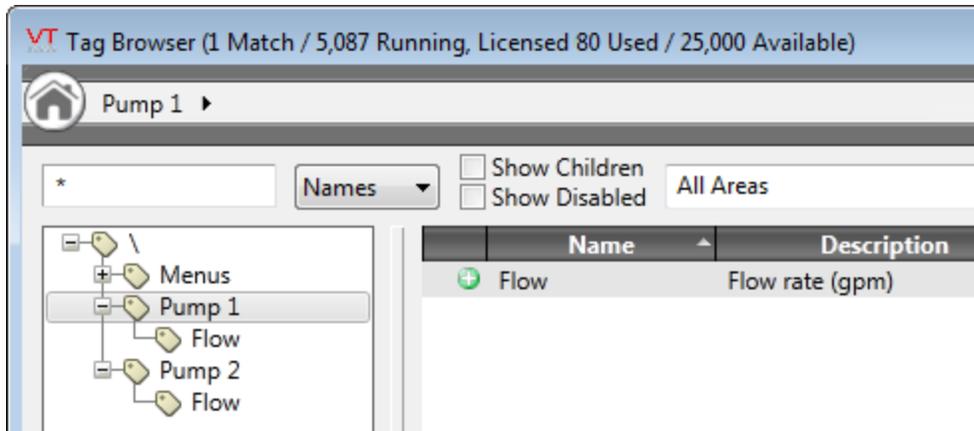
Related Information:

...Move and Rename Tags – Re-organize the tag tree.

...Access a Tag Value or Application Property – Use a Tag's Value in an Expression. See: VTScada Programmer's Guide

Icons That Describe Tags

Many of the tags shown in the browser will have symbols drawn in front of them. These tell you something about how the tags were created. You can hover over each symbol in the browser to see a tool-tip definition.

Symbol	Meaning
⯐	Has child tags. This is a parent tag.
⊕	Any tag instance that you have created directly. Not one of the built-in tags, and not generated automatically like the child tags of a MultiSmart Station.
★	An automatically-generated tag, in which you have overridden one or more of the default parameter values.

✗	Disabled. The start condition evaluates to FALSE, or you have explicitly disabled the tag. Visible only when the Show Disabled option is checked.

Related Information:

...Configuration Field Colors – Tag configuration field colors.

Add a New Tag or Child Tag

Tags are described in the opening topic of this chapter. A child tag is just a tag like any other. Its distinguishing characteristic is that it has a parent. This provides benefits of organization, clarity in naming, ease of application scaling, and security management. Tags are designed to link automatically to their nearest appropriate parent. For example, driver tags will link to the nearest port. I/O tags will link to the nearest parent driver. Alarm and Logger tags will link to the nearest parent having a numeric value.

The Select Type dialog will open every(*) time you add a new tag. The last four types that you created will be listed at the top, making it easy to create another of these types. Below those, tags are grouped according to their use. Within each group, tags that have extra features are marked. You can hover the pointer over a symbol to learn more:
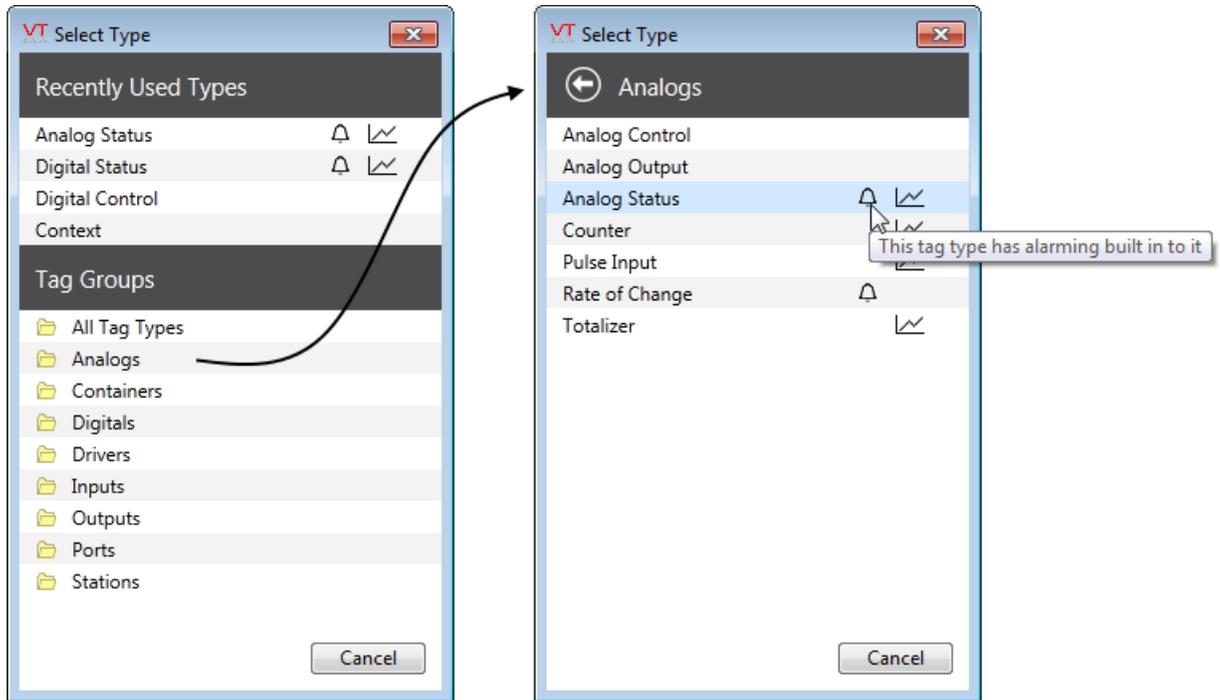
(*) If you had set the Tag Browser's type filter to a specific tag type, then the selection part is done. When you add a new tag, the properties dialog will open for a new instance of the type that matches the filter.
If the filter is set to a group, then the Select type dialog opens to that group.

To add a new tag to your application:

1. Open the Tag Browser, using either the button at the top of the page, or from within the Idea Studio.
2. Check the Address bar at the top of the Tag Browser. Ensure that you are adding your tag to the correct branch of the tag structure.
   Just as new files are added to the current operating system folder, new tags will be added to the selected parent tag.
3. Check the currently-set display filters.

Note: A filter in the tag browser won't stop you from adding a new tag, but it may interfere with your ability to see that tag in the list after it has been added. A type filter will limit the list available in the Select Type dialog. This may be helpful or it may be a hindrance.

4. Click New.

   You will be prompted to select the type of tag that you want to create. (See the notes at the beginning of this topic for more about the Select Type dialog.)

5. Open the appropriate group and select the type.

   A new tag properties folder will open.

6. Configure the tag's properties.

   Notes for every type of tag are provided in the topic, "The Tags".

7. Click OK.

The new tag will be referenced by its Name and Description in the Tag Browser's tag list. A tag's full name includes the name of all it's parent tags, separated by backslashes. This can sometimes become too long to display in a field, in which case VTScada will abbreviate the displayed name. Hover the pointer over the field in order to see the full name, displayed within a tool tip.

If the Auto Deploy option is selected for the application, then this operation is complete. Otherwise, this will be considered to be a local edit until you deploy your changes.

Next Steps:

...Move and Rename Tags – Change the name or the location in the tag tree.

...Copy an Existing Tag – Save time by making more of the same tag. Usually followed by modifying the new tag's properties.

...Delete a Tag – Instructions and cautions.

...Widget Reference – Draw the tag you just created.

Related Information:

...Best Practices for Tag Configuration – Make appropriate choices when creating new tags.

...Tag Identification – Tags have both Unique ID values and "friendly" names.

...Tag Reference – Full description of all types.

...Create a Child Tag- Part of the process of designing your own tag types.

...Import and Export Tags – Edit, add or delete tags outside VTScada

Adding New Menu Items

Menu item tags work differently than I/O and other configuration tags. They can be added only to the Menus section of the Tag Browser, and they are configured using a specialized properties dialog rather than a set of configuration panels.
To add new menu items, you can work in the Tag Browser, or you can right-click on the Page Menu page.
Depending on where you click, you will be presented with a different dialog, appropriate to that level of the Menus tree.
In the Tag Browser:

- Click at the top level, Menus, and the dialog will prompt you to create a new folder([1]). Specific widget or page menu links are not allowed at this level.
- Click at the Menus\Drawing Tools\Images level and the dialog will give you options for a folder, an image or a menu item reference. Similarly for shapes and widgets.
- Click on the Menus\Page Menu level, and the dialog will give you options for creating a folder, divider, page item or menu item reference.

On the Page Menu page:

- Right-click on a folder and you will have the option of creating a new child item within that folder.
- Right-click on a page and you will have an option of creating a new menu item at the same level in the menu structure.

Note: ([1]) If you add a new folder immediately below Menus, it will be ignored. Only the top three folders, Widgets Images and Shapes will be shown. Add new folders and menu items within one of those.

Note: You can rearrange the standard menu and palettes, organizing them however you please. But, if cutting a standard folder, note that it

will appear to have been removed as seen in a menu or palette, but will remain in the Tag Browser. This ensures that if a later VTScada upgrade adds new menu items, the standard folder structure will remain available.

Versions of the New Menu Item Properties dialog:

General Selection

(Content varies according to where in the Menus tree the new item is being added.)
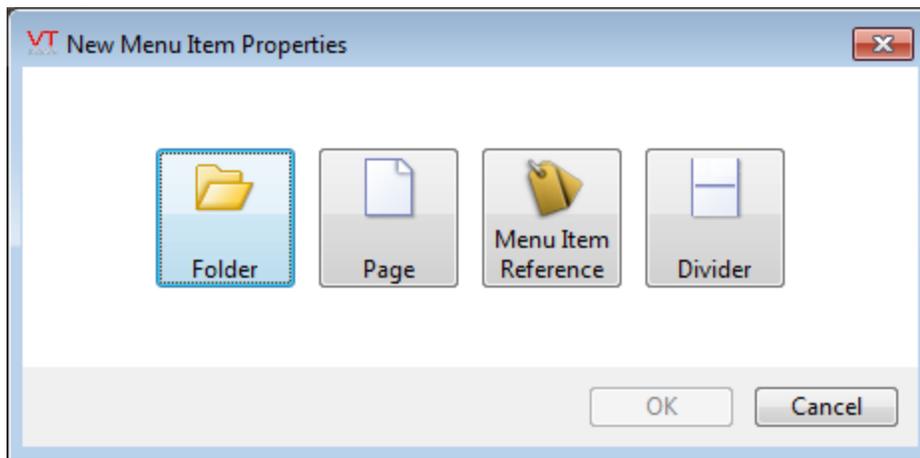
Example shows a new menu item being added at the Page Menu level. Since the content of this dialog will vary according where in the Menus tree you are adding the tag, please refer to the following descriptions for specifics.

New Folder
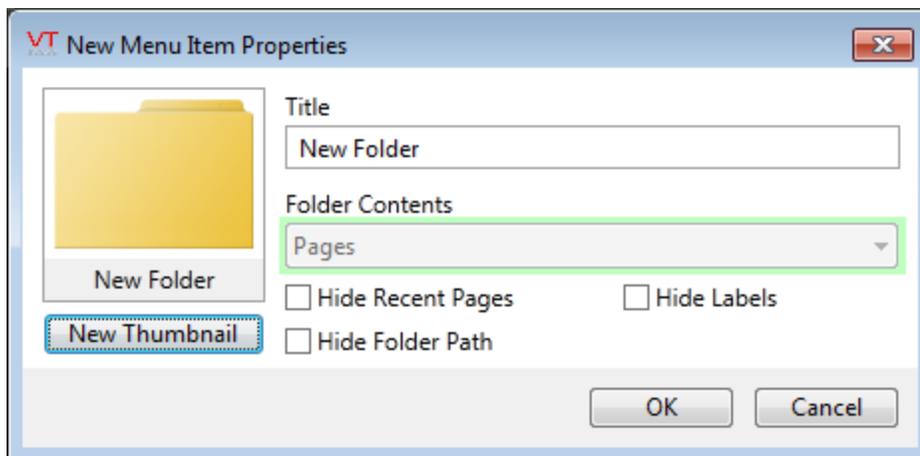
Folders exist to help organize the various menu items. They can be added at any level of the Menus structure. (See cautionary note([1]), above.)

Title: The title is the display name for the folder. You may click New Thumbnail to select any image on your VTScada images collection. The default image measures 244 x 183 pixels, therefore an alternate image will display best if it is close to those dimensions.

Folder Contents are filtered according to where in the menu tree the folder is added. The example shows "Pages", therefore this folder is being added below the Menus\Page Menu branch.

The following options exist only when a folder is being added to the Page Menu branch. They are not available in folders added to any of the palettes.

Hide Recent Pages: Recently visited pages are displayed in a list across the top of the tiled menu, making it easy for operators to reopen commonly accessed pages. By hiding the list, you make more room available on screen for the pages shown within the folder.

Hide Labels: The label is the name of the page, ordinarily shown below each page image in the folder view.

Hide Folder Path: The folder path is the navigation bar, beginning with the home icon.

Select this option when it is necessary to make the maximum possible space available to the folder contents.

Shape / New Code
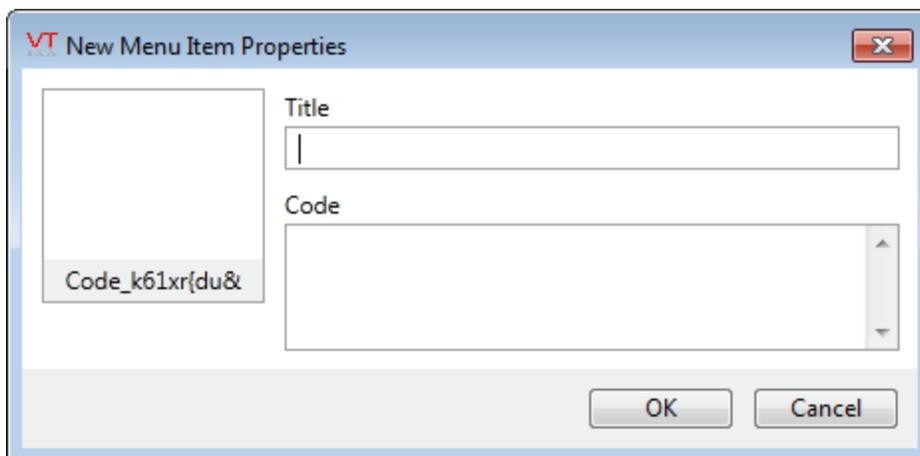
Similar to creating a new group, but potentially more powerful when used by developers who are able to work with VTScada script code.

Title: The title may be either text or an expression.

Code: To add code, open the expression editor using the editor button.

The simplest method of adding code is to draw something in the Idea Studio, then copy the image(s) and paste directly into the expression editor. Edit as required to achieve your design. References to tags may be added as required.

Shapes are based on VTScada script code. When editing a shape, refer to the function reference in this manual for details.

Menu Item Reference
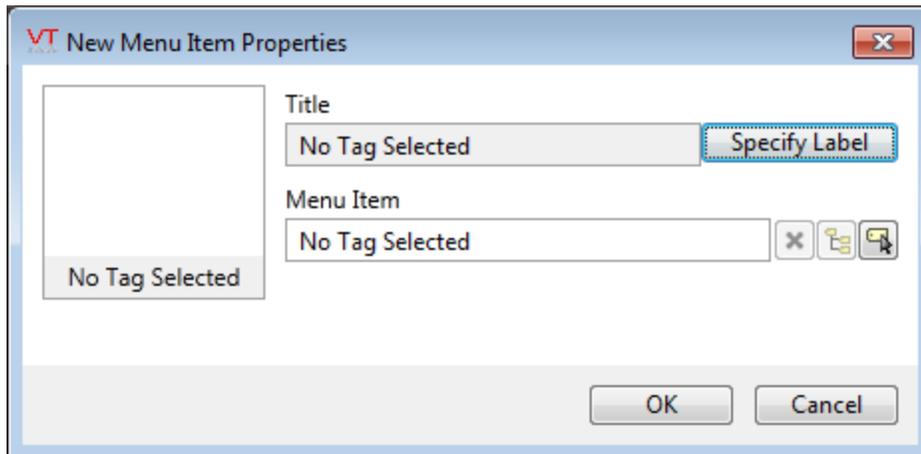
Used to create a reference to an existing menu item tag. This is used to include the same widgets, images, etc. in multiple folders. If the reference is to a parametrized page, there is no option to specify values for the parameters. The values used for the original will also be used by the reference.

The advantage of a reference over a copy is that if the original item is deleted through the Idea Studio, (but, not if through the Tag Browser), then all reference instances will also be removed, automatically.

Page Menu Item
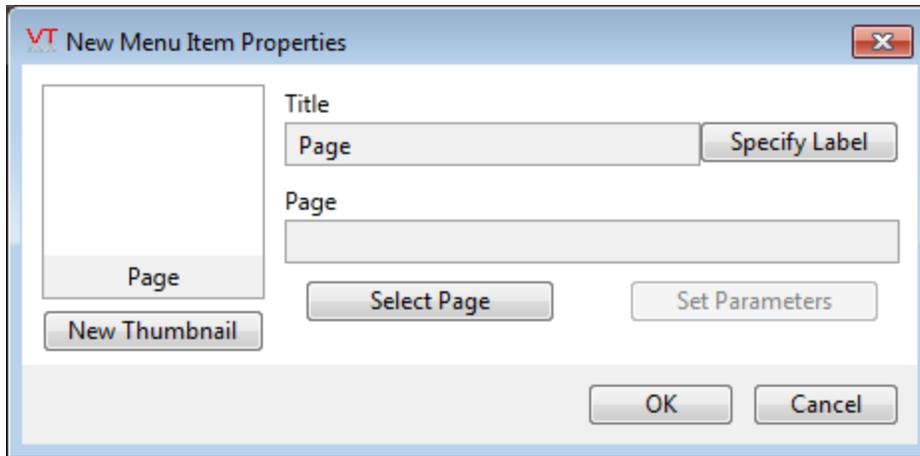
A page must already exist before it can be added using this dialog. Use the Select Page button to open a dialog, from which you can choose the page to be added to the menu.

Title versus Label: The title is the name of the page. You may wish to assign a label other than the title, in order to add clarity for the operators. If creating a series of links to the same parametrized page, each using a different set of tags or values, it is strongly recommended that you assign a meaningful name to each menu item and parameter set.

Select Page: Opens a list of all pages in the application, from which you may select the page to be linked to this menu item.

Set Parameters: The Set Parameters button will be enabled only if the selected page has parameters. If so, it is strongly recommended that you assign tags or values for the parameters in the menu link so that operators are not confronted with a dialog prompting for values.

Divider
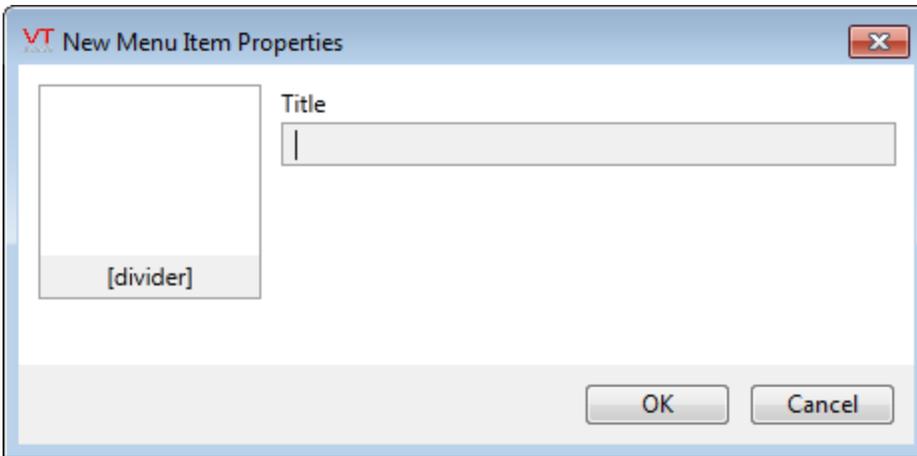
When the **title** is left as [divider], this menu item adds a horizontal line across the list menu. Has no effect in the tile menu.

Use dividers to help operators navigate a large list menu by grouping similar pages between dividers.

You may substitute any word or phrase for the title. If doing so, it is recommended that you do so in a way that makes it clear that this is a divider, not a page. Leading and trailing punctuation characters are commonly used for this purpose.

Widget
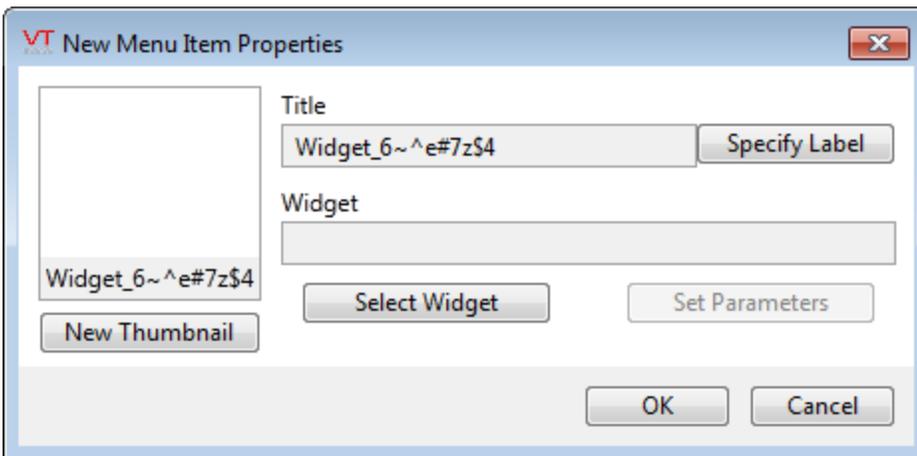
Used to organize items in the Widgets palette.

Widgets are used to represent tag values using some type of animation, or to provide operator-interface controls. An extensive collection of widgets is provided with VTScada, and you can create your own.

The widget must exist before it can be added to a new menu item tag. Use the Select Widget button to locate and select it.

A default title is provided, but it is strongly recommended that you provide your own. Click the Specify Label button to enable typing in the Title field.

Image
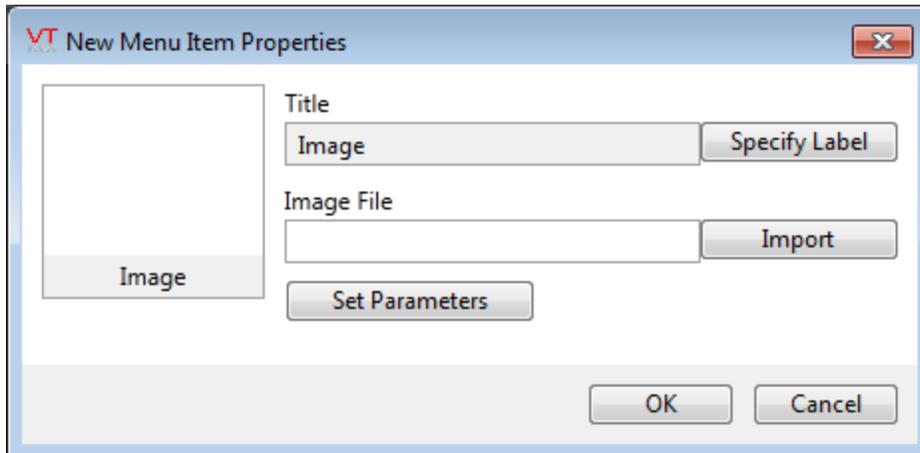
Used to organize items in the Images palette.
Used to simultaneously import new images into your application folder and add them to the Images group of the Menus tree.
In general, you should navigate to the appropriate folder of the Images sub-tree before adding a new menu item. Since well over a thousand images are provided with VTScada, it is important to keep the menu organized by type so that you can find any particular image that you need.

Related Tasks:

...Define Navigation Between Pages

The Tag Widget Reference Folder

When you click the Draw command for a tag in the Tag Browser, VTScada will check the list of Tag Widget Reference Folders for an entry matching the type of tag you are about to draw. If found, you will be given the matching list of widgets to choose from. If there is no matching reference folder, then VTScada will make a longer search for widgets that can be used with a type, and present a list of folders containing those widgets.

When there is a small set of widgets that match a type then it is more efficient for those to be grouped together in a reference folder.

For example, there is a reference folder for the Roster type, containing the widgets most commonly used when drawing that type. When drawing a Roster, you will see only those widgets.

By contrast, the Analog Status tag can be represented by many types of widgets, found in many folder groups. It is also common for developers to create their own widgets that can be linked to the Analog Status. In this case, it is better to let VTScada search out all matching widgets and present those, still grouped by folder.

If you were to create a new widget for Roster types, then you might add a Menu Item for it to the folder referenced by the Rosters entry in this list: [Widgets\Tag Type Widgets\Communications\Alarm Dialer\Roster Widgets]. Note however, that there is no requirement that you do anything at all with these reference folders.

If you have a custom tag type and have created several widgets that may be linked to that type, you can achieve the effect of having a Tag Widget Reference Folder by placing those widgets into a folder that is named after your custom type. This folder may be placed anywhere in the Widgets hierarchy.

Move and Rename Tags

Full tag names describe the tag's place in the hierarchy. Thus, moving a tag and renaming a tag are essentially the same operation. The result is that the tag's name is changed. The difference between the two is that renaming changes only the immediate (friendly) name while moving changes the full name including the parentage.

A tag's unique ID stays the same, therefore logging, alarms, reports, widgets, etc. will not be affected. Relationships that will be affected are any that are based on the friendly name, including any expressions you may have created.

Related Information:

...Examples and Cautions – What can go wrong when moving a tag, and how to avoid problems.
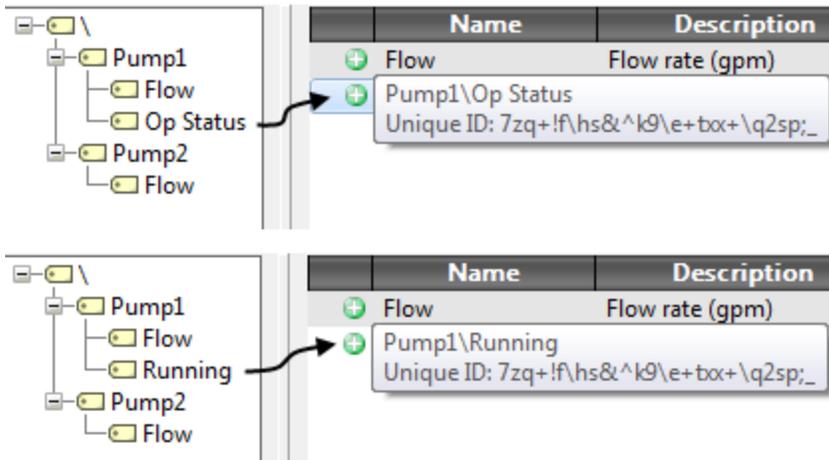
Examples:
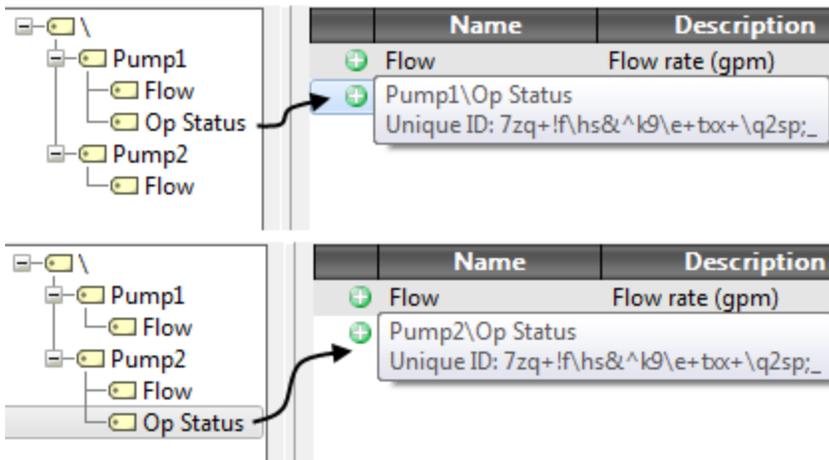
To rename a tag:

1. Select the tag in the Tag Browser
2. Click Properties, or right-click on the tag and select Properties from the context menu.
3. Change the text in the name field.
4. Click OK.

To move a tag:

1. Select the tag to be moved, in the Tag Browser
2. Right-click on the tag and select Cut, or click the Cut button.

... or ...
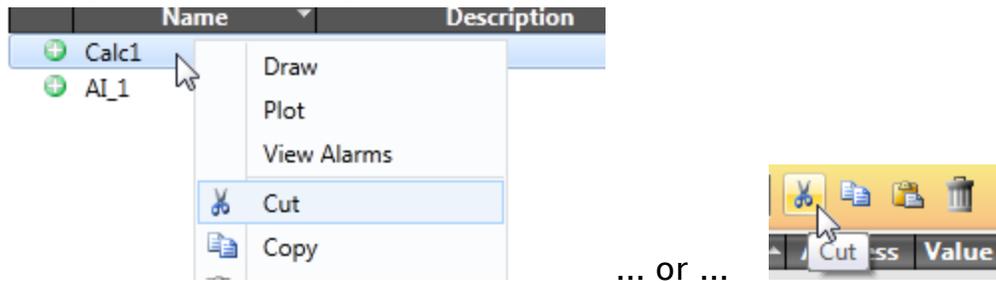
3. Right-click on a tag elsewhere in the hierarchy, under which you want to place the original tag.
4. Select Paste as Child

You will be reminded that references may need to be updated – see following notes on the effects of changing the name.
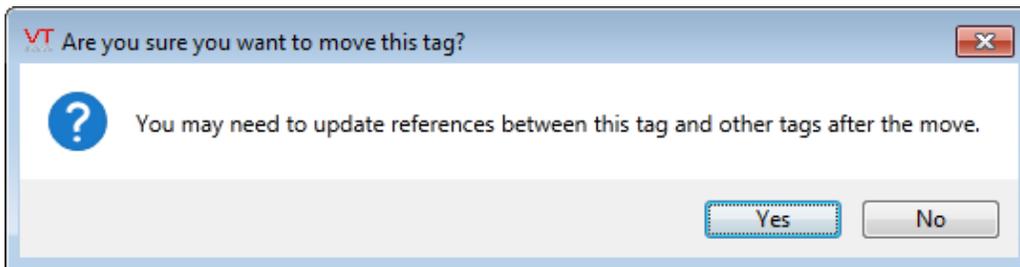
5. Click Yes to acknowledge the message and complete the move, or No to cancel the move.

Effects of changing the name:

- Expressions that referred to the tag may need to be updated.
  Expressions are created by users, not by VTScada, and use the tag's name rather than the unique ID. Changing the name of the tag will usually mean that the expression will need to be changed to use the new name. An exception to this is if both the referencing tag and the referenced tag are moved together. In this case, the reference is likely to survive, but it is wise to check.

- The link between tags may be affected depending on whether that link is Ancestor Relative Path, Open Relative Path, Fixed Depth Relative Path or Absolute Path.

- Potential for confusion
  If operators are used to seeing the tag with one name, changing its name to something else is likely to cause confusion.

Note: If you have created a new type (tag definition), the names and locations of the tags within that hierarchy are part of its definition. You may still move and rename tags, but the definition (and therefore all instances of the tag) must be updated immediately after each change. A dialog will remind you of this and ask you to confirm the action before it takes place.

Related Information:

...Relative Paths - Tag Relationships

Examples and Cautions

Rule: Logged history, tags drawn on a page, reports, built-in alarms and network values are all associated with the tag's unique ID, not the name.

Example 1:

You have an Analog Status tag named AAA, which has logged history and which is drawn on a page. You delete that tag. You then create a new Analog Status tag and name it AAA.

The new tag has a new unique ID and therefore nothing that was associated with the old tag's unique ID will be associated with the new tag's ID. This may surprise anyone who used older versions of VTScada where tags were identified only by their name.

Example 2:

You have created a new type of tag, ValveType. ValveType was created using two tags: a parent Context tag (Valve1 in this example), and a child Analog Status tag named Flow.
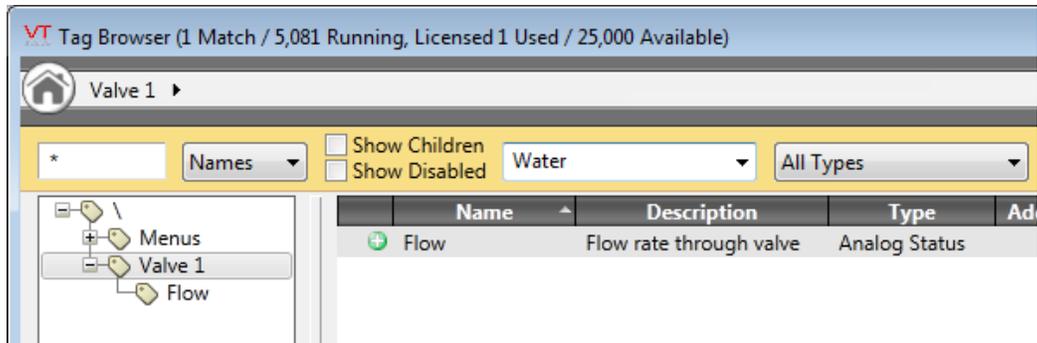
You move Flow from Valve1 into WaterPipe by doing a cut and paste operation.

When asked whether you want to redefine type ValveType, you decide upon "No".
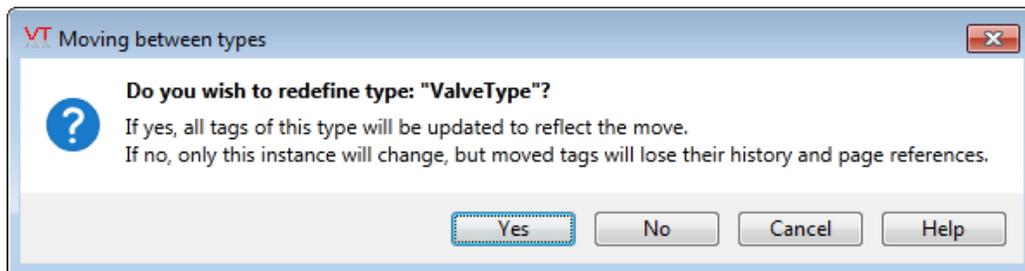
By choosing "No," you cause the operation to be a cut (delete) and paste (create new tag) rather than a move. WaterPipe gets an Analog Status tag named Flow that is identical to the original, but which is a new tag with a new unique ID. Logged history, etc from the original Flow is now unrecoverable.

Rule: When synchronizing version changes, a move trumps a modify.

Example:
You have installed the application on two, unconnected computers. On one computer, you move a tag to a new location in the tag tree. On the other computer you leave the tag where it is, but change its properties. You then connect the two computers (or take a ChangeSet from one to the other), thus allowing the version control system to synchronize the application on both machines.

In the final synchronized version, the tag will have been moved. The property changes will not be included in the running version.

Copy an Existing Tag

VTScada applications often have many similar tags. For example, ten pumps or twenty wells, all with similar I/O. For this reason, the Tag Browser includes a Copy button that can help you to create large numbers of similar tags quickly.

The steps to create a new tag based on a copy of an existing one are as follows:

1. Open the Tag Browser.

2. Use the Tag Browser's searching and filtering capabilities to locate the tag whose properties you want to copy.

3. Right-click on the tag and select "Copy" from the context menu, or select the tag and click the Copy button at the top of the Tag Browser.
    Every new tag must have a parent. This may be any other tag, or it may be the root level (\).

4. In the tag browser, locate the parent tag and right-click on it to open a menu.

5. Select "Paste as Child".
    If the parent already has a child with the same name, you will be prompted to create a new name for the copy. Do so now.

6. Open the new tag's configuration folder to adjust properties as required.

If the Auto Deploy option is selected for the application, then this operation is complete. Otherwise, this will be considered to be a local edit until you deploy your changes.

Next Steps:

...Move and Rename Tags – Change the name or the location in the tag tree.

...Delete a Tag – Instructions and cautions.

...Widget Reference – Draw the tag you just created.

Related Information:

...Searching and Filtering the Tag List – Find the tag(s) you want to copy.

...Tag Identification – Unique id and the tag name.

Delete a Tag

Tags can be deleted using the tag browser.

Note: If you delete a parent tag, you will also delete all of its child tags.

1. Open the Tag Browser.

2. Use the Tag Browser's searching and filtering capabilities to locate and select the tag you wish to delete.

3. Right-click on the tag and select the Delete option from the context menu. Alternatively, you can select the tag, then click the trash can icon at the top, right of the browser.
   A confirmation dialog will open and request confirmation that you wish to delete the selected tag. If you delete a parent tag, you will also delete all of its child tags.
   If the tag is used in any pages, you will be warned and given an opportunity to open the Go To Page dialog. Use this to review how the tag is used in any page that you have security rights to open.

4. Click OK.

If the Auto Deploy option is selected for the application, then this operation is complete. Otherwise, this will be considered to be a local edit until you deploy your changes.

Note: Warning: If you delete a tag, everything that referred to that tag's unique ID value will become inaccessible. Widgets will display the Unlinked indicator, and there will be no way to retrieve logged data.

A new tag, created with the same name as the one deleted, will have a new unique ID value and therefore no connection to anything that referred to the deleted tag.

If you later change your mind after deleting a tag, you can use the Version Log (provided that your VTScada license includes this option) in order to revert the change.

Disabling a tag has the same effect on the application as deleting it (the tag stops, related widgets are hidden, the tag count drops). The difference is that you may re-enable the tag at any time. Previously logged data can again be retrieved.

Disable Tags

You can disable any tag listed in the tag browser. The tag will stop running, and will be removed from view unless Show Disabled is checked.
Disabled I/O tags are not counted against your license.
To disable a tag, right-click on it and click to remove the check mark from the line, "Enabled".
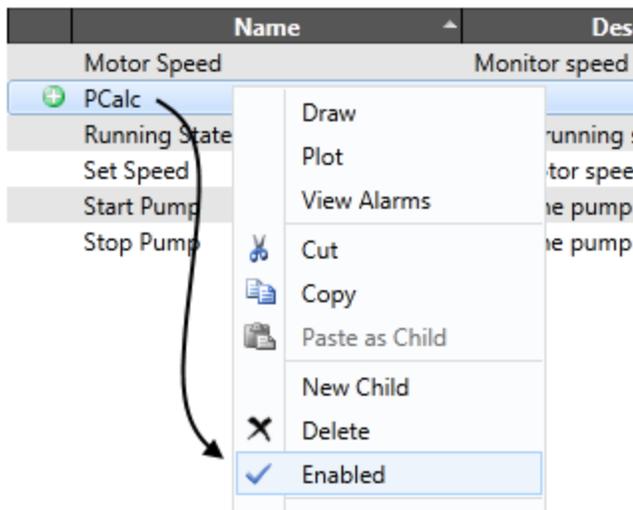
You might choose to disable a tag if you do not intend to use it for a period of time, but you do expect to use it again and therefore do not want to delete it.
You can view disabled tags by checking the Show Disabled option in the Tag Browser.
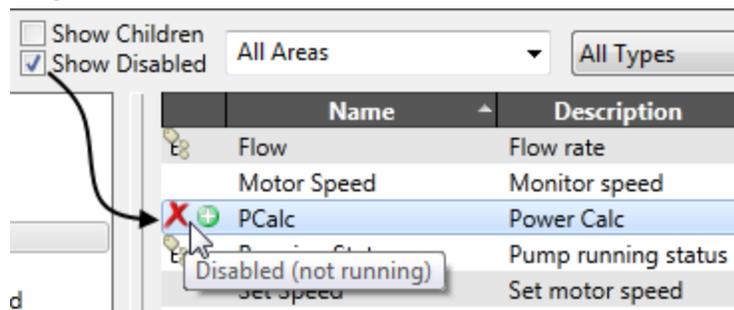
Note: Tags can be disabled automatically by use of a Start Tag Expression in the Name field. This is often used when building tag structures automatically. For example, you might create a lift station or well that can have one to four pumps. The tags for Pump 2, Pump 3 and Pump 4 would be configured so that they will be disabled in stations where they are not required.

Related Information:

...Delete a Tag

...Create a Start Condition

Go To Page

Use the Go To Page command, found in every tag's right-click context menu in the Tag Browser, to navigate to any page that contains a widget linked (directly or indirectly) to this tag. With this feature, you can quickly view a tag as drawn in the context of the application.
If the Idea Studio is open when you use this feature within the Tag Browser, then that page will be opened and made current in the Idea Studio.
Note that, for containers including Station tags, Context tags and Polling Drivers, the Site Details is always included as a destination page, even if there are no other tags within this container.

To go to a page:

1. Right-click on the tag in the Tag Browser.
2. Click, Go To Page.
   If only one page has a widget linked to the tag, that page will open immediately. If there is a choice of pages, then a dialog will ask you to choose which page to view.
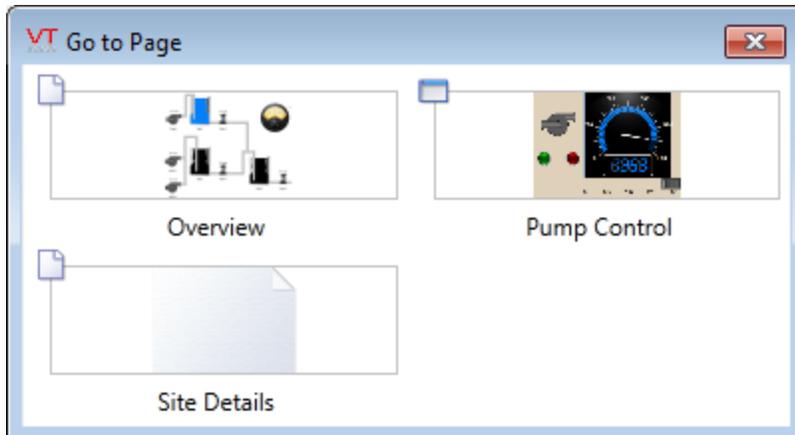
Entries in the Go To Page dialog respond to a left versus right click in the same way that entries in the menu do. That is, if the page has no restrictions on how it can be opened, a left-click will open it in the main application window and a right-click will open it as a pop-up.

If there are no pages containing widgets linked to this tag, or if all such pages are protected by security privileges that your account does not have, then you will see the following message:
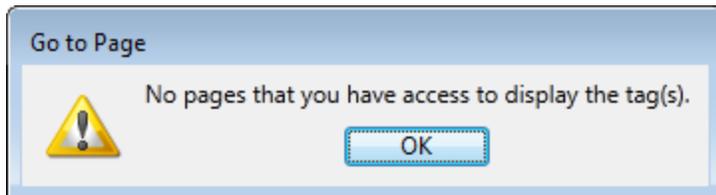
The Go To Page feature can be disabled by setting the application property, DisableGoToPage to TRUE.

Related Information:

...View the Related Tag – The same command in the Alarm Page. See: VTScada Operator's Guide.

...Plot Trends and Graphs – Using the Historical Data Viewer. See: VTScada Operator's Guide.

...See: DisableGoToPage in the VTScada Admin Guide

Plot Trends From the Tag Browser

If a tag can be viewed as a trend plot in the Historical Data Viewer (HDV), then you can open the HDV for that tag directly from the Tag Browser

list. All selected tags will be included in the plot.

To plot from the Tag Browser:

1. Select one or more tags.

2. Right-click on one of the selected tags in the list.
   A context menu opens.

3. Click, Plot.
   The HDV page opens as a pop-up window, showing the trend plot of the
   selected tags.
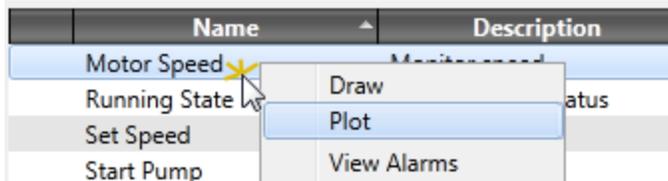
Troubleshooting:

- The plot entry is disabled.
   The selected tag cannot be plotted in the HDV.
   Your account does not have permission to view the Historical Data Viewer
   page.

Related Information:

...Plot Trends and Graphs – Operator's Guide to the Historical Data
Viewer

Tag Change History

Every change to a tag's configuration is recorded in the VTScada version
control system. This enables you to determine exactly who changed a
field, and when the change was made. You can easily spot fields that
have been changed during the current editing session, as the back-
ground color of changed fields will turn green. The background highlight
lasts only for as long as you are editing the configuration folder – each
time you open a tag's configuration folder the fields will not be green,
waiting for fresh changes.
(Certain configuration actions may cause the field to turn yellow or blue,
where each color has a meaning.)
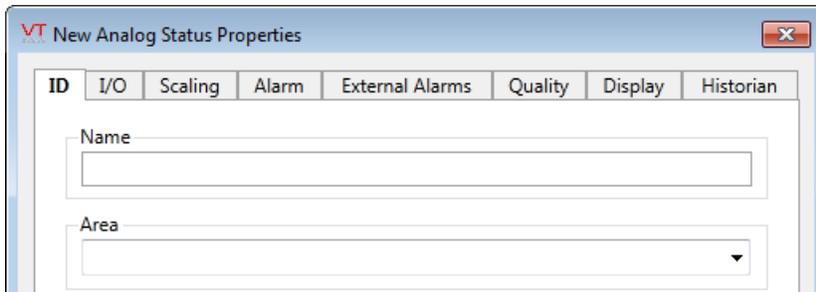
Before editing.
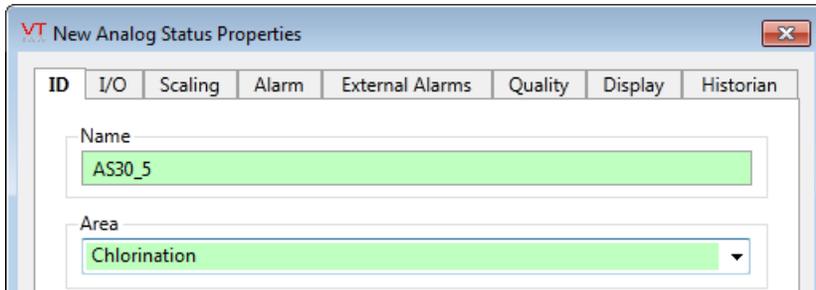
After editing.

If security has not yet been enabled, no user name will show in the history. Otherwise, the name of the user logged in at the time the change was made, will be used.
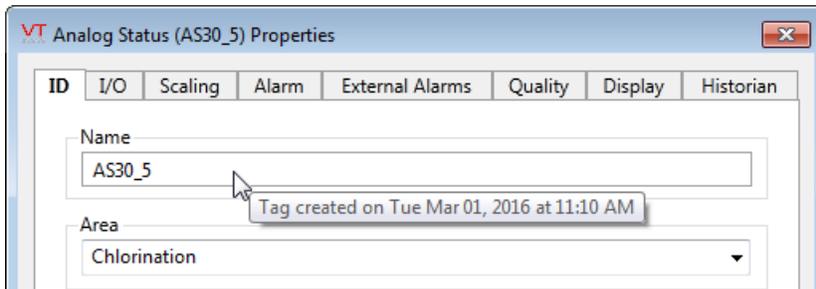
Changes made to any field will show the latest modification date and the identity of the user logged in at the time (if security is enabled).
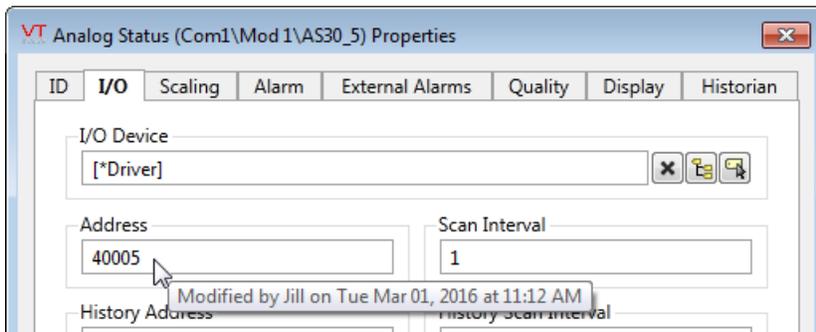
Useful hint: You can use the date of the last modification to look up other changes made at the same time in the version history. This will

allow you to find any other changes that the developer may have made at or near the same time.

> Note: Operational changes such as opening and closing valves are stored in the Alarm History, not the version history.

Changes to tag configuration that result from importing a tag file will be attributed to the user who imports that tag file, and therefore assumes responsibility for ensuring that what is being imported is correct.
The format of the displayed tooltip is controlled by application properties.

Related Information:

...Configuration Field Colors

...Import and Export Tags

...Alarm History List in the Operator's Guide

Design Your Own Tags

"Analog Status", "Digital Control", "Calculation" are all types of tag supplied with VTScada. You can easily create your own types to join this list, such as "LiftPump", "PumpingStation" etc. Each one of your new types may contain as many child tags as required to describe the matching equipment.
Each new type starts as a Context tag. Any Context tag with a valid value for the Type field can be turned into a new type of tag. All child tags of

that Context tag will be included in the new type definition template. The Type property will be used as the name of the new type. Within the application (and any applications that use this one as an OEM layer), the new type will be available alongside the standard tag types. The following images illustrate the process.

Step 3: The multi-tag structure is now just another type in the list.
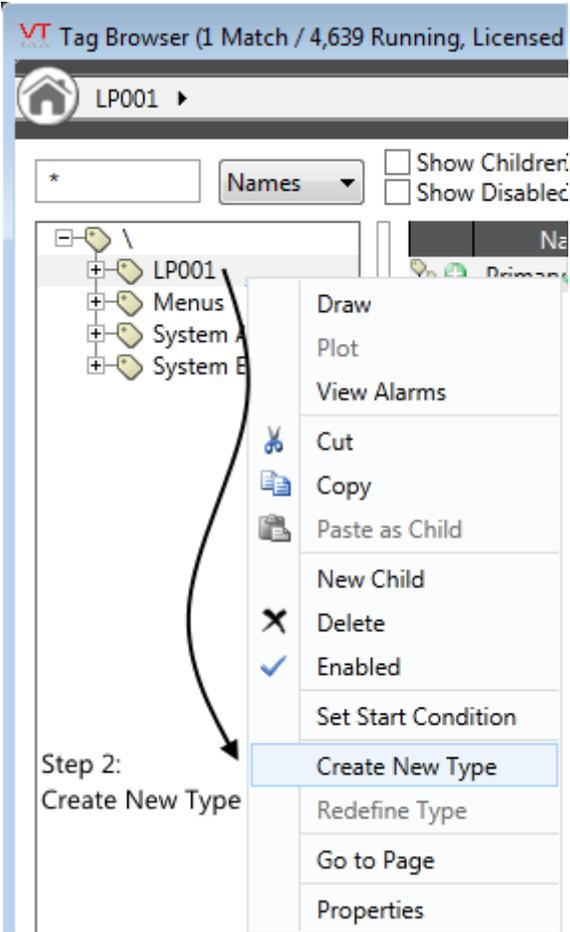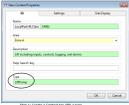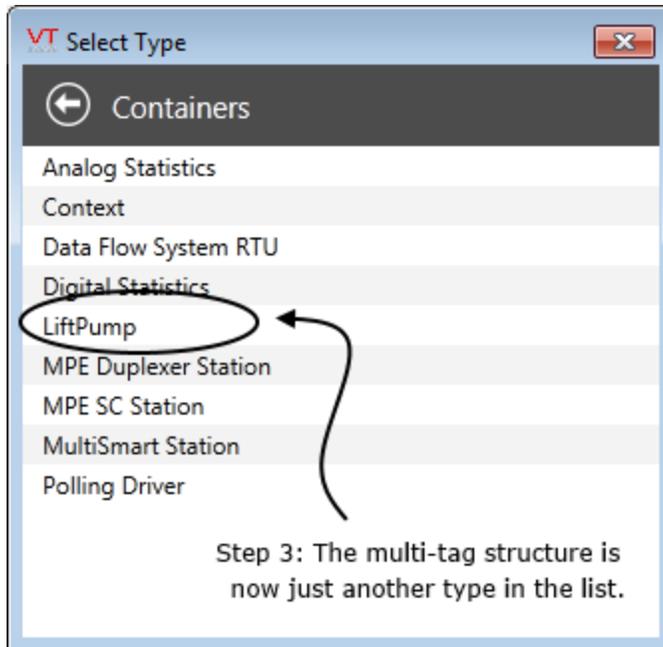
New instances of your customized type can be created as easily as any standard VTScada tag types. If the type definition includes child tags, as is normally the case, then new instances of those will be created at the same time. Note that you can further leverage this feature by creating your own widgets that display all the child tags in one object. Configuration fields in the child tags can use parameter expressions (that you create) so that they will be configured automatically using information from the parent tag.

When adding new instances of your custom tag type, developers will only configure the parent type. All child tags should be configured auto-matically using parameter expressions. (It will be possible to override If some child tags will not be needed in every instance of the type, then use Start Tag expressions. These are similar to parameter expressions, but are stored in the tag's name field and control whether the tag will be enabled or disabled. For example, in a lift station that can have up to three pumps, all three will exist in the type definition, but two will be con-trolled by Start Tag expressions and be enabled only according to the number of pumps configured in the parent station.

You can modify the design of the new type by making changes to one instance, then issuing an "Redefine Type" command on it in the Tag

Browser. For example, you might add a Context tag having two child tags: an Analog Input, and a Logger storing the Input's data. Later you decide to add an Alarm to the structure. By running a Redefine Type command, all new *and existing* instances of that type in the current application will include those child tags.

Note: The Context tag's type name must not duplicate any existing VTScada tag type or other reserved word. An error message will be displayed if the type name cannot be used.

To be safe, adopt a naming convention that adds a distinct identifier to every name (for example "MA_Type" for My Application...)

Related Tasks:

...Redefine Type – Change the structure of a custom type.

...Create New Type – Create a tag to represent equipment in your plant.

...Edit Parent Type Properties – Add or remove configuration properties in your custom type definition.

...Manage Types Using a Spreadsheet or Database – Export a type definition for editing outside VTScada.

...Delete Custom Tag Types – Type definitions that are no longer needed can be removed.

Related Information:

...Context Tags – Reference. New types start as Context tags.

...Automated Tag Configuration – Create parameter expressions and start tag expressions to automate child tag configuration.

...Relative Paths – Tag Relationships – Options to control where one tag will look for another.

...Programming Custom Tag Types – VTScada Programmer's Guide. Write code to develop new types of tag.

Create a Child Tag

The first step is to create a Context tag. After that, the steps to create the child tags are no different from the steps to create any tag.

To create a child tag:

Method 1:

1. Right-click on the tag that is to be the parent.
   A pop-up menu opens..

2. Click, "New Child"
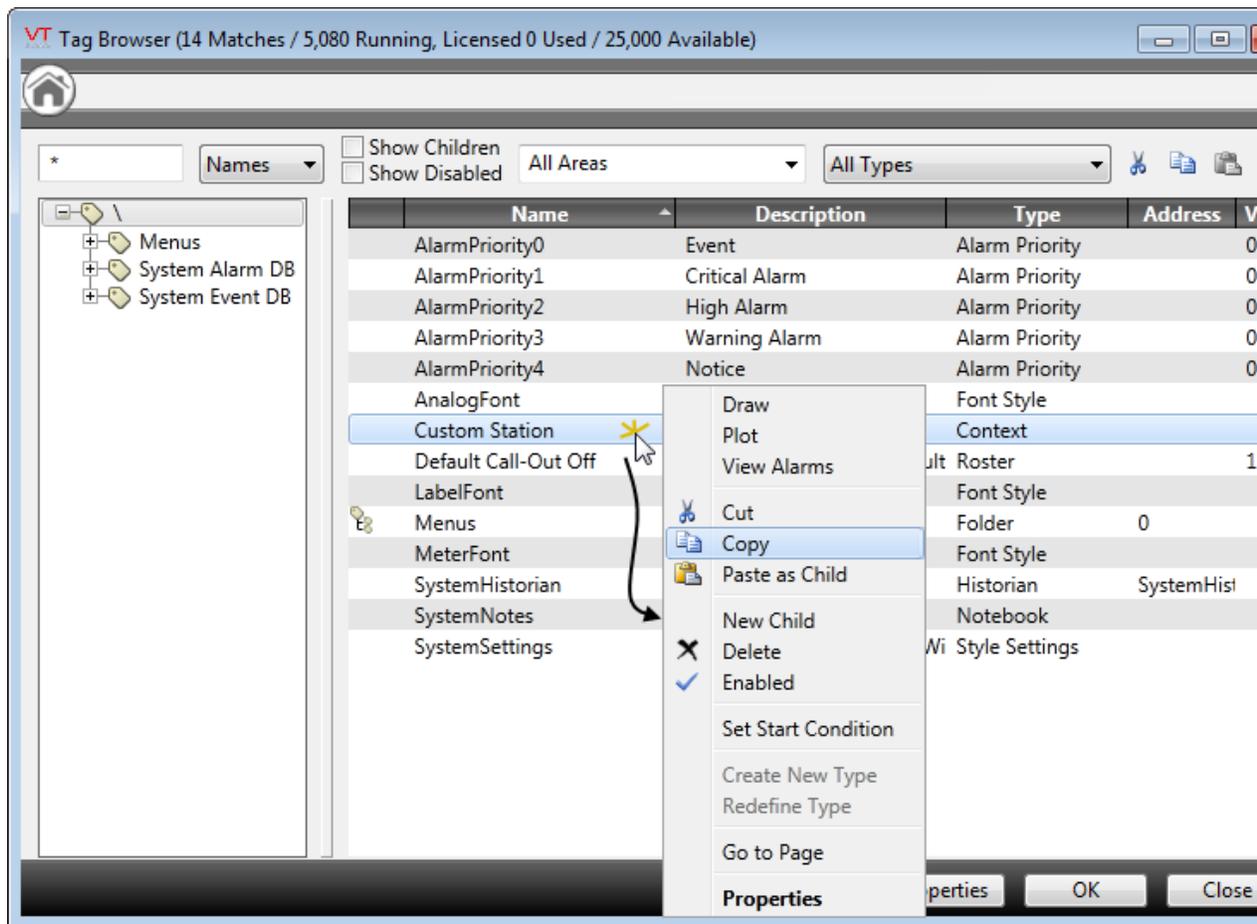
3. Choose the type of the new tag.
   The original tag will now be listed in the tree view of the Tag Browser, since it is now a parent tag.
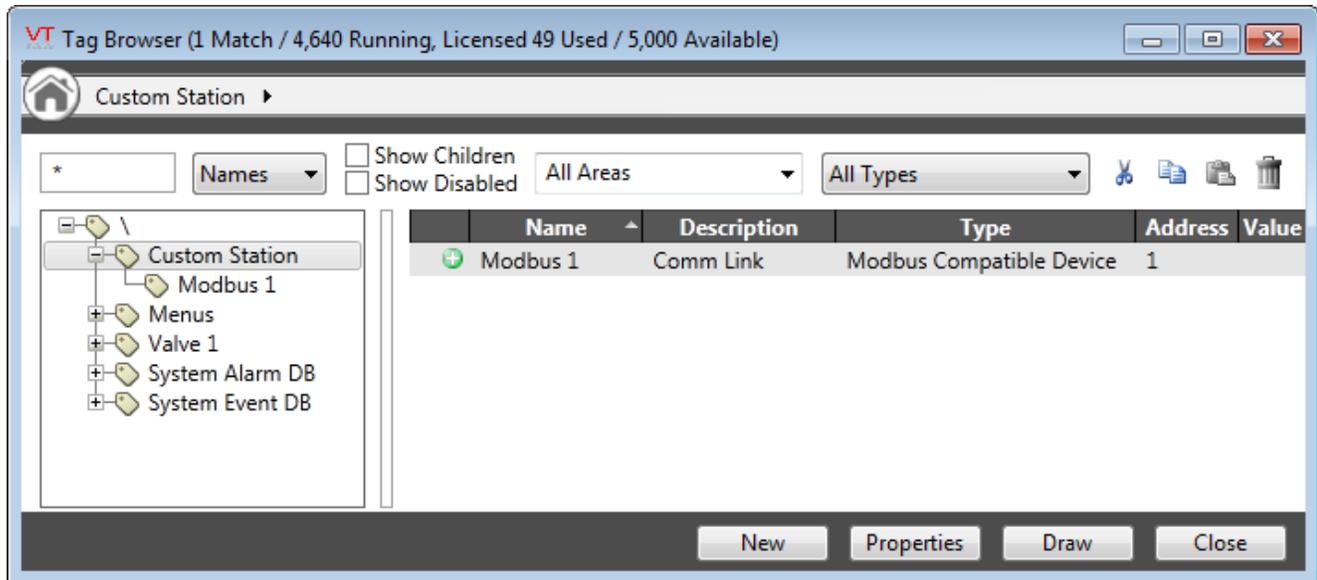
Method 2: This method can be used only if there are already parent tags in the tree view.

1. Using the left window, navigate through the existing tree of parent-child tags to the level where you want to add a new tag.
    The address bar at the top of the tag browser will show what level you are at in the structure, and can also be used to navigate.
2. Click New to add a new tag.
    The new tag will be added at the current level of the tag structure.

Method 3:

1. While configuring an Analog Input, or similar tag, add a new Alarm tag or Logger tag.
    The new Alarm or Logger tag will be created as a child of the tag being configured.
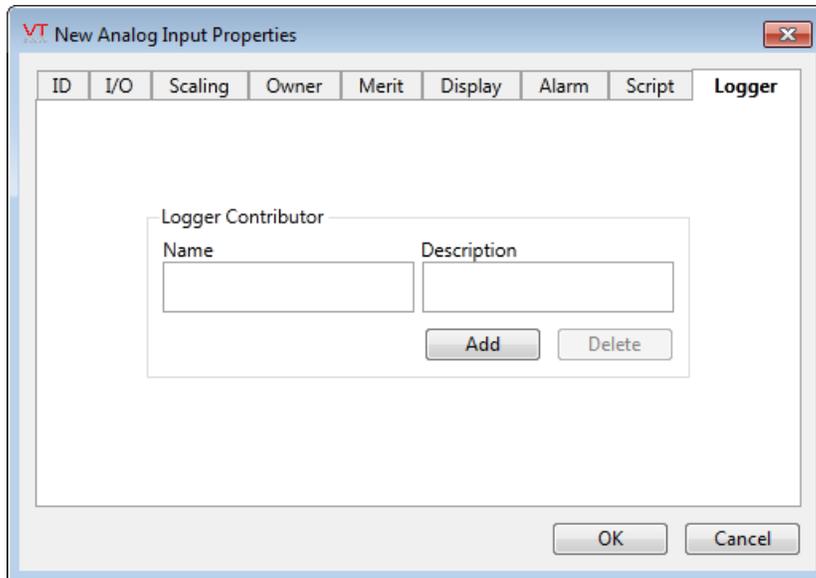
Tags are designed to link automatically to their nearest appropriate parent. For example, driver tags will link to the nearest port. I/O tags will link to the nearest parent driver. Alarm and Logger tags will link to the nearest parent having a numeric value.

A tag's full name includes the name of all it's parent tags, separated by backslashes. This can sometimes become too long to display in a field, in which case VTScada will abbreviate the displayed name. Hover the pointer over the field in order to see the full name, displayed within a tool tip.

This system provides a number of benefits:

- Organization.
  If you are looking for Pump1 in Station 1, you can find it quickly by navigating through the parent-child hierarchy.

- Clear naming.
  It might make sense for the primary pump in each well to be named "Pump1". You can do this without creating conflicts because internally, tag names include the sum of all the parents. ("Station1_drv\Well_1\Pump1", "Station1_drv\Well_2\Pump1").

- Ease of configuration.
  New driver tags will automatically find the nearest parent port. [..*Port]
  New I/O tags will automatically find the nearest parent driver. [..*Driver]
  All tags will automatically inherit their parent's Area value.

- Management of Security Rules.
  Security rules can be defined to grant privileges only in the scope of a parent tag. In this example, you might create only one application privilege, "Station Control" and protect all of a station's control tags with it. This privilege will be carried into every copy of the station. You can grant that privilege to all operators, but then limit the scope of the rule so that Joe can operate only the equipment in Station 1, while Sally operates only the equipment in Station 2.
- Application Scaling.
  When you copy a parent tag, you also create a copy of all its children. If a new station is added in this example, all the tags for that station can be created in two steps: Copy and Paste.

Troubleshooting:

- The process seemed to work, but the new tag is not shown in the Tag Browser.
  If the filter bar of the Tag Browser is orange, you may have a filter in place that excludes the new tag from view.
  Check that you are viewing child tags of the correct parent.
- The tag was created at the root level, not as a child.
  You clicked the New button while the Tag Browser was at the root level. Use Cut and Paste operations to move the tag below the correct parent.

Related Information:

...Context Tags – Reference. New types start as Context tags.

...Automated Tag Configuration – Create parameter expressions to automate child tag configuration.

Next Steps:

...Create New Type – Combine parent & child tags into a single object, describing equipment.

Relative Paths – Tag Relationships

In parent-child tag structures, it is possible that more than one tag in a given structure will have the same name (excluding the parent path). It is also possible that overall structures (e.g. Cities) may be constructed from

several sub-structures (Pumping Stations with Pumps, Wells, etc.). It therefore becomes important to have control over how one tag refers to another within the scope of the overall structure.
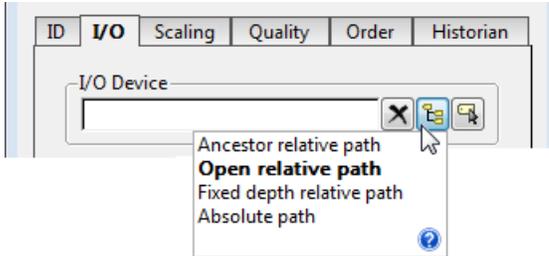
Key details:

- The full name of any tag includes the list of parent tags: "Middletown\PumpStation1\Pump1"
- The ultimate parent of every tag is VTSDB. This is seen only when specifying an absolute path, which must start at the very top level parent.
- You can use the notation ".." to mean "up one generation". Use a backslash to separate each generation and to separate a tag name from its parent or generation indicator. Thus "..\..\StationX" means "up two generations, find the tag named StationX".
- The notation used for each path type can be used in expressions.

Ancestor Relative Path

Note: this option will be disabled if the tag being created has not yet been given a name.

The type name is shown with a * prefix. The first ancestor matching this type will be selected.
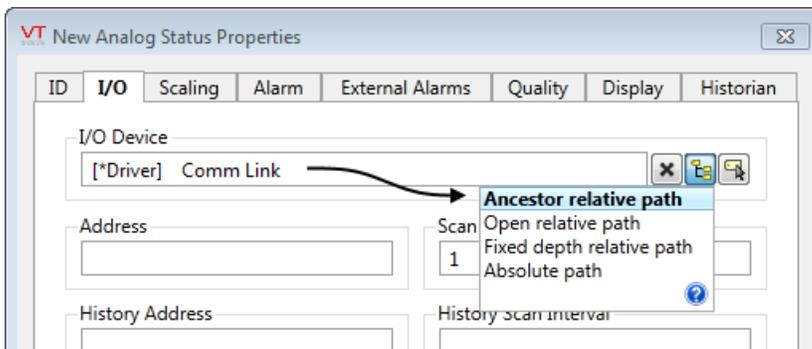
In the example shown, "[*Driver]" indicates the type of ancestor. "Comm Link" happens to be the description of the first matching tag found of type, Driver.

If this Analog Status is copied to another parent, it will look for the first parent that is a driver of any type.

Open Relative Path

All parents that are in common between the current tag and the selected tag are dropped from the path of the selected tag. If the tag is copied to a new parent structure, VTScada will search upwards through the scope of the structure until it finds a matching name.
Open relative path is always the correct choice when selecting a built-in VTScada tag such as a Font or AlarmPriority since all tags ultimately have a common parent in \VTSDB.
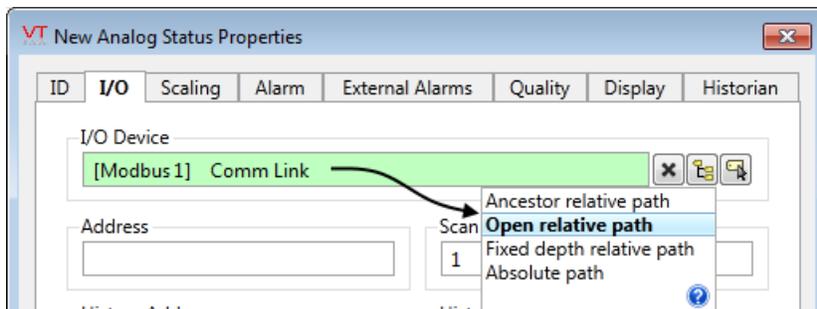
In the example shown, the I/O Device is a driver named "Modbus1" with description, "Comm Link". If this Analog Status tag is copied to another parent, it will look for the first ancestor named "Modbus1".

Fixed Depth Relative Path

Specifies an exact path through the generations where a given tag will be found. Two dots are used for each step up the tree and a backslash is used for each step back down.

Tag Browser tree, showing AI_1 is a sibling of Calc1
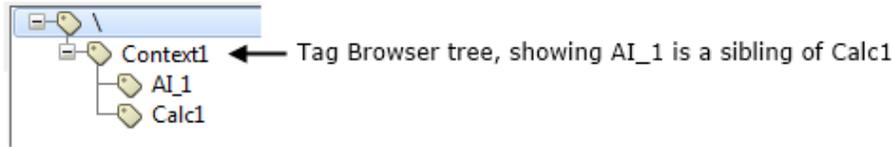
Calc1 referencing AI_1 using fixed depth relative path
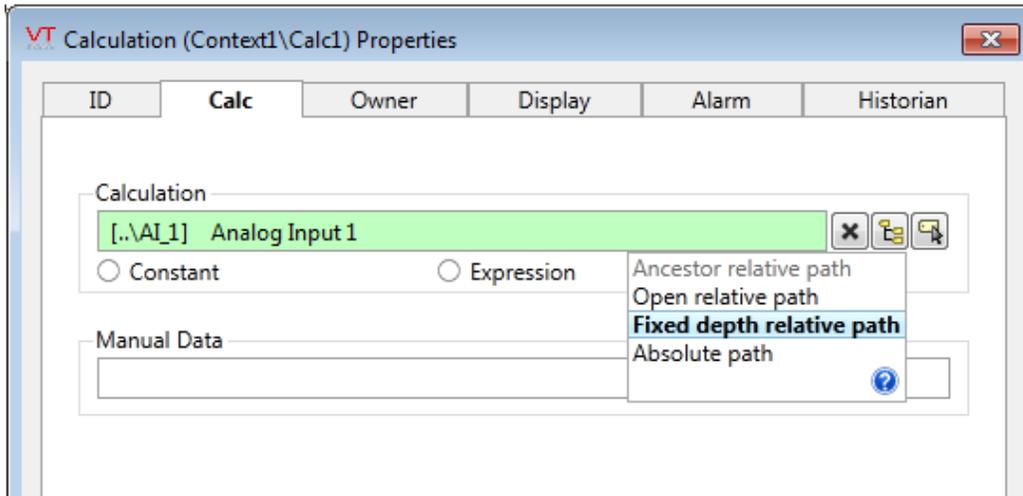
In the example shown, the calculation is referring to a sibling tag, AI_1. The fixed depth relative path in this example is ..\TagName

Absolute Path

Specifies the exact path to the selected tag, starting with the ultimate parent, VTSDB. All copies of the tag will refer to exactly the same selected tag. Internally, VTScada will use the tag's unique ID. Later changes to the selected tag's name or path will not break the link, because the unique ID will not change.
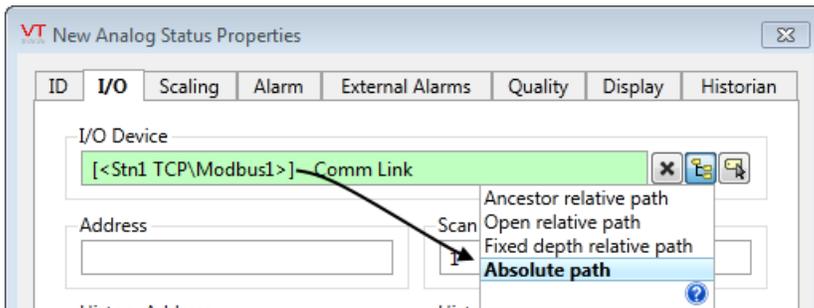
In the example shown, the Analog Status looks for a tag named "VTSDB\LocalPort\vMOD".

Create New Type

New tag types typically wrap related I/O and other tags into one new type. You can then add all the tags for an object (pumping station, chlorination unit, ski lift, or anything else), in a single step. This feature is most useful when the configuration of the various member tags can be programmed with parameter expressions, so that the child tags are configured automatically using values defined in the parent tag.
The steps to create a tag type are as follows.

1. Create a Context tag with a value in the Type field.

    -   The Type value will become the name of the new tag type.

    -   This value must be a single word and must be valid for use as tag type name. It must not match any existing tag type. If this condition is not met, an error message will be displayed.

2. In the Settings tab of the Context tag, add properties that describe the equipment this new type will represent.
    When creating instances of the new type, developers will use only these properties to configure the tag. Child tags must inherit their properties from these values.

3. Create child tags, as appropriate to describe the equipment that the new type will represent.
    Use parameter expressions for every configuration field that will vary from one instance to another of the new type.

4. Right-click on the Context tag in the Tag Browser to open its context menu.

5. Select the option, Create New Type.
    A dialog will open to confirm that you wish to proceed with this action. This same dialog provides an opportunity to control which of the built-in widgets will be associated with the new type. Some of these are not optional, and cannot be deselected. You may choose to create custom widgets at a later date for your new tag type.

5. Click, OK.

A new tag type will have been added to your application, and the selected context tag will be converted to be an instance of this type.

Troubleshooting:

- An error dialog reports that the type name is not valid.
  There must be a type name, which is a single word.
- An error dialog reports that the name exists, but you have not used that word elsewhere.
  Name conflicts with other parts of VTScada can happen. The names, "Pump" and "Calculation" for example are both taken.

Related Information:

...Context Tags – Reference. New types start as Context tags.

...Design Your Own Tags – Organize tags so that those for a similar purpose are grouped together.

...Automated Tag Configuration – Create parameter expressions to automate child tag configuration.

...Create a Start Condition – Control whether some or all child tags start in each new instance.

Next Steps:

...Redefine Type – Change and improve your new tag type.

...Design Your Own Widgets – Create a custom-designed widget to draw all components of your new type in one step.

Redefine Type

Any user-created tag type can be modified using the Redefine Type command. For example, after creating a new type, you may decide to add or remove child tags, and to change the default configuration of child tags, such as by adding a security privilege to output tags. After making these changes to one instance of the type, you may then use the Redefine Type to share those changes with all instances.
The following procedure cannot be used to edit properties in the Settings tab of the original Context tag, from which the structure was built. For that task, please refer to the topic, Edit Parent Type Properties.

To redefine a type definition:

1. Create a new instance of the tag in the tag browser.
   You could proceed using an existing instance of the type, but it is strongly recommended that you create an instance (a template) solely for the purpose of updating the type definition.

2. Modify the child tags as desired.
   - Modifications to the parent's parameters have no effect on the type definition.
   - Possible modifications include adding and deleting the child tags, and modifying default child tag configuration, such as adding or changing a parameter expression.

3. Right-click on the parent tag of the structure and select Redefine Type from the menu.

4. Confirm that you wish to proceed.
   The tag definition will be modified to include your changes, and all existing instances of that type will follow the new definition.

You can also turn any standard VTScada tag into a multi-tag structure, noting that by doing so, all new *and existing* instances of that standard type will then match the new structure. For example, you might want a Logger tag attached to all instances of Analog Input tags. Since this can have undesirable side-effects that are difficult to undo, an application property, CanRedefineOEMTypes, must be added to your Settings.Dynamic file and set to TRUE (1) before Redefine Type will be available to use on any of the VTScada types.

> **Careless use of Redefine Type on VTScada tags can have severe negative consequences. See the following troubleshooting notes.**

Troubleshooting:

- You enabled the CanRedefineOEMTypes property, then accidentally ran Redefine Type on a standard VTScada type that had child tags. Now every instance of that type has those child tags.
  This is best addressed using the Version Control system, by reversing that change.
  If your license does not include the Version Control system, follow the steps in this topic to create a new instance of the standard type, delete all children from that instance and then run Redefine Type.
  Know that you will need to re-create the child tags that were under the original instance (and any similar).
  All logged data from the original child tags will be lost and all widgets will need to be re-linked.

> **Avoid leaving the CanRedefineOEMTypes property enabled.**

Related Information:

...Conflicts When Creating or Redefining Types – What to do when names conflict.

...Design Your Own Tags – Concepts related to custom type definitions.

...Manage Types Using a Spreadsheet or Database – Change the type definition outside VTScada.

...Edit Parent Type Properties – Add or remove configuration properties in your custom type definition.

...Delete Custom Tag Types – Type definitions that are no longer needed can be removed.

Conflicts When Creating or Redefining Types

When creating or redefining a type, it is possible to create a conflict between child tag names. For example, suppose that you have created a type named LiftPump, then modify it as follows:

- Two instances of LiftPump are created: LP1 and LP2.
- In LP1, you create a special-purpose tag to measure flow. This is an Analog Input named FlowRate.
- In LP2, you do the same, but create it as an Analog Status tag, also named FlowRate.
- Realizing that every lift pump will need to measure the flow rate, you then decide to use the Redefine Type command on LP1. This will add FlowRate to the type definition.
- When VTScada attempts to update LP2's definition to add the Analog Input, it will find that there is already an Analog Status of the same name, and therefore there is a conflict to be resolved.

The problem in the preceding scenario is the work flow. When modifying the structure of a user-defined type by adding or removing child tags, you should always work in a temporary instance (template) rather than within a working instance. But from a real-life perspective, there is nothing unreasonable about the development process that led to this naming conflict. Other, more complicated scenarios involving OEM layers or Unique ID values can be imagined.

Whatever the combination of steps that led to a naming conflict in the type definition, the important thing is to resolve the conflict quickly and in a way that will not interfere with future development work. You have the following choices:

- Remove the child tag (or its name) from all instances of the type except the one that will be used for the Redefine Type operation.
  The instance that is to be redefined may exist in the OEM layer, rather than the current application.
  If widgets were linked to the tags being removed, they will need to be linked to the new child tag after the Redefine Type operation.
  If there is logged data that must be retained for those child tags, it may be possible to move or rename them rather than deleting them. This may not be an option for some communication tags.
- Create or redefine the type without the child tag.
  The tag will then need to be added, individually, to each instance of the type on an as-needed basis. Extra care will be required so that an instance is not

forgotten.

If later redefining the type definition for another purpose, the tag with the conflicting name must not be present in the instance being used for the Redefine Type operation.

Edit Parent Type Properties

After a context tag has been turned into a new type, using the Create Type command, the properties tab of its configuration dialog will change from one where you have full control over adding, changing, deleting properties to one where you can only edit values that are stored as part of the type definition.

The configuration panel of a new Context tag:
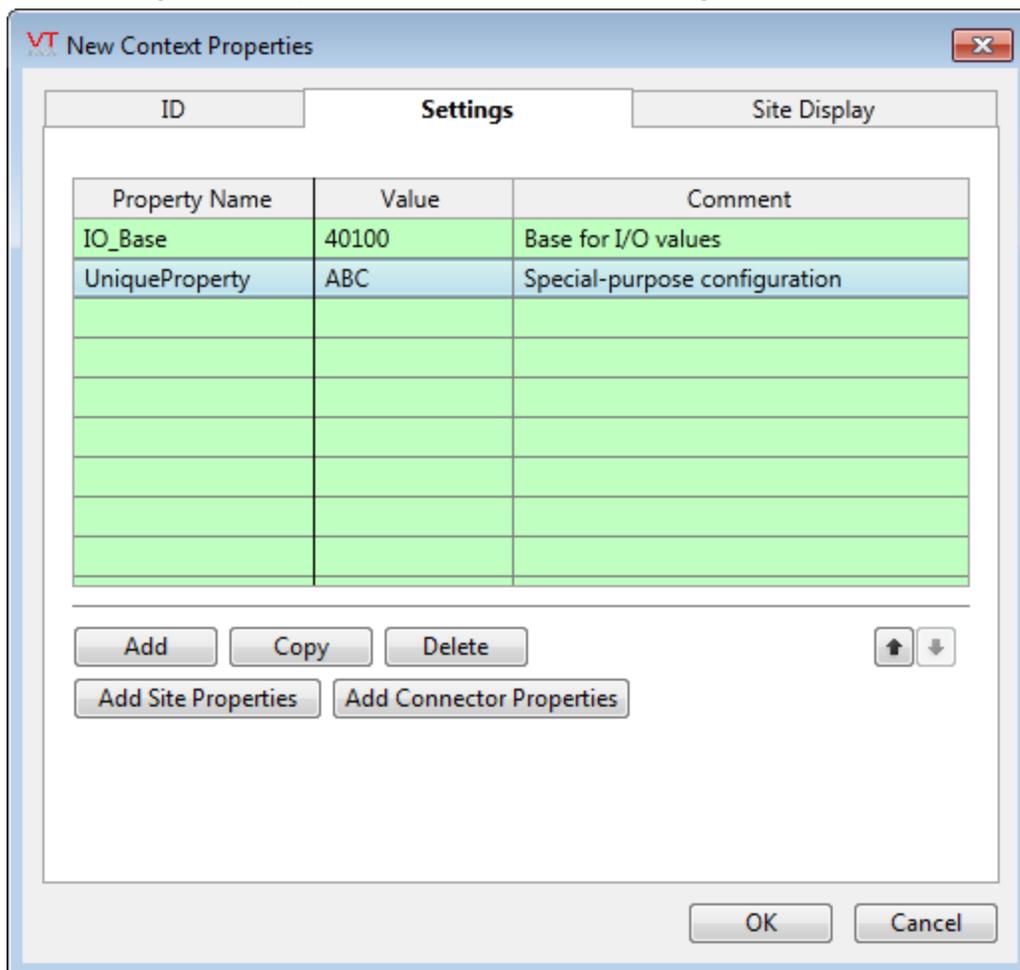
The configuration panel of the same tag, after it has been turned into a new type:

LiftPump (Local TCP Port\LP001) Properties

| ID | **Other Parameters** | Site Display |

HelpKey

Base for I/O values

40100

Special-purpose configuration

ABC

OK   Cancel
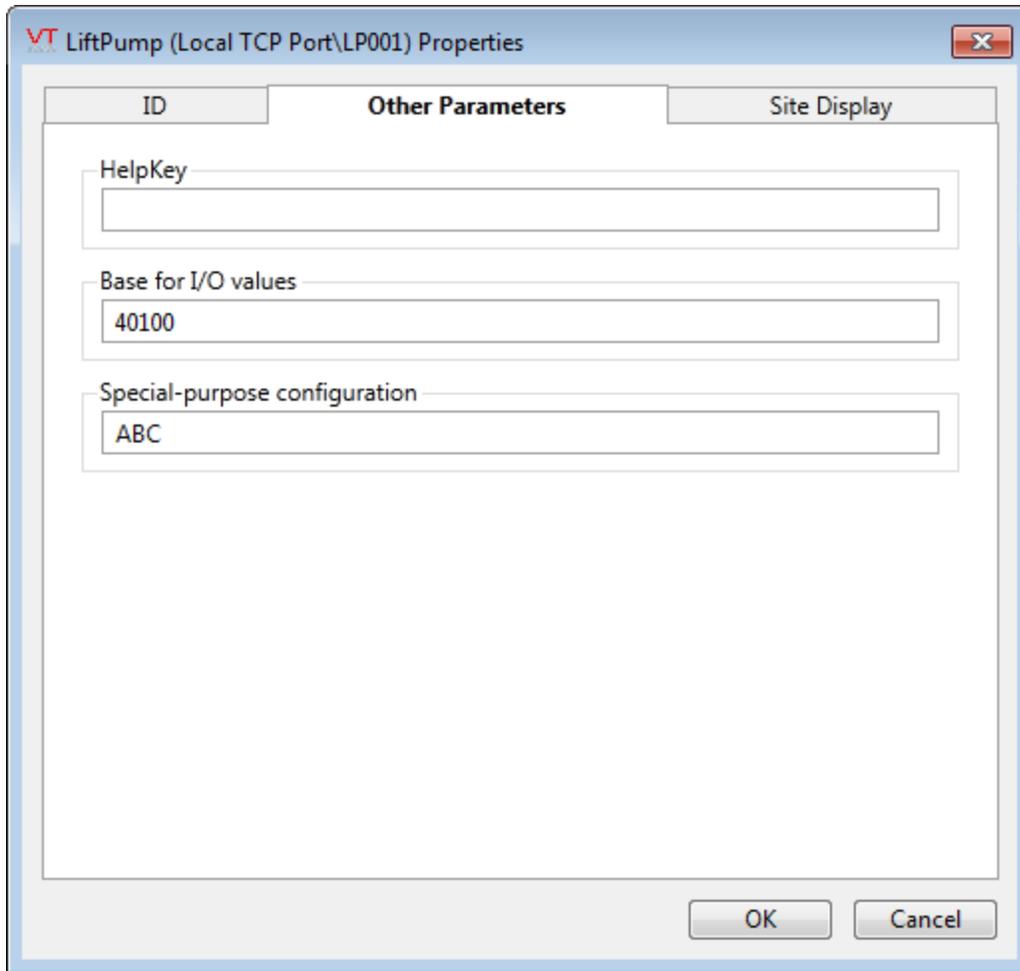
You can edit the parameter list for the type (and therefore, of all instances of that type) by using the Edit Type Properties page from the Application Configuration dialog. This will change the type's parameter definitions, rather than with values stored in those parameters for a single instance of the type.

Note: if security is enabled, the privilege Manage Types must be granted before you will be able to proceed. This privilege is not granted to any role by default.

1. Access this page as follows:
    a. Open the Application Configuration dialog.
    b. Click, Manage Types.
2. Find and select the type in the table.
3. Click, Edit.

Edit the properties of "LiftPump"

You may add, delete and rearrange parameters. Parameter labels are used in the Type's Config Folder.

Parameter Name	Parameter Label
Name	Name
Area	Area
Description	Description
HelpKey	HelpKey
IO_Base	Base for I/O values
UniqueProperty	Special-purpose configuration

Sidebar menu:
- Edit Properties
- Edit Server Lists
- Edit Security
- Create ChangeSet File
- Apply ChangeSet File
- Export/Sync Tags
- Manage Types
- Import/Export Files
- Maintain File Manifest
- Deploy Changes
- Revert Changes
- Show Version Log
- Information

Buttons: Add | Delete | Add Site Parameters | ⬆ ⬇ | Back | Apply

Note: It is not possible to edit the parameter definition of an OEM or VTScada type. Parameter definitions of OEM types can be edited only in the OEM application.

Related Information:

...Design Your Own Tags – Concepts related to custom type definitions.

...Redefine Type – Change and improve your new tag type.

...Manage Types Using a Spreadsheet or Database – Change the type definition outside VTScada.

...Context Tags – Reference. New types start as Context tags.

Manage Types Using a Spreadsheet or Database

If you have created a new type of tag, you can export the type definition to a spreadsheet or database file. Some developers find it easier to work with a type definition in a spreadsheet format. After modifying the tag structure in the exported file, you can import those changes to update the type definition within the application.

Note: Your account must have the security privilege, Manage Tag Types, before you can work with this dialog.

Type definitions may be built-up across OEM layers. Only what was added, changed or overridden in the current layer will be exported. For OEM types, the descendents list will count only the child tags defined in the local application, not in the OEM application.

Options:

Show OEM Types	Enabled only if the application property, CanRedefineOEMType has been set to TRUE. Use extreme caution.
Remove	Allowed only for user-defined types, and only if there are no instances of that type in the tag browser.
Edit	Applies if the original parent was a Context tag before the type was created. You can use this to add, remove and edit the properties of that tag, using a dialog similar to the Settings tab of a Context tag's configuration panel.
Sync	Requires an existing spreadsheet or database, created using the Export button. Synchronize will import any changes that were made in that file, modifying the type definition within the application. What it does next depends on your choice of Import Clean Up Options. If Update is selected, a fresh copy of the structure will be sent to the external file, including an updated version number. If Delete is selected, the external file will be removed.
Export	Sends the structure of the selected tag to a file of the selected type, Access, Excel or other ODBC.

If there is no check mark in the Removable column of any of the tags (preceding image), then there is at least one instance of each in the application. (Standard VTScada types cannot be removed, regardless of whether there are any instances.) Note that the Remove button and the Edit button can only be enabled while the application is running.

The check mark in the Editable column indicates that this type was based on a context tag. You can click the Edit button at the lower left of the display to work with that tag's parameters. All instances will update to have the new or changed parameters when you finish. See Edit Type Properties.

If you intend to work with the tags in a spreadsheet or database, start by clicking the Export button.

An operating system dialog will open, to prompt you for a file name. The suggested name for the export file will match the type name, but this is not required. The VTS_Reserved worksheet or table in the file contains the name of the type that will be modified when the file is re-imported.

> **Note:** Each type must be exported to, or imported from its own file. You cannot put two type definitions into one file.

All the child tags of the current type are exported, no matter how many generations deep in the structure. But, if any of those child tags are themselves user-defined types, then their structure will not be included, since there can be only one type definition per export. The exception to this rule is that tags with values that have been overridden will always be exported.

Each entry will be named according to its place in the tag structure. In this example, SpecialPurposeTag has the following structure (shown with an instance named "MyStation1").
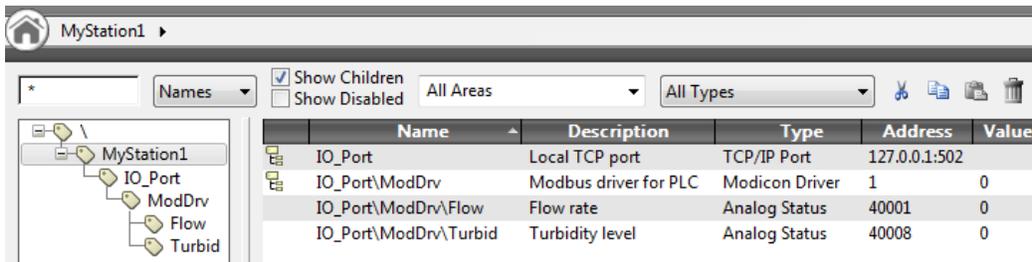
In the export file, child tags are named relative to the parent. So, in this example, the TCP/IP Port worksheet will have the following entry for the TCP/IP Port child named IO_Port. Since this is the first child in the structure, its name does not include the top-level parent.

The Modbus Compatible Device tag is a child of the TCP/IP port. It's worksheet name will reflect its place in the structure:

Note the doubled backslashes. A single backslash indicates the division between parent and child tag names, but the spreadsheet program forces us to use the doubled backslash. See also: Relative Tag References & Parameter Expressions.

And, so on for the two Analog Input types that are children of the Mod-
bus Compatible Device in this structure.

Note the use of relative tag paths (*Driver) for the each I/O tag's driver
field.

You can add new child tags of any type to the structure, by creating
entries in the appropriate worksheet or table, being careful to follow the
naming example shown here. You can also remove or modify any child
tag's definition in the structure.

> Note: Warning: Column A must be left blank for new tags. Do not
> change or remove this field for existing tags.

Upon re-importing the file by using the Synchronize button, the file, the
structure within the application will be modified with your changes.
Version control applies to exporting and importing type definition files.
The exported file contains a key that identifies the application's current
version number at the time of the export. VTScada is able to compare the
contents of the file being imported to the definition of the type as it exis-
ted at that revision. Only differences between the file's current contents
and the type definition as it existed at the time of the export will be
merged.

Copy Types to Other Applications

After creating a custom tag type, you may decide to copy the type defin-
ition to another application. A common scenario is a Systems Integrator
creating a type for a specific project, then deciding to add that type to an
OEM layer so that it can be reused in many projects. The process
provided here will copy the definition template for the type, but no
instances of the tag.

The example images within the steps show the custom tag, HoldingTank,
being copied from the Completed Tutorial application to a new applic-
ation.

To copy a custom tag type definition from application A to application B:

1. Find the source file that defines the tag.

   This will be a .SRC file with the same name as the tag, stored in the application's main folder.
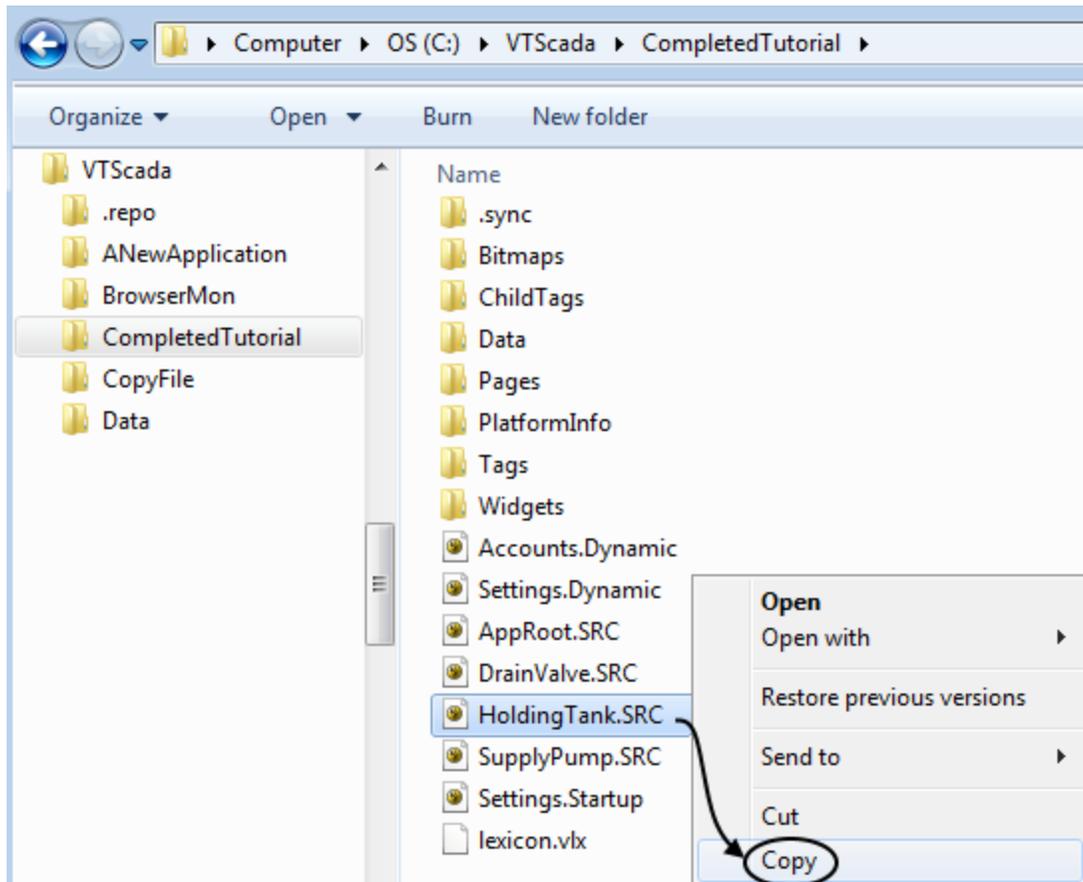
2. Copy that file to the main folder of application B.

3. Ensure that application B's folder structure includes a folder named "ChildTags".

   If not, create that folder now. Note that "ChildTags" is one word.
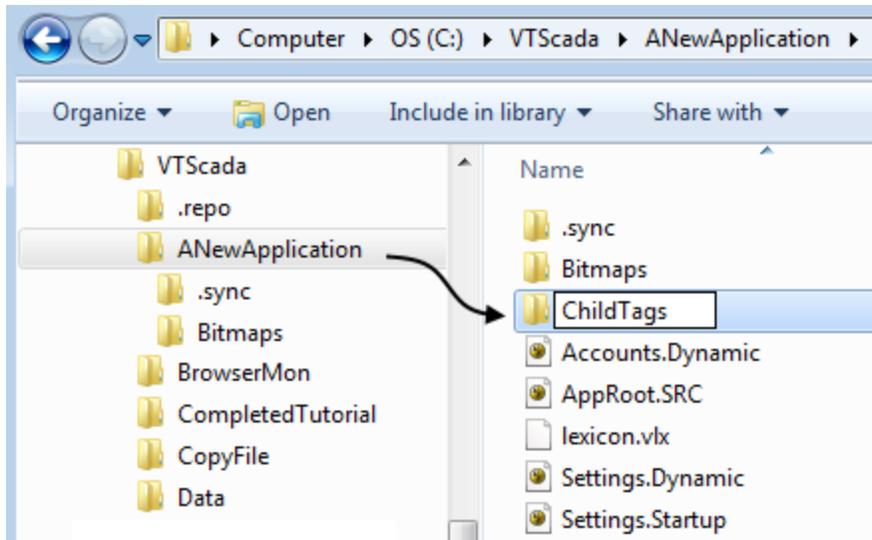
4. Open the ChildTags folder of application A.

5. Find the sub-folder that has the same name as the tag type.
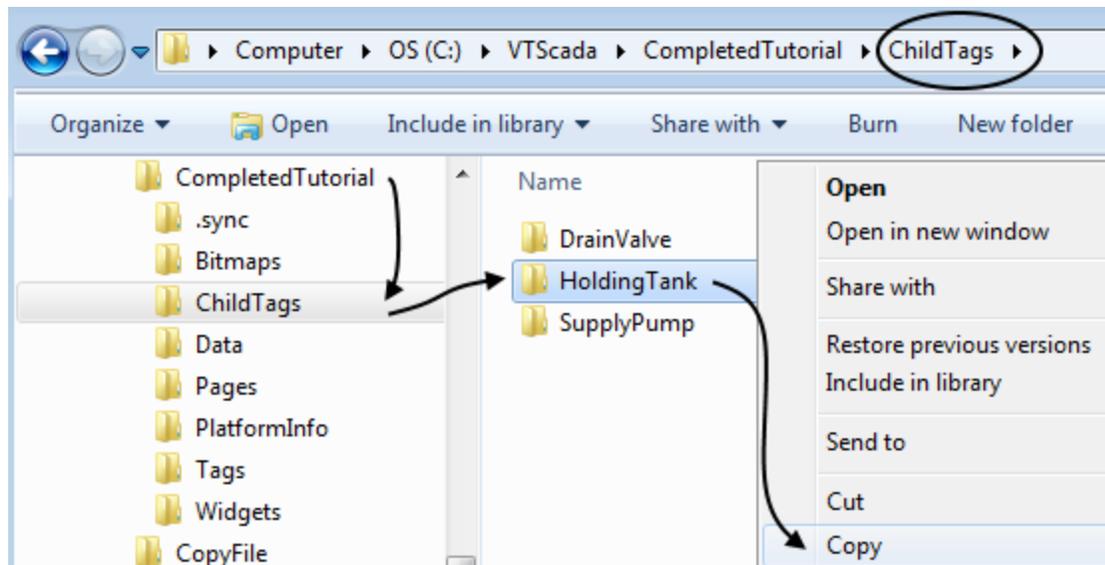
6. Copy that entire sub-folder to the ChildTags folder of application B.

7. Return to the main folder of application A.

8. Using an ASCII text editor that does not add formatting, open the file AppRoot.SRC.

9. Scroll to the section labeled [(POINTS).

10. Copy the line that declares the tag type.
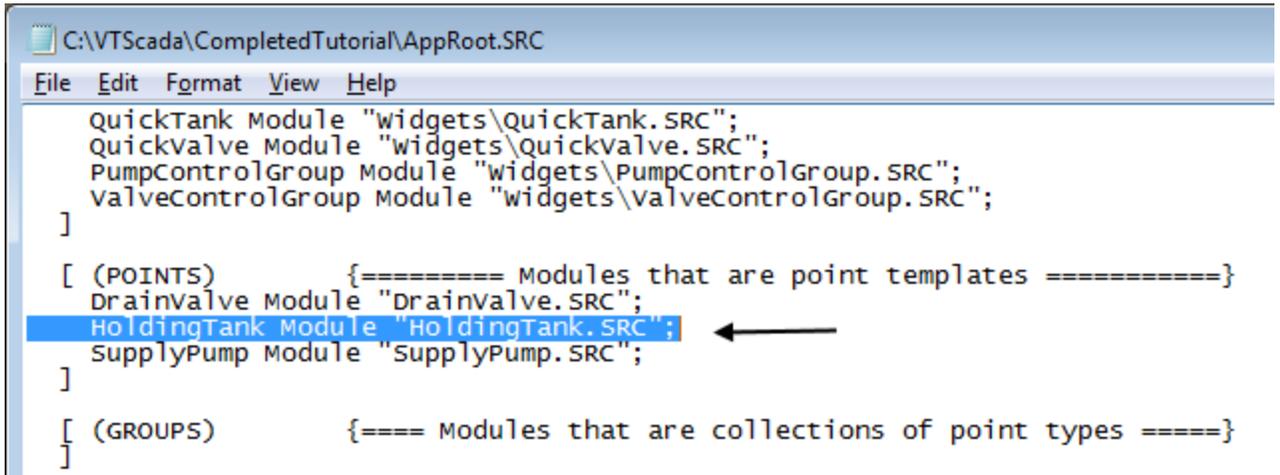
11. Open the AppRoot.SRC file of application B.

12. Paste the line into the [(POINTS) section of that file.
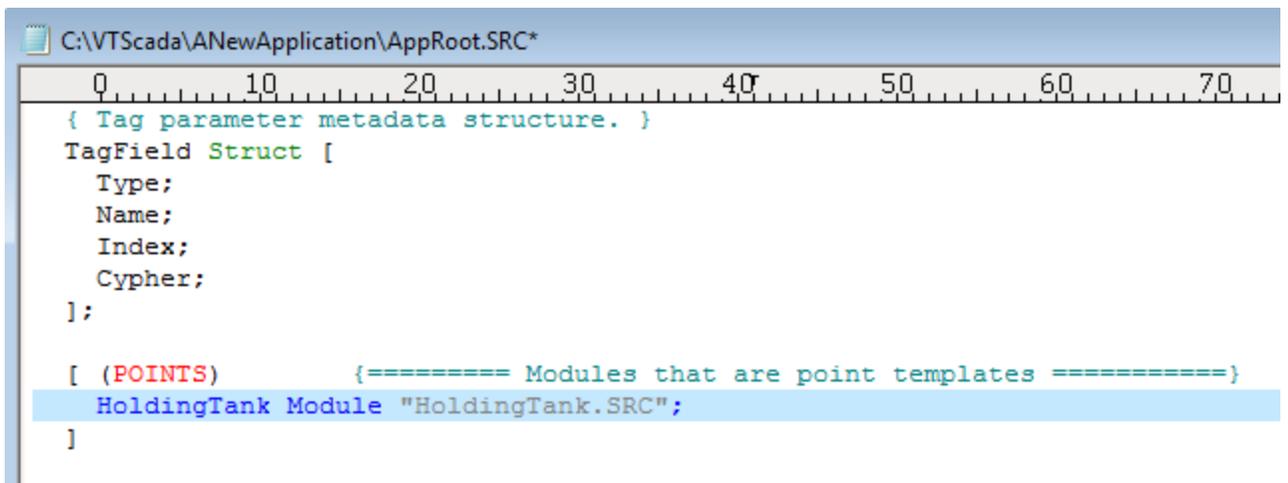
13. Save your changes.

14. In the VAM, click the Import File Changes button for application B, then fol-
low the steps to import the files and recompile. (The number of files to be
imported will vary.)
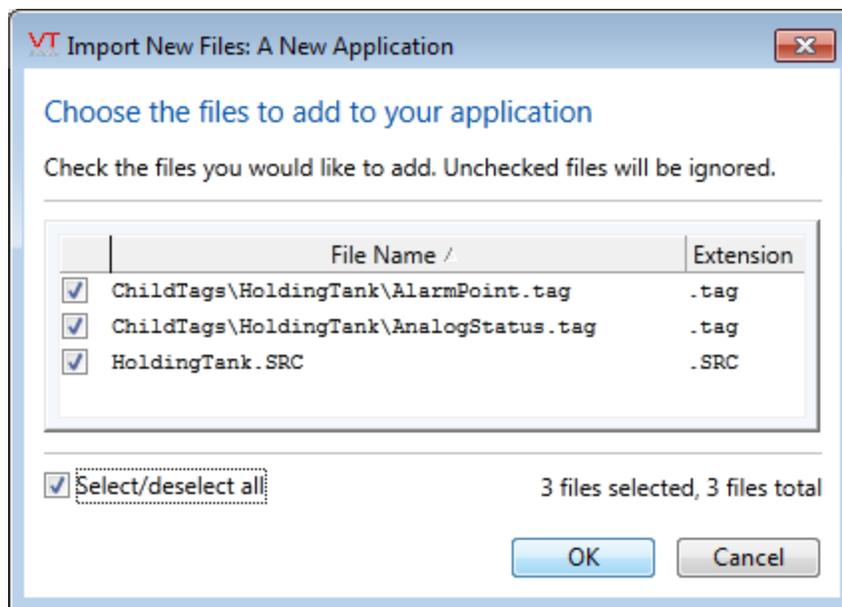
Delete Custom Tag Types

Only type definitions that you have created can be deleted. You cannot edit or delete a standard VTScada type.

Note: If you delete a child tag from a custom type structure, it will be flagged as disabled until you run Redefine Type on the parent. It cannot be fully deleted so long as it exists as part of the type structure.

Prerequisites:

- There must be no instances of that type in the application.
- If the application is secured, you must possess sufficient configuration priv-
  ileges.
- You must be working in the application where that tag was created – you can-
  not edit the definition of a type in an OEM layer from within an application
  based on that OEM layer.

To delete a type definition:

1. Ensure that there are no instances of that type in the application.
2. Open the Application Configuration dialog.
3. Open the Manage Types page.
4. Select the type in the list.
   If there is no check mark in the Removable column, the type is in use some-
   where in the application (either as an instance or as a child of another type)
   or else this is a standard VTScada type, which cannot be removed.
5. Click the Remove button.

There is no confirmation dialog – the type definition will be removed
from the application.

Select a Type to export, edit or remove. Selection not required for import.

Type /	# of Descendants	Removable	Editable
Selector Switch	0		
Serial Port	0		
Siemens S7 Driver	0		
SMS Appliance	0		
SNMP Driver	0		
SpecialPurposeTag	4	✓	✓
SQL Logger	0		
SQL Logger Group	0		
TCP/IP Port	0		
Totalizer	0		
Trigger	0		
UDP/IP Port	0		
WorkStation Driver	0		

[Remove] [Edit]   [Import] [Export]

Automated Tag Configuration

All properties of a tag can be configured using an expression rather than a static value. This is an extremely powerful feature that you can use to create tags that configure themselves based on where they are placed in the application.

If you have created a tag that is a child of another tag, then you have seen this in action. The Area field of every new tag will automatically be the same as the parent tag's area (unless you apply an override to set a new area value). If you move a tag from one parent to another, its area field will change to match that of the new parent. The expressions described here are most effective when used in parent-child tag structures, where they can leverage information stored in a parent tag.

Fields that contain parameter expressions that you have created can be identified by their blue shading.

Parameter Expressions & Start Conditions

When using an expression to configure a parameter value, you can choose whether the value should be evaluated only when the tag starts (and re-starts), or whenever there is a change in one of the parameters. *This is an important decision. It changes the list of functions that you can use in the expression, and may cause undesirable side effects. Each non-optimized expression you create adds to the application's memory usage and CPU load. If there is a large number, the impact may be noticeable. Non-optimized tag expressions are intended for situations where configuration values must be obtained from equipment, which is reachable only after the application starts.*
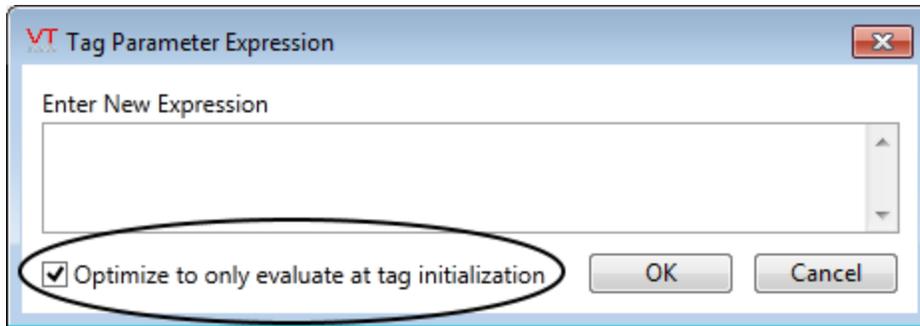
> **Warning:** Do not deselect the optimization option unless there is a clear need to do so. Misuse of non-optimized tag expressions can have undesirable side-effects. In particular, note that configuring a tag so that its properties can change dynamically may have an impact on logged values or cause unintended alarms.

When optimization is selected:

- The parameter value will be evaluated only when the tag is created, when the application re-starts, or when you explicitly change any property of this tag or a parent using the Tag Browser.
  The expression will not be re-evaluated during normal operations.
- Your selection of functions is limited to those that can be evaluated in script mode.
- Example: Configure the I/O address of child tags to use a base address configured in the parent context, plus an offset to a specific register.

When optimization is not selected:

- Expressions can use information obtained from hardware after tags have initialized and I/O operations begin.
  This is the primary purpose of non-optimized parameter expressions.
- The expression will be re-evaluated whenever any of the parameters used in the expression change.
  Be careful to consider all the possible effects of this fact.
- When a parameter changes, this tag and all of its child tags will be restarted, slowing the application temporarily and increasing the load on the CPU.
- Each non-optimized tag expression requires extra RAM, roughly equivalent to half that required for a typical tag.

- Your selection of functions is limited to those that can be evaluated in steady-state.
- Example: Configure the description and area to use values provided by hardware.

Using properties from other tags

To access the value of a property in a parent tag, preface the name of the property with two dots and a backslash (..\). If there is no ambiguity about the variable's location (that is, the variable name exists only in the current tag, or in an immediate ancestor) then only the backslash need be used. For a variable such as "Area" that will exist in both the current tag and the parent, add two dots in front of the backslash to force the expression to look upwards to at least the immediate parent. (Example: "..\Area"). VTScada will search upwards through the Parent-Child tree to find the first instance of a matching field name.

Do not add quotation marks to property names. Quotation marks should be used only to indicate literal text.

Links to further information are provided at the end of this topic.

Start Tag Expressions

An expression in the name field of a tag will not change the tag's name. Rather, this expression will control whether the tag starts or not. Examples of this can be seen in the various Station Tags, which may be configured with a varying number of pumps for each station. This is done by tying the number of pumps in the overall station tag configuration to an expression in the name field of each pump. If the number is 1, only the first pump tag will start and the station tag will have one pump. If the number if 2, then two pump tags will start and that is the number that the station appears to have.

Start tag expressions are a special case of parameter expressions, and also result in the field being shaded blue.

Overrides

Having made use of an expression to control a tag's configuration, you can choose to override that value in a given child tag. For example, a child tag which would otherwise inherit the area value "North County" might be assigned the area "South street". When a value that would otherwise be assigned based on a calculation is changed, this is called an "override". Assigning a value to a field that has no expression isn't overriding anything and is simply called "assigning a value". The difference can be seen in the color that the field will turn: orange for overrides, and green for simple assignments.

Related Information:

...Creating Expressions – VTScada Programmer's Guide – Reference for the VTScada scripting language.

...Access a Tag Value or Application Property – VTScada Programmer's Guide – Syntax rules for addressing tags in code.

...Tag Configuration Overrides – Set a non-calculated value for a tag instance.

...Context Tags – Can hold developer-defined parameters, to be referenced by child tag expressions.

Related tasks:

...Create or Remove a Parameter Expression – How to add or remove an expression.

...Create a Start Condition – Instructions for creating.

Create or Remove a Parameter Expression

Parameter expressions are created in the fields of a tag's configuration panel. You must have a configuration panel open and have configuration privileges to proceed.
If the current tag contains a parameter with the same name as the parameter you wish to access from the parent tag, you must add "..\" before the name to signify "parameter of a parent."

To create a parameter expression:

1. Right-click on the configuration field that is to be defined using an expression.
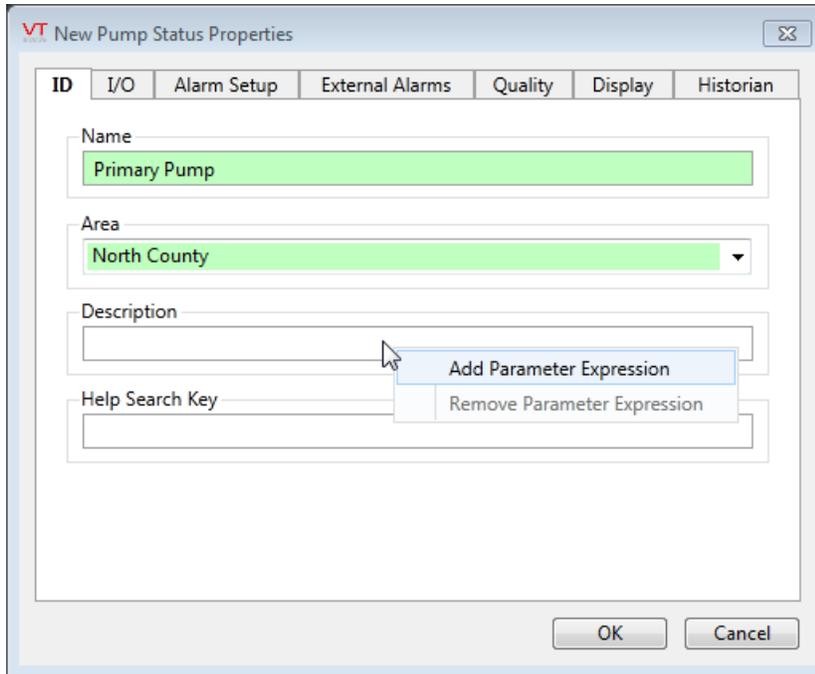
2. Click, "Add Parameter Expression".
    The expression editor will open.

3. Enter a valid expression.
    VTScada will not allow you to save an expression that contains an error.
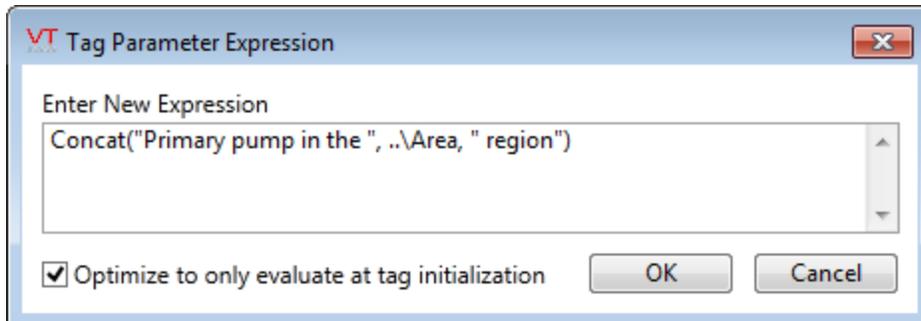
4. Click OK to save your work and return to the tag's configuration panel.
    If successful, the field should now be blue and show the calculated result of your expression. Hover the pointer over the field to view the expression in a tool-tip window.
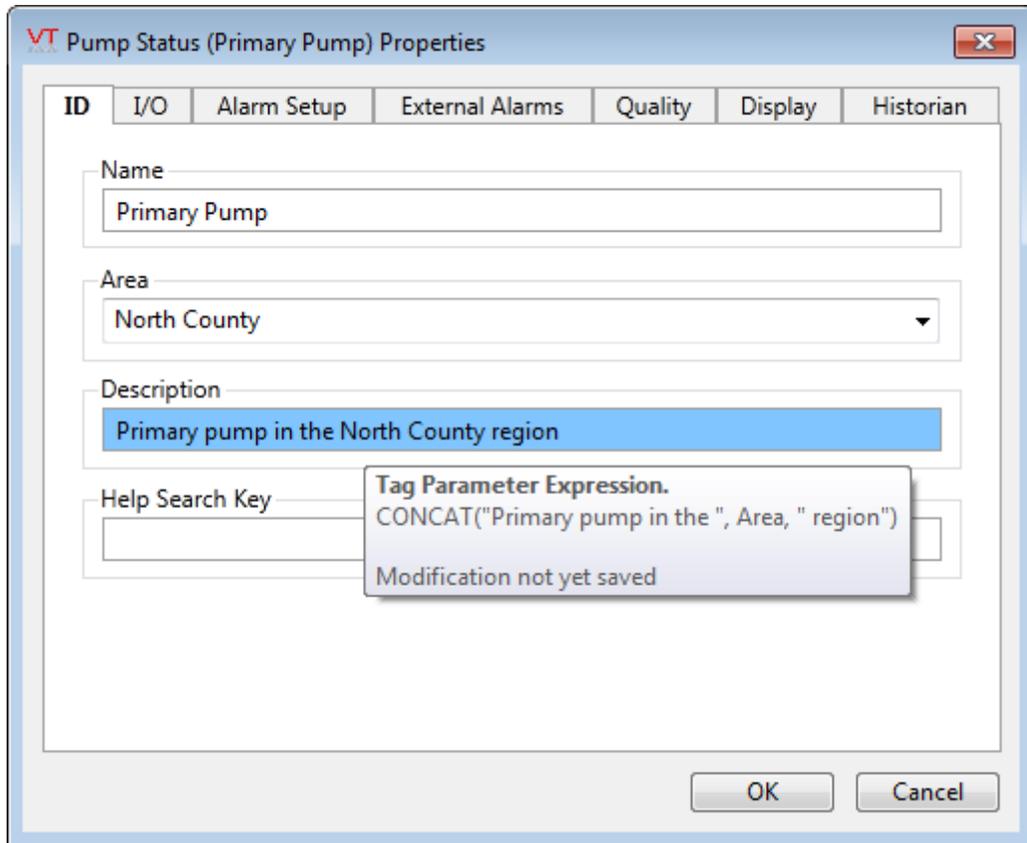
To remove a parameter expression:

1. Right-click on the blue field of a parameter expression.
2. Click, Remove Parameter Expression.
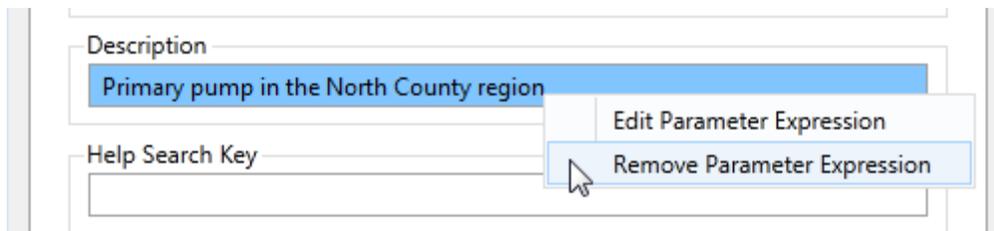
Related Information:

...Creating Expressions – VTScada Programmer's Guide – Reference for the VTScada scripting language.

...Create a Start Condition – Instructions for creating.

...Tag Configuration Overrides – Set a non-calculated value for a tag instance.

...Context Tags – Can hold developer-defined parameters, to be referenced by child tag expressions.

Create a Start Condition

If you have a custom type of tag that should be started in some contexts but not others, you can create a start condition by adding a parameter expression to the Name field. Pumps within a Station tag (MultiSmart or MPE) are examples of this[1]. The station tag structure contains a set number of tags, but will appear to hold only the number defined for any given station instance. Those in excess of the given number do not start and therefore seem not to be part of the configuration.
The expression used for a start condition must evaluate to TRUE (defined as non-zero) or FALSE (defined as zero).
Start conditions must be optimized to be evaluated only on initialization. There is no option to disable this feature.

> Note: You can explicitly set a tag's start condition to FALSE using the Tag Browser by right-clicking on the tag and using the menu that opens.
> If a tag's start condition evaluates to false, then it (and all of its child tags) will not be drawn and will not be included in your tag count. Disabled tags will not appear in the tag browser unless you check the option, Show Disabled.

To add a start condition to a tag:
1. Open the tag's properties dialog and right-click on the name field.
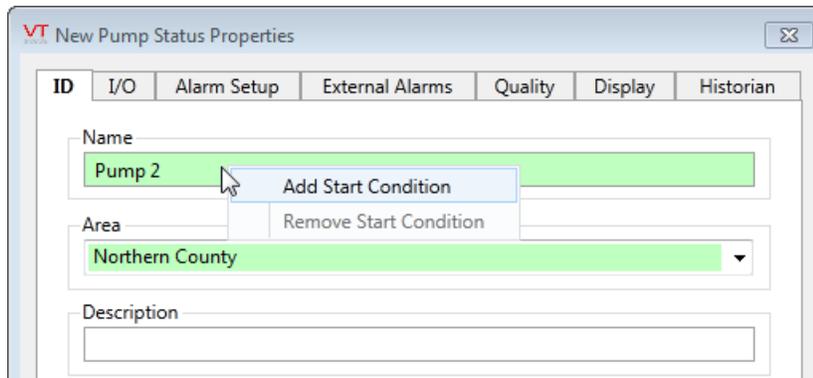
2. Select Edit Start Condition from the menu.

The Parameter Expression dialog opens.

3. Create an expression that will evaluate to TRUE or FALSE according to start-ing condition that will affect this tag.
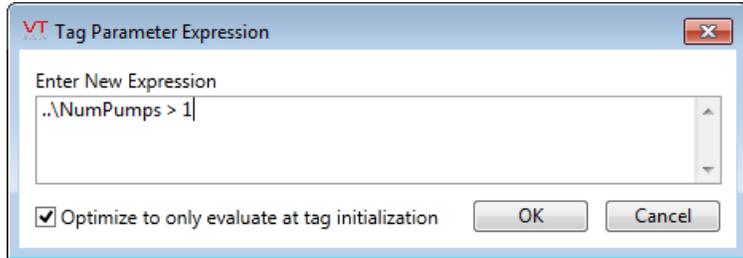
(See following note)

4. Click OK to save your work.

If the value of the station's NumPumps variable is 1 or less, then this pump (and all of its child tags) will not start.

(^{1}) The start condition of a station tag is more complex than this. For example, if you direct the tag browser to show disabled tags, the remain-ing pumps within the station will not appear. The analogy is still valid, since a start condition is exactly the tool you would use to emulate a sta-tion tag in your application.

(^{2}) In this example, it is assumed that the parent is a Context tag con-taining a variable named NumPumps, which can be set for each instance of a station. This would be created in the Context tag's Settings dialog, as shown:
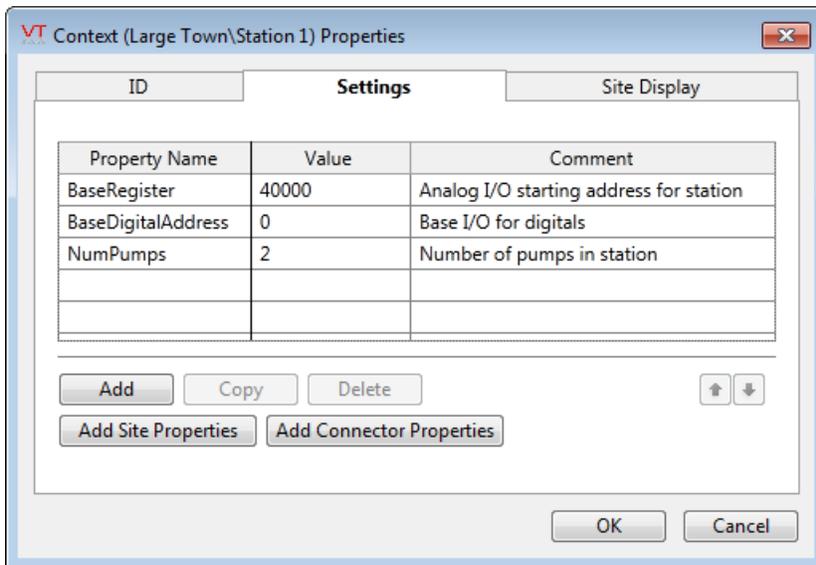
Related Information:

Relative Tag References & Parameter Expressions

Tag names within parent-child tag structures and user-defined tag types are built as shown in the following example:
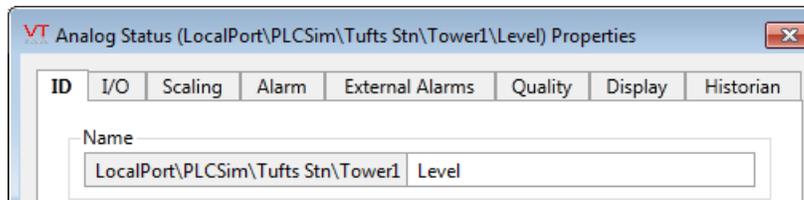
The name of each parent tag is part of the full tag name, where each parent name is separated from the next with a backslash.

Relative tag references (common when defining a port or driver) will look similar to the following:
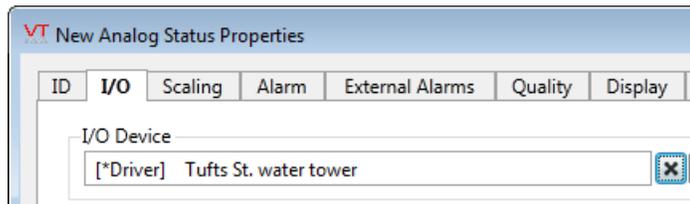
In Excel, Access and other programs, the backslash has its own meaning. Therefore, it is necessary to indicate that this character should be treated as just a backslash and not use that other meaning. This is commonly done by doubling the character:

A similar rule applies to parameter expressions, except that here there can be both backslashes and equal signs to mark. An extra leading back-slash is used for both:
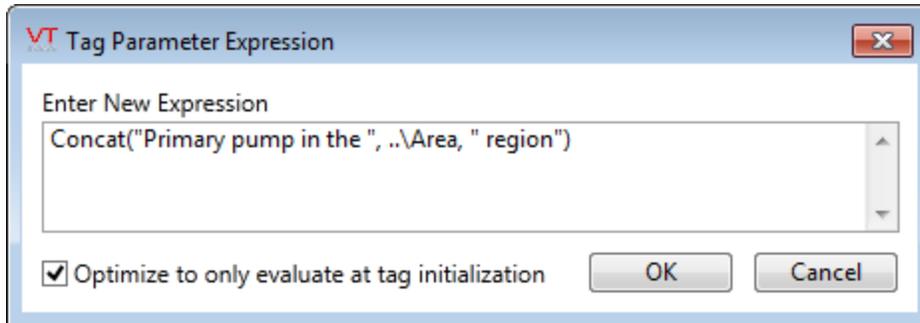
Note: When modifying tags in an exported spreadsheet or database, you must also add the leading backslash, as shown in these examples.

Related Information:

...Access a Tag Value or Application Property – VTScada Programmer's Guide - Syntax rules for addressing tags in code.

Tag Configuration Overrides

Note: This information applies only if the expression in question exists in a defined type. The code for the expression must continue to exist somewhere in order to be overridden. Otherwise, it is simply replaced.

Tag fields can be configured using expressions rather than static values. The most commonly encountered example is the Area field of any child tag; unless otherwise configured, it will use the Area value from the closest parent that has a defined area.

If you change the value of such a field, rather than allowing the expression to set the value, then you are applying an override to that field.

The difference goes beyond whatever importance might be attached to knowing what was changed when you set the value you wanted. It also changes what will be included if you export your tags to a database or spreadsheet.

For example, a MultiSmart Station tag has some 38 or more child tags. If you export your application's tags to a spreadsheet, you will see only the

one entry for the MultiSmart – not for the child tags. This is because those tags can be re-created by code and therefore do not need to exist as individually configured tags in the database.

However, if you override a parameter within one of the MultiSmart's child tags, and then export to a spreadsheet, you will find an entry for that child in addition to the parent MultiSmart. The override value is stored in the tag database rather than being calculated through code.

When you override a parameter, the field will turn yellow instead of green, signifying that an expression has been overridden rather than that a simple configuration has been applied.

To remove an override: Right-click on the orange field, and select "Remove Override" from the menu. The contents of the field, as set in the tag type definition, will return.

Related Information:

...Manage Types Using a Spreadsheet or Database – Export a type definition for editing outside VTScada.

Import and Export Tags

Tag configuration is stored within the VTScada repository. This topic describes how you can work with tags outside of the Tag Browser.

Note: Do not attempt to edit any of the files in the .sync folders. Damage to, or destruction of your application is likely to result.

Note: If your goal is to copy a custom tag type definition from one application to another, see: Copy Types to Other Applications.

Warning: Certain specialized applications may contain parameter data that exceeds the 8k cell limit of Excel. An export will proceed, but those parameters will not be included. VTScada will not allow the spreadsheet to be synchronized back into the application.

Some developers find that it is efficient to create or edit tags in a tabular format, such as a spreadsheet. This is particularly true when there are a

large number of similar changes to make. VTScada makes it easy for you to export your tag database to either a Microsoft Excel™ or Microsoft Access™ file, then synchronize the changes made in that file with the running application.

Tags are both exported and imported using the same page of the Application Properties dialog, "Export/Sync Tags".

Notes:

- If selecting the Excel format, the Microsoft Excel program must be installed on the computer. Microsoft COM objects, provided only with a copy of Excel, must be present in order to create a spreadsheet that is fully compatible with Excel.

- Tags that are created automatically as children of a complex type (those within a MultiSmart or MPE station, or your own user-defined types) will be exported without configuration data that would be generated automatically when the complex type is created. Override values will be exported for these values, and you may apply or remove override values in the exported file.

- You cannot use the procedure described in this topic to modify the structure of a complex type by adding or removing child tags. A separate tool exists for that purpose – See: Manage Types Using a Spreadsheet or Database.

- If you have custom tags with a large number of parameters, you should be aware of the following limits: If exporting to Access, the maximum number of tag parameters is 256. If exporting to Excel, the maximum number of parameters is 65,536. None of the tags built into VTScada approach these limits.

The export file will contain information that identifies both the application and its current version. This will be used by the VTScada configuration management system when synchronizing the modified database. Look for the table or worksheet, VTS_Reserved. The synchronization process gives you the following option for what will happen to the spreadsheet or table after you import it back to the application.

- You may direct VTScada to update the file by performing a fresh export, thereby ensuring that you can continue editing the file and synchronizing those changes. Choose this option when you are making a series of consecutive edits and synchronizations.

- You may direct VTScada to delete the file, thereby ensuring that you must perform a fresh export before the next editing session. Choose this option when you expect to work in the Tag Browser before going back to the external database.

In general, it is best to work on an exported file that matches the current tag database. Importing an older file that does not include recent changes, may lead to errors.

Tags can be imported (synchronized) without re-starting your application. Note that if there are a large number of tags to import or export, the operation may take a few minutes. The operation will be faster if the application is not running. When you request either operation, a dialog will remind you of this:
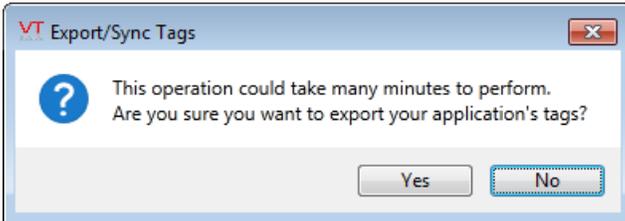
When exported to an external data file, each tag type will be stored in its own table or worksheet. Within a tag type table, each instance of a tag will be stored on a separate row, with a field for each attribute of the tag. (See following example) If a tag is part of a parent-child tag structure, the full name of the tag within the structure will be shown.

Relative tag references and parameter expressions will be adjusted to avoid conflicts with Excel conventions. Backslashes and other expression operands will be preceded by an extra backslash. Commonly-seen formats include the following:

..*TagType	Reference to a parent tag of the named type
\=	Optimized tag parameter expression follows (snapshot expression)
\%	Non-optimized tag parameter expression follows (refreshing expression)

Related Tasks:

...Export Tags – Send the application's tag database to an external file.

...Synchronize (Import) Tags – Bring changes, made in the external file, into your application.

Related Information:

...Importing Tags from One Application to Another

...Importing From Older Versions of VTS– Moving tags across VTScada versions or applications.

...Relative Tag References & Parameter Expressions – Includes inform-ation on expressions are formatted in Excel or Access.

Export Tags

Sends the tag database to a Microsoft Access™ or Microsoft Excel™ file. Some developers find it more efficient to create or edit large numbers of

tags in the tabular format of those programs, synchronizing their work with VTScada to bring the new and changed tags back into the application.

> Note: The structure of custom-built tag types will not be exported.

To export your tag database:

The steps to either import or export tags begin with the same page in the Application Configuration dialog:

1. Open the Application Configuration dialog and select the Export/Sync Tags page.
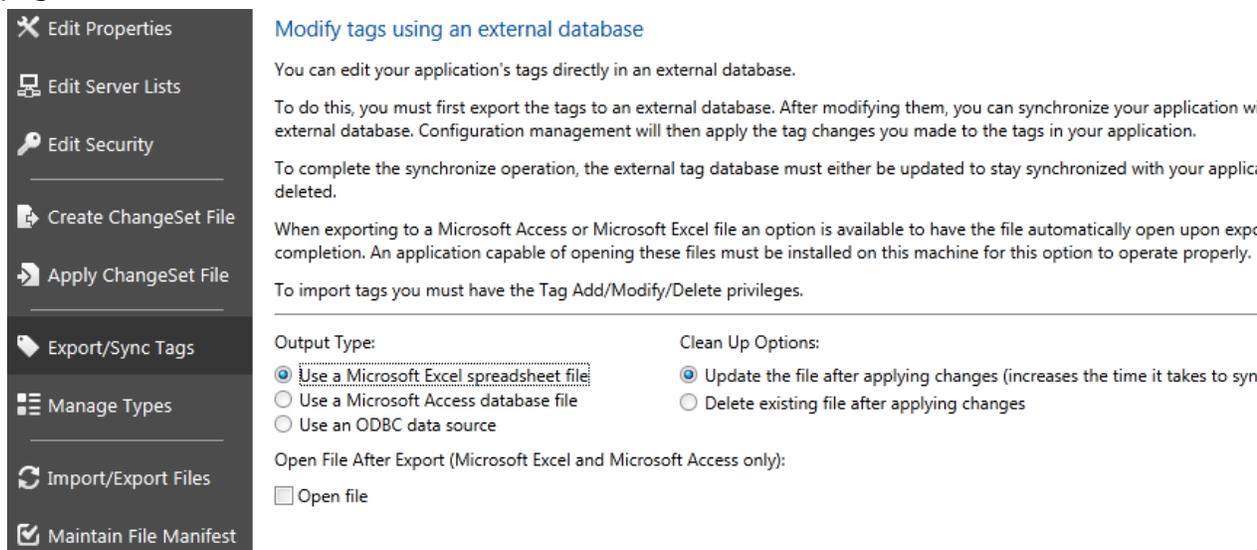
2. Select the output type.
   If ODBC is to be chosen, you must have already defined a Data Source Name (DSN) for your database using the Microsoft ODBC Data Source Administrator.

3. Click the Export button.
   If exporting to Access or Excel, you will be prompted for a file name and path.
   If exporting to an ODBC data source, you will be prompted for the DSN.

4. Provide the destination for the exported tags and click OK.

Troubleshooting:

- The file could not be created.

Ensure that you are writing to a folder that you have access to with your Windows account.

- The exported file does not contain the child tags of my custom tag structure. It won't. You can edit those using the Manage Types dialog.
- A warning message tells me that Microsoft Excel (or Access) is not installed. Excel or Access must be present. To create a file that is fully compatible with those programs, the related Microsoft objects must be present on this computer.

Related Information:

...Manage Types Using a Spreadsheet or Database – Change the definition of custom types

...Edit Parent Type Properties

Next Steps:

...Synchronize (Import) Tags – Bring changes, made in the external file, into your application.

Synchronize (Import) Tags

Given a third-party spreadsheet or database containing tag configuration, use the following procedure to synchronize your application's tag database with contents of that file.
This will import information from the file adding, updating, or removing tags in your application as required. Following the import, you can choose to delete the external file as part of the process, or you can replace it with a fresh export of the application's tag database.

Note: Tags are matched based on their GUID (Column A in a worksheet). If the file being synchronized does not contain a GUID matching one in the tag database, VTScada assumes that your intent is to delete that tag. If two rows have the same GUID value, an error message will be displayed and the synchronization operation will stop.
For all tags, any parameter value set or changed in the external spreadsheet or database will replace the current or default value in the VTScada database.

1. Open the Application Configuration dialog and select the Export/Sync Tags page.

2. Choose the format matching the file to be imported / synchronized.
    If ODBC, you must have already defined a Data Source Name (DSN) for your database using the Microsoft ODBC Data Source Administrator.

3. Select the clean-up option.
    As a guideline, use the update option when making a series of modifications. Use the delete option when there will be significant changes to the internal tag database before the next round of external edits.

4. Click the Sync button.
    If importing from Access or Excel, you will be prompted for a file name and path.
    If importing from an ODBC data source, you will be prompted for the DSN.

5. Choose the source file or DSN and click OK.
    A message will inform you of how many tags were imported.

Note: If you attempt to import tags that were exported from a different application, you will see an error message warning you that the GUIDs do not match. The tags will not be imported.

Troubleshooting:

- The error message reports that the GUID of the file does not match that of the application.

The file was created in a different application. Synchronizing it with the current application may have undesirable side-effects, but it can be done by first removing the GUID from the VTS_Reserved table or worksheet of the file.

- Conflicts were reported with one or more tags.
  For one or more rows in the external file, column A – the unique identifier – is duplicated in the file. For new tags, column A must be blank.

- Tags were deleted from the application when the file was synchronized.
  The information in column A – the unique identifier – was removed or changed for one or more rows. Repair using the VTS version control system. Do not remove rows in the external file (or their unique identifiers) unless you intend to have them removed from the application.

Related Information:

...Importing From Older Versions of VTS- Moving tags across VTScada versions or applications.

...Importing Tags from One Application to Another

Importing From Older Versions of VTS

Prior to the release of VTS 10, tags were stored in a database file named Points.MDB. For most applications, this was a Microsoft Access database. (Microsoft SQL Server was used by a very small number of applications.) These Points databases did not include any information to identify the application, or its version.
You can import the tags from an older application's Points.MDB file. The steps are the same as those provided in the previous topic but, be aware of two potential problems:

- The file, Points.MDB stores tag instances, but not type definitions. If the source file used type definitions from an OEM layer, then the destination application must also have access to that layer's code in order to use the tags.

- Conflicts between tag names in the older database and the current application may not be resolved the way you would prefer. Since the older database has no version control information, VTScada is not able to

automatically determine how to resolve a conflict between two tags of the same name.

Note that, after importing an older tag database, you must compile the current application (using the Import File Changes button) before you be able to see all of the imported tags.

Importing Tags from One Application to Another

After exporting tags from an application, if you then attempt to import those tags to a different application you will see an error message similar to the following:
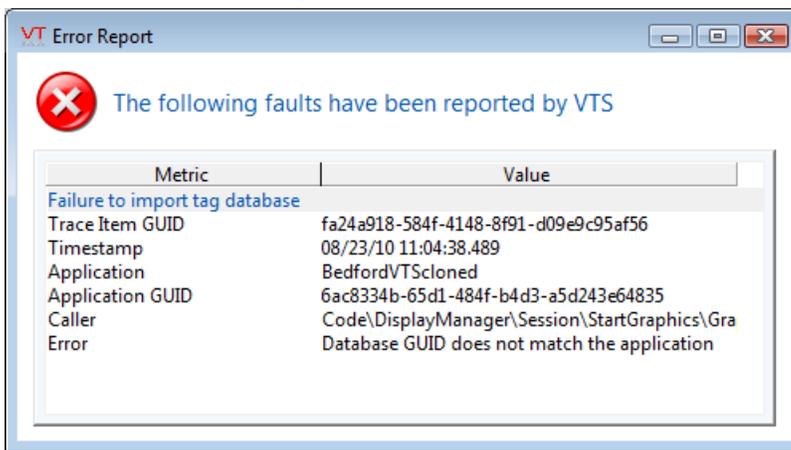
This is meant to protect your application in the case that the imported database may include type definitions that are not present in the destination, or possibly a name conflict between the current and imported set of tags.

If you are satisfied that these two problems will not occur, and you are sure that you want to import the database, then you need only remove the identifying information from the external tags database. This is stored in a table (or worksheet) named "VTS_Reserved". After the GUID entry in this table is removed or set to "00000000-0000-0000-0000-000000000000" the tags may be merged into any VTScada application just as the tags from an older VTScada version may be imported.

> Note: Warning: You should remove all menu tags before proceeding from the spreadsheet or database before proceeding.

Tag Reference

The flexible nature of VTScada is due to the custom building blocks upon which it is based. These building blocks are called "tags".

> Note: Tags were originally referred to as points. This term is antiquated but may still be found in some contexts.

A tag is a software component that can communicate with objects in the physical world. For flexibility, each tag type is a specialized component that can be combined with other tag types to build diverse systems. For example:

- Port tags handle communication between the VTScada server and the outside world, whether through a serial port or a TCP/IP port.
- A specialized driver tag is available for nearly any brand of PLC hardware. Driver tags rely on port tags in order to communicate with the hardware.
- Input and output tag types, both analog and digital, monitor and communicate with individual addresses on the remote PLC. I/O tags rely on driver tags to ensure that communications are properly formatted for the remote hardware.
- Alarm tags and Logger tags can be linked to any I/O tag to monitor and record values.

The term, "communications chain" describes the linkage of VTScada with remote hardware through the I/O tag, to the driver tag, to the port tag. By choosing appropriate tags for each link in the chain, you can build VTScada applications for nearly any situation.

It is typical to begin each application you create by configuring a port tag to accept the data that is flowing through your PC, and a device driver tag to provide an interface to physical I/O devices such as programmable logic controllers (PLC), remote terminal units (RTU), I/O boards, or to Windows system features such as Dynamic Data Exchange (DDE). Once you have these essential tags in place, you can create analog, digital and other tag other types to make your application communicate with the outside world.

Standard versus Custom Types.

All of the various types of tags that come with VTScada are described in this chapter. These are known as the standard types.

VTScada also gives you the ability to define your own types of tag. This can be done by creating groups that include a port, a driver, I/O, alarm and other tags that fully describe a type of equipment, such as a pumping station, motor, well, etc. You are also able to use the VTScada programming language to build completely new types of tag, designed specifically for your needs. Custom-programmed tags do count towards your license limit.

Related Information:

...Widget Reference – Ways to display tag values and create user-interface elements such as buttons.

...Tag Concepts and Features– Includes licensing details, and standard attributes.

...Port Tags – Reference for configuring ports.

...Communication Driver Tags – Configuring device drivers.

...Input Tags – All tags that read I/O values.

...Output Tags – All tags that write I/O values.

...Alarm System Tags – Alarm tags common to all VTScada systems.

...Alarm Notification System Tags – Alarm tags used by systems that include the Alarm Notification System option.

...Logging and Reporting Tags – Data collection and display.

...Calculation and Inquiry Tags – All tags that can use calculations to find system information.

...Station and Site Tags – MultiSmart and MPE stations

...Configuration Tags – Fonts, Context tags, Style Settings and more.

Tag Groups

VTScada has tag groups that are used to organize the standard tag types into collections based on their value, capabilities, or other characteristics. Some of the tag types belong to multiple groups based on shared characteristics.

The tag groups enable selection of a specific tag type in a tag-associated drop-down list. For example, when creating a new driver tag type to be associated with an Analog Input or Digital Input tag, clicking the tag browser button to the right of the I/O Device field opens a group dialog from which you can select the type of driver tag you wish to create.

Analogs Group

The analogs group assembles those tags that read or write analog (continuous floating point) values to a device.

The group is composed of the following members:

Analog Control tags	Analog Input tags
Analog Output tags	Analog Status tags
Counter	Pulse input
Rate of Change	Totalizer

Container Group

The container group is composed of those tags that can have contributors. (An example of a container/contributor relationship might be at a power generation station, where the container tag (also referred to as an owner tag) would calculate the overall power output for the entire station based on the value of its many contributor tags.)

The container group is composed of the following members:

Analog Statistics	Context
Data Flow RTU	Digital Statistics
MPE Duplexer Station	MPE SC Station
MultiSmart	Polling

Digitals Group

The digitals group assembles those tags that read or write digital (discrete) values to an I/O device.

The group is composed of the following members:

Alarm	Alarm priority
Data Flow RTU	Digital control
Digital input	Digital output
Digital status	Network status
Polling driver	Pump status
Roster	Trigger

Drivers Group

The drivers group consists of tags that are used to provide an interface to physical I/O devices or to Windows system features, such as Dynamic Data Exchange.

The group is composed of the following members:

Allen-Bradley driver	Control & information/Ethernet I/P driver
Data Flow RTU	CalAmp diagnostic driver
DDE Client	DNP3 driver
Driver Multiplexer	MDS diagnostic driver
Modbus Compatible Device	Motorola ACE driver
Omron host link driver	OPC client driver
Siemens S7 driver	Polling driver
SNMP driver	Workstation driver

Loggers Group

The VTScada loggers group consists of tags that have a numeric value, and that can have their data logged to disk for data examination and analysis, and plotting on the Historical Data Viewer page by having a Logger tag associated with them.

The group is composed of the following members:

Allen-Bradley driver

Analog status	Data Flow RTU
CalAmp diagnostic driver	DDE Client
Digital status	DNP3 driver
Driver Multiplexer	MDS diagnostic driver
Modbus Compatible Device	Network status
Notebook	Omron host link driver
OPC client driver	Pump status
Siemens S7 driver	SNMP driver

Numeric Group

The numeric group is composed of tags that have a numeric value:

Allen-Bradley driver	Alarm
Alarm Status	Analog Control
Analog Input	Analog Output
Analog Status	Analog Statistics
Calculation	CIP driver
Counter	Data Flow RTU
DDE Client	Deadband Control
Digital Control	Digital Input
Digital Output	Digital Status
Digital Statistics	DNP 3.0 driver
Driver Multiplexer	Function
Historian	History Statistics
IP Network Listener	MDS Driver

Modem	Modbus Compatible Device
Motorola ACE driver	
Network Status	Omron Host Link driver
OPC client driver	Polling driver
Pulse input	Pump Status
Rate of Change	Roster
Script	Selector Switch
Serial Port	Siemens S7 driver
SNMP driver	SMS Appliance
SQL Logger	TCP/IP Port
Totalizer	Trigger
UDP/IP Port	Workstation driver

Outputs Group

The outputs group is composed of tags that have can be used to write values:

Analog Control
Analog Output
Digital Control
Digital Output
MultiWrite
Selector Switch

Ports Group

The ports group consists of tags that enable communications between a VTScada application and a physical system. The ports group is composed of the following members:

Motorola Gateway IP
Serial port
TCP/IP port
UDP/IP port

Stations Group

The ports group consists of tags that enable communications between a VTScada application and a physical system. The ports group is composed of the following members:
MPE SC Series Station
MPE Duplexer Station
MultiSmart Station

Trenders Group

The Trenders group consists of tags whose value can be plotted on the Historical Data Viewer page. The tags composing the Trenders group are:

Allen-Bradley driver	Alarm Status
Analog Control	Analog Input
Analog Output	Analog Status
Calculation	Counter
Data Flow RTU	CalAmp diagnostic driver
DDE Client	Deadband control
Digital control	Digital input
Digital output	Digital status
DNP3 driver	Driver Multiplexer
Function	History Statistics
MDS diagnostic driver	Modbus Compatible Device
Network status	Notebook
Omron host link driver	OPC client driver
Pump status	Rate of Change
Serial port	Script
Selector Switch	Siemens S7 driver
SNMP driver	UDP/IP port
TCP/IP port	Totalizer

| Trigger | Workstation driver |

When Trenders is selected in the Tag Browser's Types drop-down list, and the New button is clicked, the Trenders Group dialog will open, enabling you to select the type of tag you wish to create.

To create a new tag, select the radio button corresponding to the type of tag you wish to create, and then click OK. A new tag properties folder associated with the selected tag type will open for configuration.

Tag Types Listed in Alphabetical Order

All of the VTScada tag types are listed here. For each, a reference is provided to the full description of the type.

Alarm Tags	Alarm tags are used to establish rules as to when an alarm should be triggered, and what behavior should occur.
Alarm Database Tags	Link alarms to Historians
Alarm Priority Tags	The Alarm Priority tag type enables you to specify different alarm indications (such as the frequency of the alarm tones, the tone cycle, alarm priority, and alarm highlight color) for alarms of different priorities and categories.
Alarm Status Tags	This tag looks for active or unacknowledged alarms matching a given set of characteristics.
Allen-Bradley Driver Tags	The Allen-Bradley driver provides an interface to an Allen-Bradley PLC, allowing communication via: • Serial port • KT/SD I/O board • TCP/IP Note: Please refer to CIP Driver tags for information on Allen Bradley ControlLogix tags.
Analog Control Tags	It is the role of Analog Control tags to transmit analog data entered by the user to an I/O device.
Analog Input Tags	The role of Analog Input tags is to represent incoming analog data values from equipment processes
Analog Output Tags	It is the role of analog outputs to take analog data from the user or from another tag, and write it to an I/O device.

Analog Statistics Tags	Collects maximum, minimum and average values over a variety of time periods.
Analog Status Tags	It is the role of Analog Status tags to take analog data from an I/O device.
CalAmp Diagnostic Driver Tags	These tags enable users to read statistics from any given Integra-TR or Integra-H CalAmp unit.
Calculation Tags	The Calculation tag type is used to wrap an expression, with the resulting value of the Calculation tag being the result of the expression. This new tag type replaces the older Function tag type.
CIP Driver Tags	These tags provide an interface between VTScada and hardware that uses the CIP or ENIP standards for communications.
Comm Link Sequencer Tags	Comm Link Sequencer tags may be added when your application includes several device drivers that must share the same communication channel and that are themselves unable to determine when the channel is free.
Context Tags	Context tags are used as parents in parent-child tag structures.
Counter Tags	Counter tags provide a means to count events such as pump starts or equipment cycles.
Data Flow RTU Driver Tags	Data Flow RTU drivers are used to provide an interface to a Data Flow RTU.
DDE Client Tags	The DDE Client enables tags to read from and write to a local or network DDE server.
Deadband Control Tags	Deadband control tags monitor the value of any tag with a numeric value and compare the monitored tag's value to a low set point and a high set point.
Digital Control Tags	Digital Control tags transmit digital data entered by the user or from another tag and write it to an I/O device.
Digital Input Tags	Digital inputs represent incoming digital data values from equipment, and provide widgets that change color, appearance, or text to impart the value of the digital input and associated equipment.
Digital Output Tags	Digital Output tags take digital information from the user or from another tag and write it to the I/O device.

Digital Statistics Tags	Collects cycle counts and run-time statistics over a variety of time periods.
Digital Status Tags	Digital Status tags accept incoming digital data from an I/O device.
DNP3 Driver Tags	The DNP driver is a DNP 3.0 Master that implements all the objects defined in DNP subset level 3, except for the "Frozen Counter" type.
Driver Multiplexer Tags	Driver multiplexer (DriverMUX) tags allow you to set up redundant lines of communication between I/O tags and equipment.
Enron Modbus Driver Tags	Enron Modbus is a variation on the standard Modbus protocol that allows for the transmission of stored historical data and alarm/event logs.
Fisher ROC Driver Tags	The Fisher ROC driver is used to communicate with devices that support the ROC protocol.
Font Tags	The Font Style tag type is used to specify fonts for the drawing tools that use text.
Function Tags	The Function tag type is used to perform mathematical and logical calculations using numeric values or the values of other tags or both.
GE Series 90 Driver Tags	Use this driver to communicate with GE series 90 PLCs / RTUs. Two modes of communication are supported: TCP/IP using the Service Request Transport Protocol (SRTP) or serial using the Series 90 Protocol Express (SNP-X) protocol.
Historian Tags	The Historian Tag is used to write data that is to be logged to storage.
History Statistics Tag	Each instance of a History Statistics tag will perform 7 statistical calculations, using the data collected from a numeric tag's value over a defined time period.
IP Network Listener Tags	Add IP Network Listener tags to your application if you want to be able to accept inbound TCP or UDP connections.
Koyo Driver Tags	The Koyo driver started out as a Texas Instrument (TI) driver. Koyo purchased the technology, therefore new PLCs using this protocol bear the Koyo name. This driver works with the 200 / 300 and 400 series of PLCs.

Logger Tags	The Logger tag type monitors the value of another tag and on a set interval, sends the values to a Historian for recording.
MDS Diagnostic Driver Tags	These tags enable users to read statistics from any given MDS radio.
Menu Item Tags	Represent page menu items and drawing tools within the various palettes.
Modbus Compatible Device Driver Tags	The Modbus Compatible Device provides an interface to a Modicon PLC, allowing communication via: - Serial RTU - Serial ASCII - SA-85 I/O card for ModBus Plus communications
Modem Tags	Modem tags represent a physical modem for your system, and enable outgoing and incoming calls.
Motorola ACE Driver Tags	The Motorola ACE driver should be used for all Motorola ACE / Moscad RTUs in your system.
Motorola IP Gateway Tags	This type is used to create an interface to a Motorola MDLC network. They are intended for use with the Motorola ACE driver tags.
MPE SC Series Station Tags	This tag will support either of the SC1000 or the SC2000 controllers. "MPE" stands for Motor Protection Equipment.
MPE Duplexer Station Tags	A single MPE Duplexer Station tag includes all of the I/O required to monitor and operate a MPE Duplexer station.
Multi-Write Tags	Configure a set of values that will be written to a list of registers when triggered.
MultiSmart Station Tags	A single MultiSmart Station tag includes all of the I/O required to monitor and operate a MultiSmart pumping station.
Network Status Tags	View the connectivity of up to two workstations, allow you to trigger an alarm should a monitored workstation become disconnected, and enable you to log this data to file.
Notebook Tags	Controls where operator notes are stored, and may contain pre-configured notes.
Omron FINS Driver Tags	"FINS" stands for Factory Interface Network Service, which is a protocol for sending messages between controllers on Omron

	FA networks. The VTScada FINS driver takes care of selecting which of the available protocols to use and which PLC unit, node and network to communicate with.
Omron Host Link Driver Tags	The Omron Host Link driver provides an interface to an Omron PLC via a serial port.
OPC Client Driver Tags	The OPC Client driver enables VTScada to act as an OPC Client to an OPC Server, enabling the exchange of data from a VTScada application to an OPC Server.
OPC Server Setup Tags	The OPC Server Setup tag is used to turn your application into an OPC Server.
Polling Driver Tags	Polling driver tags provide controlled communication between drivers and ports.
Pulse Input Tags	Pulse input tags accept incoming analog data from a pulse input device (such as a rain gauge, transducer, or a gas, water, or electric meter).
Pump Status Tags	Pump status tags accept incoming digital data from pumps.
Rate of Change Tags	Rate of Change tags are used to display how quickly another tag's value is changing.
Realm Display Setup Tags	The Realm Display Setup tag is used to control what options are available to a VTScada Internet Client for managing its own screen parameters and appearance.
Report Tags	The Report tag type is used to track the data of a user-defined group of tags to a user-defined output format at regular intervals.
Roster Tags	Roster tags are designed to work with the VTScada call-out system. Using a Roster tag, you may configure contact information for up to 8 operators
Script Tags	A Script tag monitors the value of another tag and enables the execution of a script when the monitored tag's value changes.
Selector Switch Tags	Selector Switch tags provide the means to create three-position switches that can be used (among other things) for implementing Hand-Off-Auto (H-O-A) switches.
Serial Port Tags	The Serial Port tag type opens a serial port for use by a driver.

Siemens S7 Driver Tags	The Siemens S7 driver provides an interface to Siemens PLCs of the following types: • S7-200 • S7-300 • S7-400
SMS Appliance Tags	Works in conjunction with the Alarm Notification System option to allow alarm notifications to be sent as text messages to a mobile cellular device
SNMP Driver Tags	The SNMP driver can be used to communicate with any device that uses the Simple Network Management Protocol (versions 1 or 2c).
SQL Logger Tags	Selects a source tag that is to be logged to an SQL Logger Group.
SQL Logger Group Tags	Identifies an ODBC-compatible database, to which selected tag values will be written.
String I/O Tags	String I/O tags are used to read or write text values.
Style Settings Tags	Allow you to link visual indicators to tag types, helping to ensure that all indicators for those tags follow a consistent visual style.
TCP/IP Port Tags	The TCP/IP port tag type enables you to connect your VTScada application to a series of hosts, allowing you to transmit data across a network or over the Internet.
Totalizer Tags	The totalizer collects the integral of another tag's value over a time period.
Transaction Logger Tags	Transaction Logger tags can be used to record a collection of historical values as a single transaction. These are useful for material-handling applications that must record transfers of custody.
Trigger Tags	Trigger tags can be used to activate a process at a set time and date.
UDP/IP Port Tags	This tag type can be substituted for a TCP/IP port tag for a driver to allow serial drivers to transparently use UDP/IP.
Workstation Status Tags	Workstation status driver tags can be associated with one or

	more Analog Input tags to access data that the driver obtains from the Windows Management Instrumentation (an API in the Windows operating system that enables devices and systems in a network to be managed and controlled)

Port Tags

Driver tags need to know where to find the hardware they will communicate with. Rather than build this information into the driver tag, VTScada provides Port tags, allowing you to define the communication details once, then use that configuration for several drivers. If the configuration changes later, then you need only update one tag.
Port tags tell the VTScada drivers about the nature of the physical connection (serial port, TCP/IP over Ethernet, or UDP) and the required configuration details such as IP address, port number, transmission speed, whether a modem is attached, etc.

Related Information:

... Motorola IP Gateway Tags

...Serial Port Tags

...TCP/IP Port Tags

...UDP/IP Port Tags

Motorola IP Gateway Tags

Note: Motorola ACE features will be available only if your VTScada license includes this option.

Not counted towards your tag license limit.
This type is used to create an interface to a Motorola MDLC network. They are intended for use with the Motorola ACE driver tags.

Reference Notes:

In addition to adding an IP Gateway tag, you must also install a Motorola configuration file, named MDLC_TYP.CFG. This lists the possible RTU table configurations. If there are errors in this file, you can expect "type mismatch" error messages.

The MDLC_TYP.CFG file must be placed in your application folder, and imported to the application using VTScada configuration management. (see: Import File Changes or Manage Files with the File Manifest.)
The Motorola IP Gateway API is made available via a Component Object Module (COM) object, ACEIPGatewayAutomation.exe. This will be installed on your system when VTScada is installed.
The gateway will support 20 concurrent polling requests.

Motorola IP Gateway: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags

representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Motorola IP Gateway: Settings Tab

Use the settings tab to configure the primary gateway and to control the time synchronization feature.
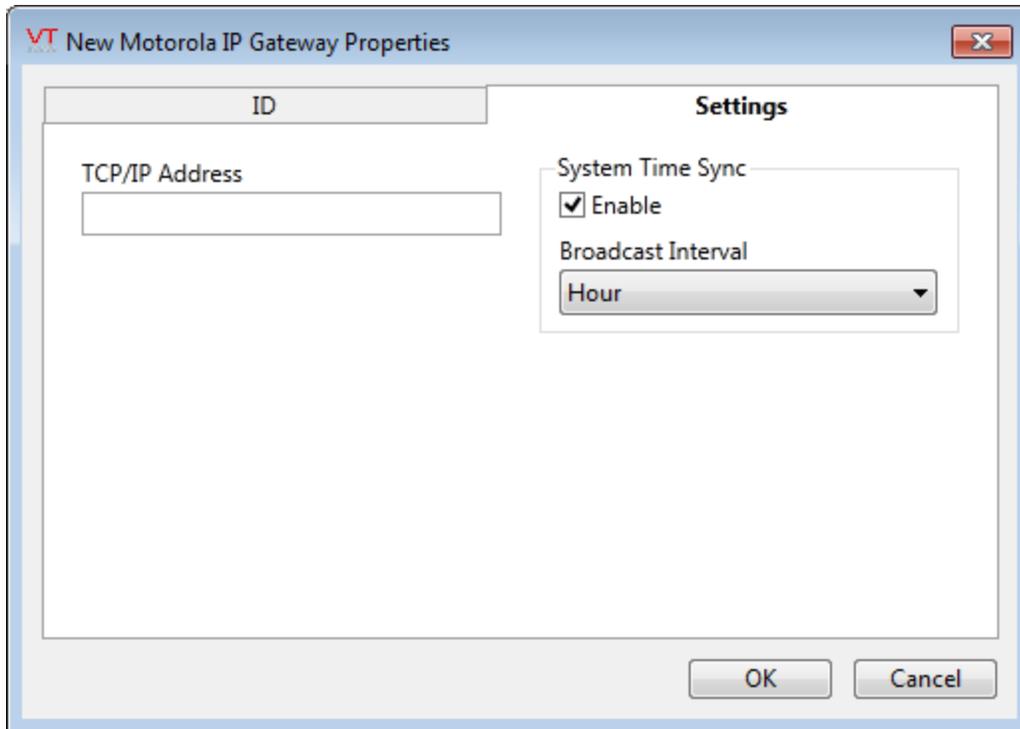
TCP/IP Address

> Provide the gateway's primary address in this field. This must be in IPv4 format, rather than a named address.

System Time Sync

> The enable box should be checked if you intend to use the system time synchronization broadcast, which sends the current local time to all devices on the MDLC network.
>
> Use the Broadcast Interval control to select the appropriate timing for your system.

Motorola IP Gateway Tag Widgets

There are no widgets for the Motorola IP Gateway tag.

Serial Port Tags

Not counted towards your tag license limit.

The Serial Port type opens an RS-232 connector port for use by a driver.

If you intend to use a modem, either for communication to the remote

telemetry unit, or for use by the Alarm Notification System, then you will need a Serial Port tag.

Reference Notes:

The transmit and receive buffer lengths are fixed at 16384 bytes.

The Data Transmit Ready (DTR) parameter of the serial port is fixed at 1

Note: Serial port tags may use the same port for different purposes on two different PCs. This is achieved by the re-use of serial ports on backup machines in a remote application[1].

The application property, SerialShareSemaphore has been marked as deprecated. In the unusual event that you have multiple devices attempting to use the same channel at the same time across multiple ports, you might consider adding Comm Link Sequencer Tags to serialize requests. A valid Serial Port Tag will have one of two values:

- 0 (zero) indicates a good connection.
- 2 indicates that the port is not available.

Related Information:

Comm Link Sequencer Tags – Serialize requests for the same communication channel across ports.

Serial Port Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

---

[1]An application that simultaneously runs on two or more computers. Usually configured with at least one primary server and one backup server.

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Serial Port Properties: Settings Tab

Use the settings tab to configure the tag for the communication parameters required by your hardware. The default settings for data bits, speed and parity are those which are most commonly in use.

When Use Modem is not checked, you must specify the COM port number in use. If Use Modem is checked, the Port Number field will be disabled and the Phone Number field enabled.

The required values for all the configuration options should be available from the specification provided with the hardware device connected to your serial port.
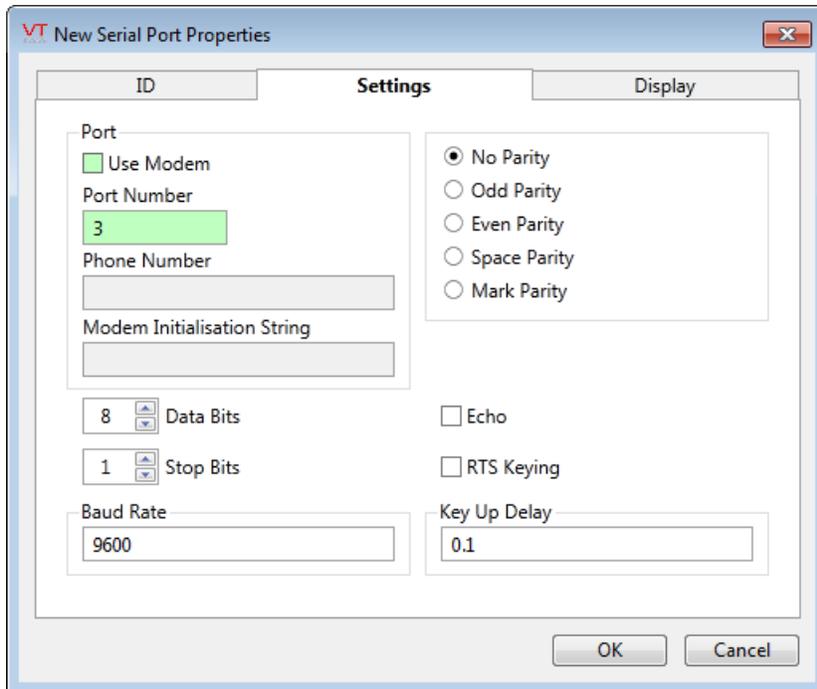
Use Modem

Provides control over whether a modem or a serial port is to be used.

This check box must be selected, and a remote phone number phone number provided if the driver attached to this port is to be a modem tag.

When the Use Modem check box is selected, the Phone Number and Modem Initialization fields will be enabled.

Port Number

Used to set the number of the serial port that is to be used. The corresponding serial port number must be configured in the Ports dialog box of the Windows Control Panel. The value should match your port number and be between 1 and 4096.

If the Use Modem check box is selected, the Port Number field will be disabled and the Phone Number and Modem Initialization fields will be enabled.

Phone Number

When a modem is to be used, fill in this field with the phone number that the modem must call in order to reach the remote system. Supply all the digits including area code, etc. Do not include spaces.

If the Use Modem check box is not selected, the Phone Number field will be disabled.

Modem Initialization String

Used only if your modem requires a customized initialization string. In most situations, you will leave this field blank. If configured, any initialization string must be prefixed by "AT". Note that the initialization string will be used only by the Trihedral TSP driver. It will be ignored by the Unimodem driver.

Data Bits

Select a number (either 7, or 8) specifying the number of data bits to use for each character. The 8 bit setting, matching the size of a byte, is used for almost all devices. The 7 bit setting might be needed for older devices that require ASCII values.

Stop Bits

Specify the number of stop bits for each data character. (In asynchronous communications, a stop bit is a bit that indicates that a byte has just been transmitted).

Baud Rate

Set the baud rate (bits per second) at which data should be transmitted.

Parity

The Parity section enables data quality verification based on one of the following parity formats. It must be set to match your device configuration.

No Parity – select to suspend parity checking for this serial port. By default, the No Parity check box is selected.

Odd Parity – select to use the odd parity mode of parity checking in which the number of mark bits (1's) in the data is counted, and the parity bit is asserted or unasserted to obtain an odd number of mark bits.

Even Parity – select to use the even parity mode of parity checking in which the number of mark bits in the data is counted, and the parity bit is asserted or unasserted to obtain an even number of mark bits.

Space Parity – select to use the mark parity mode of parity checking in which the parity bit is unasserted.

Mark Parity – select to use the mark parity mode of parity checking in which the parity bit is asserted.

Echo

Select whether or not the transmitted data should be echoed in the received data.

If the Echo check box is selected, the driver should expect that the transmitted data will be echoed in the received data.

By default, the Echo check box is not selected.

RTS Keying

> Check this box to select whether the driver should assert RTS (Request to Send) flow control while data is being transmitted.
>
> RTS/CTS (Request to Send/Clear to Send) flow control is typically used with all high-speed modems, or modems that compress data.

Key Up Delay

> The Key Up Delay property sets the delay in seconds prior to data transmission. The Key Up Delay field is only enabled if the RTS Keying check box is selected.
>
> The default value for the Key Up Delay property is 0.1 seconds.

Related Information:

Modem Tags – Details about configuring Modem tags.

See: SerPortDisconnectDelay in the VTScada Admin Guide. Port Sharing: Multiple drivers can share a Serial Port tag when it is in 'serial' mode.

Serial Port Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
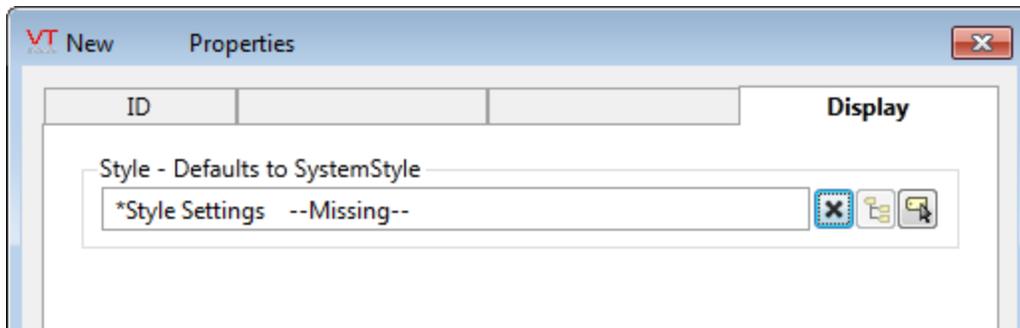
Related:

Style Settings Tags

Serial Port Tag Widgets

The following widgets are available to display information about your application's Serial Port tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Indicator Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...Gradient Color Change Widget

...LCD 7 Segment Widget

...LCD 5x7 Matrix Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

TCP/IP Port Tags

Not counted towards your tag license limit.
The TCP/IP port type is used to connect your VTScada application to a series of hosts, allowing you to send and receive data across a network or over the Internet. TCP/IP port tags can also open a TCP/IP socket to emulate a Serial Port tag for use with serial-based I/O drivers using TCP/IP-based terminal servers.

Reference Notes:

A TCP/IP port tag may also be used to accept incoming TCP/IP socket connections if a port number is defined, but a TCP/IP address is not defined (null or invalid), thus allowing drivers to act as slaves on network connections. Use caution: unintended consequences may result if the port with no address takes communications meant for other TCP/IP ports.

In the unusual event that you have multiple devices attempting to use the same channel at the same time across multiple ports, you might consider adding Comm Link Sequencer Tags to serialize requests.

A valid TCP/IP and UDP/IP port tags can have any of three possible values:

- 0 means that the port is good.
- 0X103 means that the port did not connect.
- 0x104 means that the connection was lost.

Related Information:

Comm Link Sequencer Tags – Serialize requests for the same communication channel across ports.

TCP/IP Port Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

TCP/IP Port Properties: Connection Tab

The Connection tab of the TCP/IP port tag properties folder contains properties that enable communications.
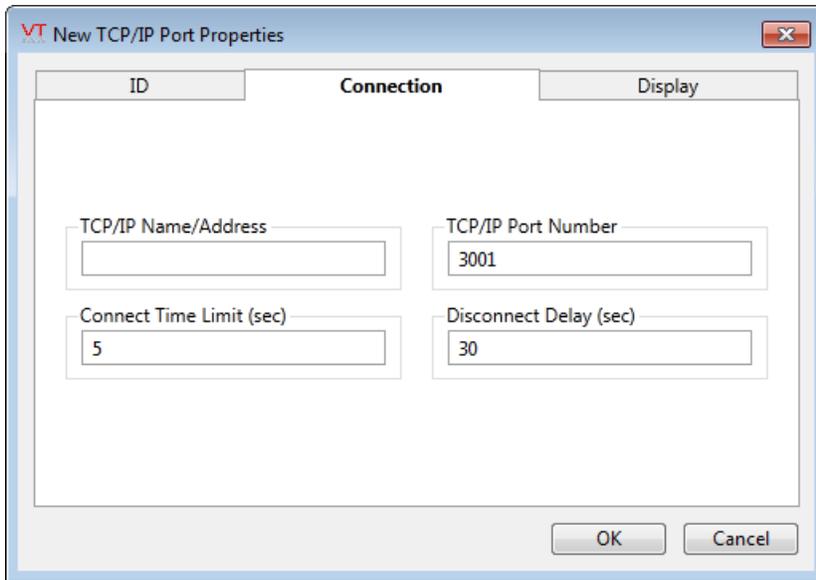
TCP/IP Name/Address

The TCP/IP Name/Address field provides a space for you to identify the name or IP address of the server to which to connect (for example, MyRTU.com, or 198.255.32.1).

Connect Time Limit (sec)

Set the number of seconds that the system should wait for a connection response from the server that is specified in the TCP/IP Name/Address field.

TCP/IP Port Number

The TCP/IP Port Number field refers to the port number on the host address through which communications are enabled. This information should be available from your hardware specification.
The following drivers have standard port numbers, although individual instances may vary.

Driver	Port

Allen-Bradley SLC	2222
Allen-Bradley Controllogix	44818
Siemens PLC	102
OpenModbus TCP	502
CIP	44818

A TCP/IP port tag may also be used to accept incoming TCP/IP socket connections if a port number is defined, but a TCP/IP address is not defined (null or invalid), thus allowing drivers to act as slaves on network connections. Use caution: unintended consequences may result if the port with no address takes communications meant for other TCP/IP ports.

Disconnect Delay

Specify in seconds or fractions of a second, the amount of idle time that should pass before the connection to the server is terminated.

TCP/IP Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
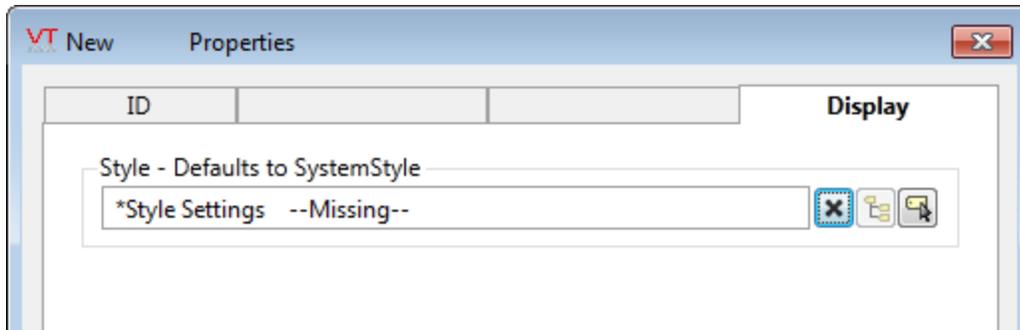
Related:

Style Settings Tags

TCP/IP Port Tag Widgets

The following widget is available to display information about your application's TCP/IP port tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Indicator Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...Gradient Color Change Widget

...LCD 7 Segment Widget

...LCD 5x7 Matrix Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

UDP/IP Port Tags

Not counted towards your tag license limit.

The UDP/IP port type can be substituted for a TCP/IP port tag for a driver to use the User Datagram Protocol (UDP). This protocol differs from TCP in that it does not provide for error checking, but does provide good support for packet broadcasting and multi-casting.

In the unusual event that you have multiple devices attempting to use the same channel at the same time across multiple ports, you might consider adding Comm Link Sequencer Tags to serialize requests.

Warnings:

All UDP/IP port tags use an internal listener port to receive datagram responses. The is what the Local Port Number defines (it is also used as the transmit port number).

The engine will automatically route responses back to the UDP socket stream if the remote IP address matches that of the stream. This means that multiple UDP/IP port tags can make use of the same local port number (internally they share the UDP port listener). This UDP port listener requires that the port be free, thus any conflict for use of the port runs the risk of disrupting UDP/IP communications.

Take note of the following possible errors. Some may not cause problems until the next re-start of the application.

1) UDP/IP Port and IP Network Listener tag sharing the same port number.

The IP Network Listener tag uses a socket server to function. This can interfere with the UDP listener port used by the UDP/IP port tag. The VTScada engine will already listen for UDP/IP datagram packets and place them into existing UDP streams if the remote IP address matches that of the UDP stream.

In general, do not configure an IP Network Listener tag with a UDP port unless you are certain that one is required. If you are certain, then ensure that they do not share the same port number. See also, warnings 2 and 3.

2) UDP/IP address left blank

The UDP/IP port tag functions as a socket server when the Name/Address field is invalid (blank). It will listen on the local port number for any datagram messages and will bind to the first one received. In this way it is similar to an IP Network Listener, only without the discrimination/filtering abilities. Note that you can have only one listener in the system, whether that be a UDP port configured to listen or a IP Network Listener.

If you are unsure of the remote address then enter 0.0.0.0 rather than leaving the field blank.

3) Disconnect Delay defined.

Do not define unless you are using an IP Network Listener in combination with the UDP/IP port. Do not use an IP Network Listener with a UDP/IP port unless you are certain that one is required.

This option was included exclusively for use with IP Network Listener tags. There is one situation where the engine-level UDP handling is insufficient; when the remote device can respond with different IP addresses over time. Use the Disconnect Delay to disassociate the remote IP after a period of inactively. The VTScada engine is then able to establish new remote IP address associations with the remote device via IP Network Listener tags. It is extremely unlikely that there would be a reason to use an IP Network Listener with a UDP port unless this is the case.

Related Information:

Comm Link Sequencer Tags – Serialize requests for the same communication channel across ports.

UDP/IP Port Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field.

You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

UDP/IP Port Properties: Connection Tab

The Connection tab of the UDP/IP port tag properties folder contains properties that enable communications.
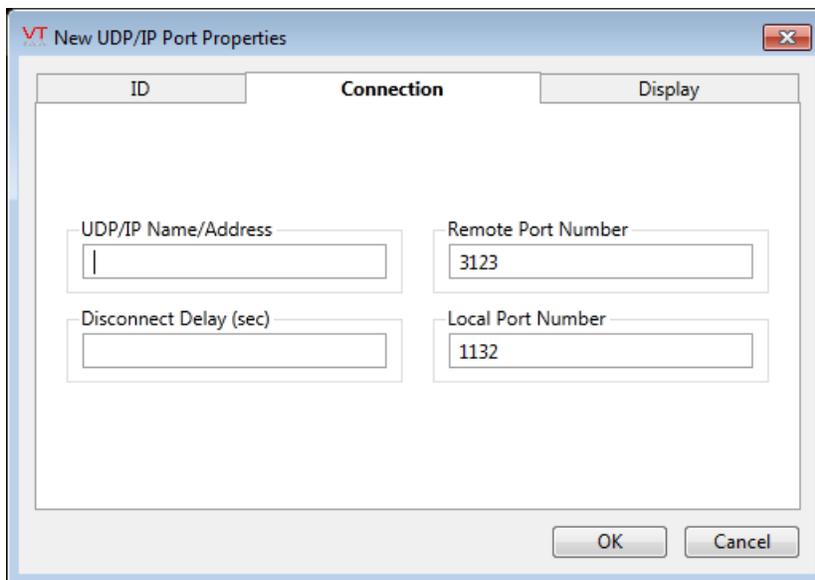
UDP/IP Name/Address

The UDP/IP Name/Address field provides a space for the name or IP address of the server to connect to (for example, MyRTU.-com, or 198.255.32.1).

If left blank, the UDP/IP port will function as a socket server. It is important that you read and understand the warnings in the

introduction section of this article before choosing to leave this field blank.

Remote Port Number

Specifies the port number on the host address through which communications are enabled.

Local Port Number

Sets the port number on which the local PC should listen.

Disconnect Delay

Do not configure unless you have determined that an IP Network Listener is required and you are absolutely certain that a disconnect is required due to the RTU using differing IP addresses.
Specify in seconds or fractions of a second, the amount of idle time that should pass before the connection to the server is terminated. To be used only when the IP address of the remote device changes between each transmission, thus allowing a new connection.

UDP Properties: Display Tab
When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
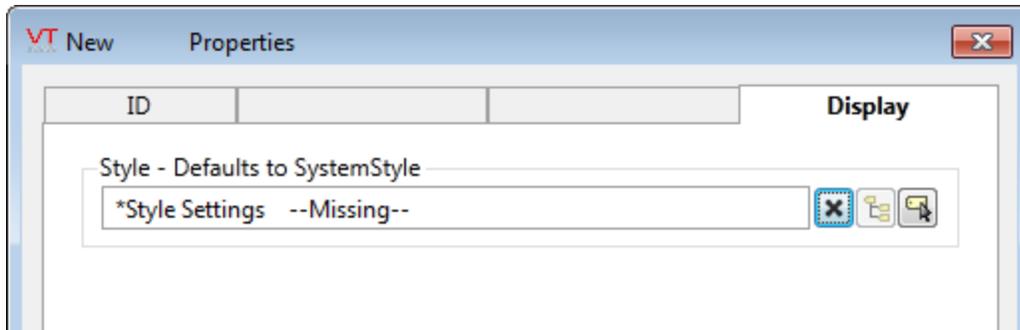
Related:

Style Settings Tags

UDP/IP Port Tag Widgets

The following widgets are available to display information about your application's UDP/IP Port tags:

…Alarm Priority Box

…Alarm Priority Icon

…Comm Indicator Widget

…Color Line Widget

…Comm Indicator Widget

…Equipment / Status Color Indicator

…Indicator Light Widget

…Gradient Color Change Widget

…LCD 7 Segment Widget

…LCD 5x7 Matrix Widget

…Multi-Color Widget

…Multi-Text Widget

…Numeric Value Widget

…Plot Data Widget

Communication Driver Tags

Communication driver tags are designed to translate data being transmitted to and from specific types of I/O devices, such as PLCs and RTUs. Each brand of I/O device generally requires its own communication driver, depending upon the communication protocol used by the device. Details for address formatting follow each driver description.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Communication Driver Tags Included with VTS:	
Allen-Bradley Driver Tags	CalAmp Diagnostic Driver Tags
CIP Driver Tags	Data Flow RTU Driver Tags
DDE Client Tags	DNP3 Driver Tags
Driver Multiplexer Tags	Enron Modbus Driver Tags
Fisher ROC Driver Tags	GE Series 90 Driver Tags
Koyo Driver Tags	MDS Diagnostic Driver Tags
Modicon - See Modbus	Modbus Compatible Device Driver Tags
Motorola ACE Driver Tags	Omron Host Link Driver Tags
Omron FINS Driver Tags	OPC Client Driver Tags (and, OPC Server Setup Tags)
Polling Driver Tags	Siemens S7 Driver Tags
SNMP Driver Tags	SQL Data Query Driver Tag

Note: Trihedral Engineering Limited has many other communication drivers available. Refer to the VTScada driver library on the Trihedral web site for further information

The basic driver types allow VTScada to provide an interface to:

- Physical I/O devices, such as programmable logic controllers (PLCs), RTUs, and I/O boards.

- Windows system features, such as Dynamic Data Exchange (DDE).

Related Information:

...Communication Driver Log-Enabled Variables - Describes the variables that may be logged for your driver.

...Data Suffixes for Tag I/O Addressing - Common strings to control how I/O data is interpreted.

...Communication Driver Alarms - Example and instructions to create an alarm on any of the log-enabled variables.

VTScada I/O Device Driver Library

The following communication driver tags are standard to VTS:
Allen-Bradley Driver
CalAmp Diagnostic Driver
CIP Driver
Data Flow RTU
DDE Client
DNP3
Enron Modbus
Fischer ROC
GE Series 90
Koyo
MDS Diagnostic Driver
Modbus Compatible Device
Modbus Plus
Motorola ACE
Omron FINS
Omron Host Link
OPC Client and Server
Siemens S7
SNMP

Company	Device	Medium	Protocol	Additional Components

Allen-Bradley	Ethernet Direct			
	PLC-2	Serial	DF1	Allen-Bradley Serial Interface Module
	PLC-3	DH+	DF1	Woodhead 5136-SD (ISA or PCI)
		Serial	DF1	
	PLC-5	Serial	DF1	
		DH+	DF1	Woodhead 5136-SD (ISA or PCI)
		Ethernet	DF1	PLC-5E only
	SLC500 (all processors)	Serial	DF1	
		DH+	DF1	Woodhead 5136-SD (ISA or PCI)
		Ethernet	DF1	
	Micrologix	Serial	DF1	
		RS-485/RS-232 Serial	Modbus RTU	
		Ethernet	DF1, CIP	
	Compact Logix	Serial	DF1	
		Ethernet	DF1, CIP	
	Control Logix	Serial	DF1	
		Ethernet	DF1, CIP	
Aquatrol	1300 RTU	Serial	Aquatrol	Special com-

			munication hardware	
	1500 RTU	Serial	Aquatrol	
Bailey	Network 90 DCS	Serial	Bailey	Bailey CIU
Campbell Scientific	10X Data Logger	Serial	Array Based	
	510 Data Logger	Serial	Array Based	
	Logger	Various	API	
Control Microsystems	Telesafe Micro 16 RTU	Serial	Modbus RTU, Modbus ASCII, DF1, DNP	DF1 & DNP are Telesafe options
	SCADAPack RTU Family	Serial/Ethernet	Modbus RTU, Modbus ASCII, Modbus TCP/IP & UDP/IP, Open Modbus TCP, DNP3 Datalogging and Store & Forward Supported	
DAQ		Serial	DNP 3.0 Level 1	
Data Flow	RTU	Serial, Eth-	Data Flow	Includes TACII,

Systems		ernet (TCP/IP& UDP/IP)	Systems	HyperTAC, HyperServer Database Conversions
DDE (Dynamic Data Exchange)	DDE Compliant Devices		DDE Client	DDE Server
DNP 3.0	DNP 3.0 Compliant Devices	Serial, Ethernet	DNP 3.0 Level 1	
Federal Pioneer	DSM-2P Power Meters	Serial	Federal Pioneer	
Foxboro	731C Loop Controller	Serial	Foxboro	
	743C Loop Controller	Serial	Foxboro	
	761C Loop Controller	Serial	Foxboro	
	C50 RTU	Serial	DNP 3.0 level 1	
General Electric	Series 1 PLC	Serial	General Electric	
	Series 5 PLC	Serial	General Electric	
	Series 6 PLC	Serial	General Electric	
	Series 90 PLC	Serial	SNP, SNPX	
Granville	307 Vacuum	Serial	Granville	

Phillips	Controller		Phillips	
Hewlett-Packard	HP48000 RTU	Serial	Subset, HEX-ASCII	
IDEC	Micro CP12	Serial	IDEC	
	Micro CP13	Serial	IDEC	
	Micro FA2	Serial	IDEC	
	Micro FA3	Serial	IDEC	
IEC	IEC 60870 Compliant Devices	Serial, Ethernet	IEC 60870-5-101	
Koyo	DL Family of PLCs (205, 305, 405)	Serial	Direct Net	
Landis & Gyr	Telegyr 8979	Serial	Landis & Gyr	
Lantronix	UDS-10, UDS-100 Serial to Ethernet Converters	Serial/Ethernet	N/A	
Mitsubishi	A Series	Serial	Mitsubishi	
Modbus	Modbus Compliant Devices	Serial, Ethernet, TCP, Ethernet UDP, Modbus Plus	Modbus ASCII, Modbus TCP/IP, Modbus UDP/IP, Open Modbus TCP, Modbus	Modbus PLUS requires additional hardware

			PLUS, Modbus RTU	
Modicon	Premium, Quantum, Momentum, Compact, Ultra-Compact, TSX Micro	Serial, Ethernet, TCP, Ethernet UDP, Modbus Plus	Modbus ASCII, Modbus TCP/IP, Modbus UDP/IP, Open Modbus TCP, Modbus PLUS, Modbus RTU	Modbus PLUS requires additional hardware
Moore	APACS ACM 040-4M	Moore	Moore	Requires Moore interface card for PC
Motorola	Moscad	Serial, Ethernet	DDE, Modbus	
OSI Soft	PI Database (Read Only)	API	API	Requires PI Client Software
Omron	C-series Programmable Controllers	Serial	Hostlink	
	CS Series PLCs	Serial	FINS	
	CJ Series PLCs	Serial	FINS	
OPC (OLE for Process Control)	OPC Servers for various Products	Serial, Ethernet	OPC Client	
Opto 22	Optomux	Serial	Optomux	

Power Measurement Ltd	ION Series Meters	Serial, Ethernet	ModbusDNP 3.0 level 1
Powers Process Controllers	330, 350, 512, 535, 545	Serial	Powers
Quindar	SCADA Master	Serial, Ethernet (TCP/IP & DECNet)	QNet
Schlumberger	Q1000 meter	Serial	DNP 3.0 level 1

Siemens   S5 Series PLC

S71200	Ethernet	Profinet or PG	
	Serial	Modbus RTU	
SIMATIC TI 305, 405, 505 Series PLCs	Serial	Direct Net	
S7-200 Series PLC	Ethernet	OP/PG	Requires Siemens Ethernet Communications Processor
	OPC over Serial, OPC over Ethernet	OPC Client	Requires OPC Server
S7-300 Series PLC	Ethernet	OP/PG	May Require Siemens Ethernet Communications Processor
	OPC over Serial, OPC over Ethernet	OPC Client	Requires OPC Server
S7-400 Series PLC	Ethernet	OP/PG	Requires Siemens Ethernet Communications Processor
	OPC over	OPC Client	Requires OPC

		Serial, OPC over Ethernet	Server	
Sierra Misco	see Stevens Water			
Square D	Sy/Max PLCs	Serial	Square D	
Stevens Water	SM0850	Serial	Stevens	
Surfline	9009, 9015 Series	Serial	Surfline	Requires Surfline CP Module
Texas Instruments	TI 305, 405, 505 Series PLCs	Serial	Direct Net	
Toshiba	T2, T2E PLCs	Serial	Toshiba	

Communication Driver Log-Enabled Variables

All of the standard drivers have 7 variables that may be logged. When recorded in a report, these allow you to monitor driver communications over time to spot problems or ensure that response times are acceptable.

Logged Variable	Description
1. ErrorValue	An error value or code associated with a driver's communication error.
2. FailedCount	Incremented on each communication error.
3. FailedRetryCount	Incremented if failure occurs on a retry. Drivers will usually retry more than once before designating an error to be an "error on retry". Thus, this count will always be less than or equal to the FailedCount.
4. SuccessCount	Count for successful reads and writes, incremented on each successful operation.
5. Quality	A mathematically derived indication of communication error rates. Used to show the driver's overall "health". (1)

6. ResponseTime	The time it takes the PLC/RTU to receive a command, process it and send a response. (2)
7. ErrorAddress	If an address was associated with the error, it will be recorded here.

A publicly accessible variable, CommStatsTimeStamp, holds the most recent timestamp that was used to update the above 7 variables and write them to a DAT file.

To view errors, use the Reports Page. Two reports are standard for all communications drivers. The Driver Communication Error Detail Report and the Driver Communication Summary Report. Both are described in "Reports Page: Types of Reports".

You may also view these error variables by creating an expression in a Calculation tag. For example, given a driver named "MyDriver", the expression to view the quality variable would be [MyDriver]\Quality.

(1) Calculation of Quality:

Quality is an indication of a driver's ability to communicate without errors. The value ranges between 0 and 1 where the closer the quality is to 1 (100%) the fewer communication errors are happening. You will need to determine what an acceptable level of quality is for your situation. To put it simply, the closer the quality is to 1, the better the driver communication.

On an initial, successful communication with a device, a quality of 1 (or 100%) is recorded. If subsequent communications attempts end with errors, the level of quality drops according to the following formula. Successful communications serve to then raise the level of quality back to a maximum of 100%.

You may find that the quality calculation changes through the day. This might indicate interference from a source that can then be tracked down. The quality calculation formula:

```
Q = Q * qF + (1 - qF) * (1 - !Error)
```

Where:

    Q = Quality

    qF = Quality Factor

Error is a Boolean stating whether the driver reported an error state.

The default Quality Factor is .99 & all drivers start with a default Quality of 1.

Every time the driver completes an operation, the quality is recalculated. Thus, when it first starts and does a successful operation:

Q = 1 *.99 + .01*1 = 1

When an error occurs, Q = 1*.99 = .99

If a subsequent error occurs, Q = .99*.99 = .9801

If a subsequent success occurs, Q = .9801 *.99 + .01*1 = .980299

The quality value over time will reflect the drivers overall health. It should always be a value between 0 & 1.

Quality includes all errors including any low level errors, retries, or addressing errors.

(2) Response Time

Response Time is calculated with the formula:

ResponseTime = EndTime – StartTime – XmitTime – RcvTime

Where:

XmitTime is the time it took to transmit any data

RcvTime is the time it took to receive any data, based on

- the number of bytes sent,
- the baud rate,
- parity,
- data bits and stop bits,
- whether there is an echo expected,
- whether RTS key delays are used.

The resulting Response Time is the time it took the PLC/RTU to receive a command, process it and send a response.

Related Information:

...Calculation Properties: Calc Tab

Refer to the VTScada Operator's Guide for descriptions of the following reports:

...Driver Communication Error Detail Report

Data Suffixes for Tag I/O Addressing

The data type to read from or write to is usually implied by the address. If it is not, the strings shown in the following table may be appended to the address (without spaces) to force the data to be interpreted as a specific type:

Suffix	Meaning
/ABFloat	Allen-Bradley PLC/3 Floating Point (4 bytes) (Used for Allen-Bradley exclusively)
/AB5Float	Allen-Bradley PLC/5 Floating Point (4 bytes) (Used for Allen-Bradley exclusively)
/BCD2	2-digit (1 byte) Binary Coded Decimal
/BCD3	3-digit (2 bytes - lowest 12 bits) Binary Coded Decimal
/BCD4	4-digit (2 bytes) Binary Coded Decimal
/BCD8	8-digit (4 bytes) Binary Coded Decimal Note: while any driver may use this suffix to read data, only the Modbus Compatibly can use it to write data.
/Bit	Attempts to convert the value to a single bit. The bit number used is always 0. (ex: 40001/Bit)
/Double	IEEE Double Precision Floating Point (8 bytes)
/Float	IEEE Single Precision Floating Point (4 bytes)
/SByte	Signed Byte
/SDWord	Signed 32-bit Integer
/SWord	Signed 16-bit Integer
/UByte	Unsigned Byte
/UDWord	Unsigned 32-bit Integer
/UWord	Unsigned 16-bit Integer

For example:
40001/UDWord

Related Information:

Each I/O device uses a different addressing system. Details can be found in the following topics:

Allen-Bradley Driver I/O Addressing

CalAmp Driver I/O Addressing

CIP Driver I/O Addressing

DNP3 I/O Addressing

DriverMUX I/O Addressing

MDS Diagnostic Driver I/O Addressing

Modbus Compatible Device I/O Addressing

Modbus Plus PLC Addressing

Motorola ACE Driver I/O Addressing

Omron I/O Addressing

OPC Client Driver I/O Addressing

Siemens S7 Driver I/O Addressing

SNMP I/O Addressing

Workstation Status Driver I/O Addressing

Communication Driver Alarms

Using the information in the topic, Communication Driver Log-Enabled Variables, you can create Alarm tags to warn operators of a loss in communication or a decrease in communication quality.

Reference Notes:

The alarm tag will use an expression as its trigger, where the expression refers to one of the log-enabled variables in the communication driver. It is recommended that you create the alarm tag as a child of the driver tag.

Example:

An alarm to warn of driver quality falling below 0.95.

Expression: [*Driver]\Quality

This expression assumes that the Alarm tag was created as the child of a driver. If the alarm cannot refer to a parent driver, substitute the full name of the driver tag for "*Driver".

Shown configured in an alarm tag:
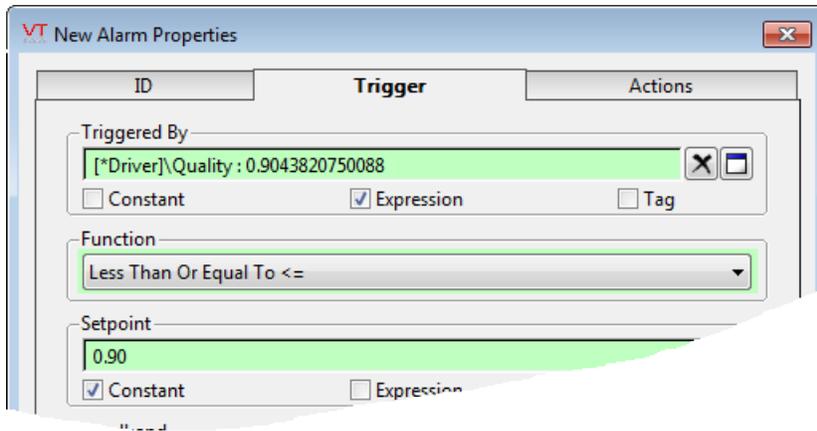
Related Information:

Communication Driver Log-Enabled Variables – Describes the variables that may be logged for your driver.

Allen-Bradley Driver Tags

Not counted towards your tag license limit.
The Allen-Bradley driver provides an interface to an Allen-Bradley PLC, allowing communication via one of:

- Serial port
- KT/SD I/O board
- TCP/IP

Note: Please refer to CIP Driver tags for information on Allen Bradley ControlLogix tags.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Allen Bradley devices have the ability to collect data between polls. To access this data with either a VTScada Analog Status or Digital Status tag, configure the "History Address" edit field with the location in the PLC of the logged data value using the format:

where "x" is the queue number (between 0 and 255) and "y" is the field in the queue where the data value is being logged (between 3 and 11 because 1 is reserved for the date and 2 for the time).

Example:

The PLC is configured to log values of addresses T4:5.ACC and T4:6.ACC into fields 3 and 4 of queue 0 once every minute.

To configure an Analog Status tag to read the current value of address T4:6.ACC from the PLC every 10 minutes and to read its logged values every hour, the following needs to be set in the "I/O" tab of the Analog Status tag's configuration:

> Address = T4:6.ACC,
> Scan Interval = 600,
> History Address = DL0:4,
> History Scan Interval = 3600

This driver also has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the [Type](#) tab to decide whether this feature should be used in your application.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Related Information:

Allen-Bradley Driver I/O Addressing - I/O address formats.
Communication Driver Log-Enabled Variables - Log Enabled Variables.

Important Notes

When configuring the data-logging feature on the PLC:

- Make sure that both date and time stamps are being logged into the relevant queues.

- Failure to configure a queue in this manner will result in the Allen Bradley Driver tag simply ignoring all records from that particular queue. This behavior is by design since a list of logged values without time stamps is of little value.
- Make sure that a real-time clock is present and enabled in the PLC.
- For the same reason as above, if a real-time clock in not present in the PLC or if it is present but not enabled, the driver tag will ignore all records from all data logging queues.
- Make sure the PLC's real-time clock configuration (date, time, time-zone, daylight savings time) matches the workstation's clock configuration.
- Failure to do so will result in the Allen Bradley Driver tag setting the wrong timestamps to all logged values read from the PLC.

Allen-Bradley Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Allen-Bradley Driver Properties: Type Tab

The Type tab for an Allen-Bradley driver tag properties folder consists of attributes used to identify the type of Allen-Bradley programmable logic controller that this driver will represent.
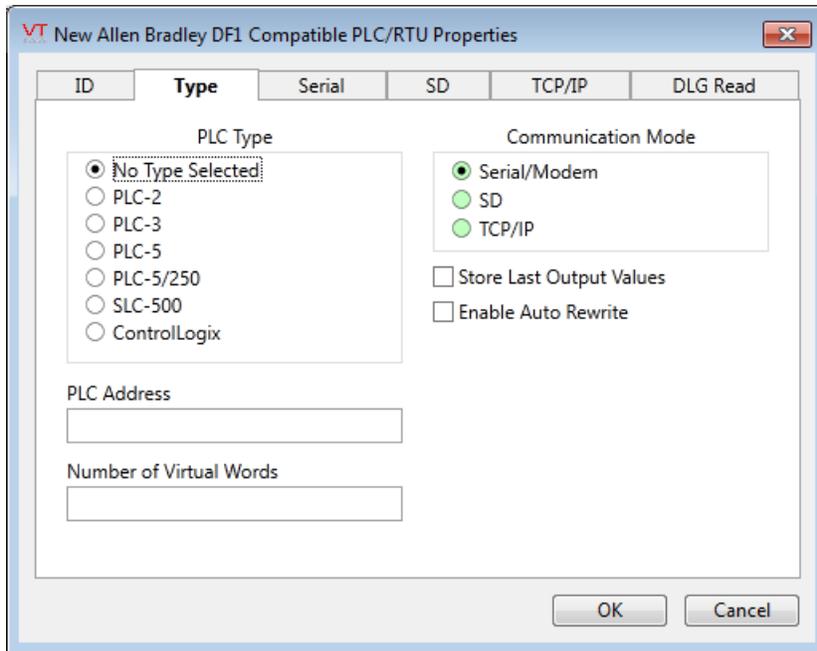
PLC Type

Select the type of Allen-Bradley PLC that this Allen-Bradley
driver tag will represent. This may be one of:

- No Type Selected <default option>
- PLC-2
- PLC-3
- PLC-5
- PLC-5/250
- SLC-500
- ControlLogix

Note: The ControlLogix driver is used in SLC-500 addressing.
The TCP/IP port number to use for this driver is 44818.
Note also: it may be necessary to adjust the timeout interval on
the TCP/IP tab if using ControlLogix.

Communication Mode

Set the mode to match the linked port.

PLC Address

This field should contain the address of the Allen-Bradley pro-
grammable logic controller. One of the following 3 formats is
permitted. (Note that this address must be provided in octal.)

- 012 (octal)
- 034 0357 (octal with a bridge address)
- 1.2.3.4 (IP address)
- 012 (octal)
- 034 0357 (octal with a bridge address)
- 1.2.3.4 (IP address)

The PLC Address field and the TCP/IP Address fields are linked,
and therefore are completed simultaneously. When you enter
an address in the PLC Address field, the same address is auto-
matically entered in the TCP/IP Address field on the TCP/IP tab
for this tag. Conversely, when you enter an address in the
TCP/IP Address field, the same address is automatically
entered in the PLC Address field.

Number of Virtual Words

Use this field to specify the number of registers that the virtual
PLC will create. The virtual PLC will then emulate a PLC-2 and
respond to requests to read and write data.

Store Last Output Values

When checked, the driver will maintain a record of the last
value written to each output address. This may be useful in at
least two situations:

- For hardware that does not maintain its state during a power loss
  and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would
  like to start that device with the values last written to the old one.

- If the last output values are stored, they may be re-written by either of two methods:
- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated.

This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.

Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Allen-Bradley Driver Properties: Serial Tab

The Serial tab for the Allen-Bradley driver tag properties folder consists of attributes used to identify and establish a connection the Serial Port tag that the Allen-Bradley driver should use to transmit data to the I/O device. The Serial tab is also used to select the communication protocol and error checking to be used in transmissions of data between the I/O device and the Allen-Bradley driver.
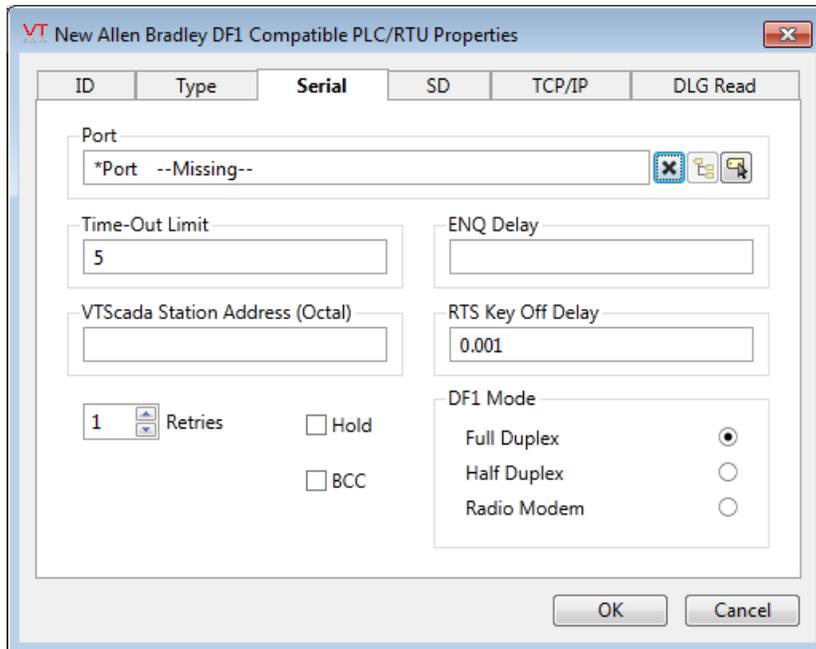
Port

Select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag. A Serial Port tag opens a serial port to enable communications between VTScada and your PLC or RTU. A TCP/IP tag is used to connect to a series of hosts, allowing you to transmit data across a network or over the Internet. Right-click on the selected tag to open its properties folder.

Time-Out Limit

Sets the receiver time-out limit (i.e. the time in seconds or fractions of a second that this driver should wait for a reply from the PLC or RTU).

VTScada Station Address (Octal)

Specify the address (using the octal addressing format) of the VTScada system on the serial link.

Retries

Select the number of attempts that will be made by this driver if there is no reply to a message, before an error is declared.

ENQ Delay

Specify the amount of time (in seconds or in fractions of a second) that this Allen-Bradley driver should wait before sending an ENQ in half duplex mode.
Note: Half duplex mode refers to the transmission of data in one direction at a time.

RTS Key Off Delay

The RTS Key Off Delay field represents the amount of time (in seconds) that this Allen Bradley driver will wait before dropping RTS at the end of a data transmission.

Hold

The Hold check box indicates whether this driver should hold data from the PLC or RTU in the event of a communications failure.

- If the Hold check box is selected, the last received value from the PLC or RTU will be held.
- If the Hold check box is not selected, the data will be invalidated in the event of a communications error.

By default, the Hold check box is not selected.

BCC

Click this check box to select whether the BCC error checking method will be used on communication messages.
By default, the BCC check box is not selected (i.e. CRC error checking is used on all communication messages).

Note: The CRC acronym stands for Cyclic Redundancy Check, and the BCC acronym stands for Block Check Character. Both are techniques used to detect data transmission errors. In general, CRC is a better error checking method, as it is a 16-bit error check, whereas BCC is an 8-bit error check.

DF1 Mode

Enables you to choose between the following data transmission modes in communications from the driver to the PLC.

- Half Duplex: Half duplex mode refers to the transmission of data in one direction at a time. If the driver and the PLC are communicating using the half duplex mode, either the driver or the PLC may transmit data at one time. The device not communicating must wait until the device that is transmitting data is finished before it can transmit its data.
- Full Duplex: Full duplex mode refers to the transmission of data by both parties simultaneously.
- Radio Modem: Certain Allen Bradley PLCs use the Radio Modem Protocol (RMP) to optimize DF1 communication on radio modem links. The protocol is an enhancement of the Full Duplex Protocol and uses the same message format with all the ACK and NAK responses removed.

Allen-Bradley Driver Properties: SD Tab

The SD tab for the Allen-Bradley driver tag properties folder consists of attributes used to identify and establish a connection to the SD I/O board, if one has been configured for this I/O device.
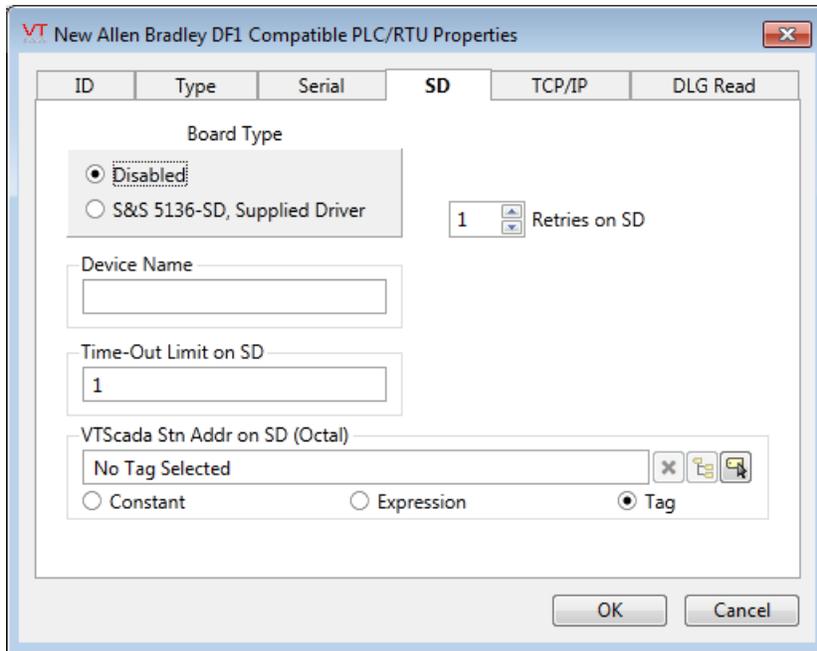
Board Type

Specify the type of I/O board that has been installed. This can
be one of:

- Disabled – When selected, indicates that the I/O board for this
  driver is disabled.
- S&S 5136-SD, Supplied Driver – When selected, indicates that a
  5136-SD card is installed.

Device Name

The name of the device being monitored by this tag. This is a
unique name that must be entered exactly as it appears in the
Windows® Device Manager's devices list.
Note: To launch the Device Manager in Windows, click Start,
and then click Control Panel. Double-click System. On the Hard-
ware tab, click Device Manager.

Time-Out Limit On SD

Set the amount of time (in seconds or in fractions of a second) that the driver will wait for a reply from the PLC's SD card.

Retries On SD

Select the number of retry attempts you wish to be made by the driver if there is no response to a message

VTScada Station Address On SD (Octal)

Specify the address of the VTScada system on the SD link. This value must be in Octal format.
The address may be set via a constant value, an expression or a numeric tag whose value is the octal address.

Allen-Bradley Driver Tag Type Properties: TCP/IP Tab

The TCP/IP tab of the Allen-Bradley driver tag properties folder can be used to configure the address, port, and time limit to be used if communications are to be made using the TCP/IP communications protocol.
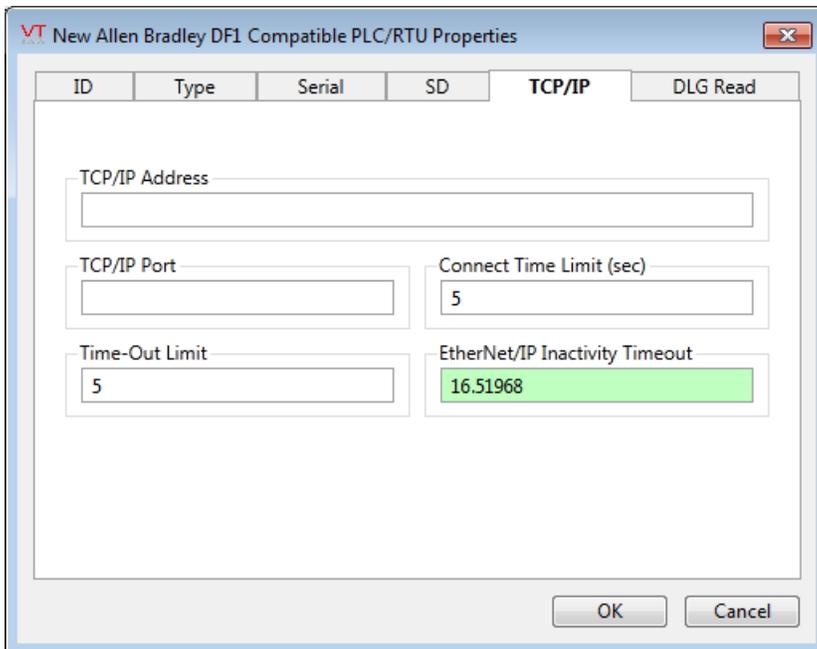
TCP/IP Address

The TCP/IP Address field is used to specify the port number through which this Allen-Bradley driver tag will communicate. Note: The TCP/IP Address field and the PLC Address fields are linked, and therefore are completed simultaneously. When you enter an address in the TCP/IP Address field, the same address is automatically entered in the PLC Address field on the Type tab for this tag. Conversely, when you enter an address in the PLC Address field, the same address is automatically entered in the TCP/IP Address field.

TCP/IP Port

The TCP/IP Port Number field refers to the port number on the host address through which communications are enabled. This information should be available from your hardware specification. The default value depends on the PLC Type selected.

Connect Time Limit

Use the Connect Time Limit field to set the amount of time (in seconds or fractions of a second) that the driver will await responses from the PLC. Defaults to 5 seconds unless otherwise specified.

Time-Out Limit

Specifies the time to wait for the PLC to respond once connected. This is the same as the timeout on the Serial tab.

Ethernet/IP Inactivity Timeout

Applies only if the PLC Type is set to ControlLogix. The maximum network link inactivity time in seconds. If there is no communication between VTScada and the PLC for this duration, VTScada will shut down the TCP/IP socket connection to

the PLC. This setting can be used to automatically terminate connections to PLC that are polled infrequently in order to reduce network traffic. The TCP/IP connection is automatically re-established by the driver when the PLC polling is resumed.

Allen-Bradley Driver Properties: DLG Tab

Certain Allen Bradley PLCs (of PLC Type "ControlLogix") are able to log time-stamped data into files in PLC memory.

The Allen Bradley Driver tag can read records from these files (referred to here and in the PLC documentation as queues). In order for the Allen Bradley Driver tag to make use of this feature, both the tag and the PLC must be configured properly. The DLG Read tab of the Allen Bradley Driver tag can be used to configure the tag for this feature. Refer to the PLC's manual for instructions on how to configure the PLC (noting that up to 256 queues can be created and data from different PLC memory addresses can be logged into different fields in different queues).
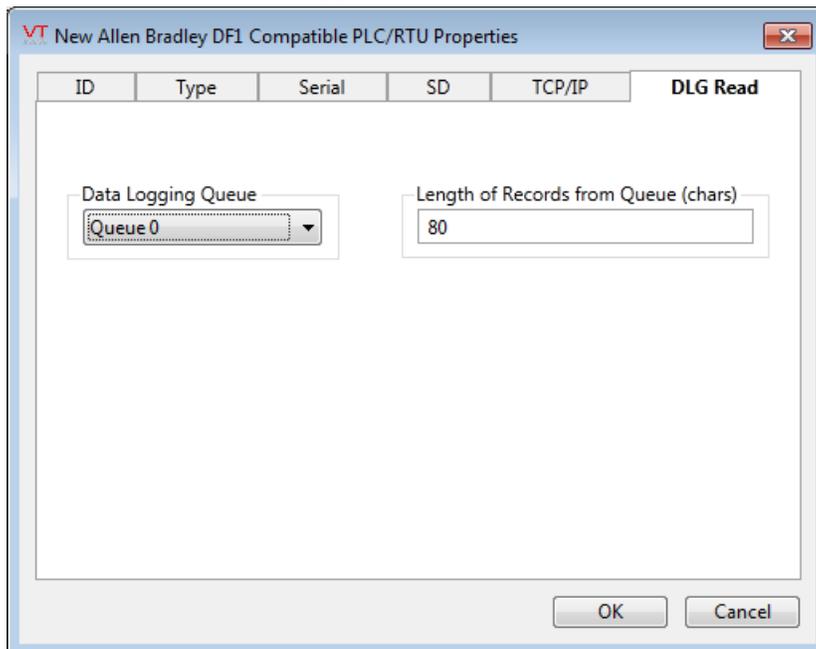
Data Logging Queue

Use to select a queue (configured according to the PLC's

manual) from which the driver will be retrieving logged data.

Length of Records from Queue

Enter the length of the records in the queue selected in the
Data Logging Queue drop list. This value is the length (in char-
acters) of the records in the queue once they are converted
into a string by the PLC for communication.
The valid range is from 1 to 80. The value must be calculated
according to how the data logging queue was configured in the
PLC itself:

- 11 characters for the date field,
- 9 characters for the time field,
- 12 characters for each long integer field
- 7 characters for each short integer field.

By default, the Allen Bradley Driver tag will assume a length of
80 chars (the maximum allowed by the PLC). While default
value will work, this may impose a penalty on the amount of
communication between the driver and the PLC. Entering the
exact length is therefore recommended.
Note: entering an incorrect value will most likely result in fail-
ure to read any records from a particular queue, therefore you
should only change the default value if you know the exact
value.

Allen-Bradley Driver Tag Widgets

The following widgets are available to display information about your
Allen-Bradley driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

Allen-Bradley Driver I/O Addressing

The following table identifies the value ranges for Allen-Bradley PLCs. Bit-wise addressing for binary and integer formats is written by following the element with "/bit-number", with the exception of timers which use a dot notation.
VTScada supports native device address formats. For example, in a PLC5 "T4:1.ACC"

Allen-Bradley Addressing	Value Range
Output Image	O:0 – O:30
Input Image	I:0 – I:30
Status	S:0 – S:n
Binary	B3:0 – B3:255 (bitwise – B3:1/15 16th bit in element number 1)
Timer	T4:0 – T4:255 (bitwise – use dot notation rather than a slash. For example: T4:100.DN to access the "done" bit.)
Counter	C5:0 – C5:255
Control	R6:0 – R6:255

Integer	N7:0 – N7:255 (bitwise – N7:1/5 6th bit)
Floating Point	F8:0 – F8:255
Network	x9:0 – x9:255
User Defined	x10:0 – x10:255
Data Logger Historical Reads	DLX:Y Where, X is the queue number and Y is the element number. The queue number can range between 0 and 255. The element number can range between 1 and 11.

Note: This addressing system is applicable for all Allen-Bradley PLCs, regardless as to the model. Where necessary (for example: ControlLogix), the PLC's processor takes care of mapping addresses internally.

CalAmp Diagnostic Driver Tags

Not counted towards your tag license limit.
The CalAmp diagnostic driver type enables users to read statistics from any given Integra-TR, Integra-H, Viper SC or Phantom II radio.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

This driver was known as the Dataradio Diagnostic driver in earlier versions of VTScada.
These tags are required only in applications where you wish to gather data about a CalAmp unit.
For example, a common arrangement might be a CalAmp unit connected to your PC, and a secondary CalAmp unit connected to an RTU or PLC at a remote location. In order to collect data from the RTU or PLC, your

VTScada application can use an appropriate communication driver tag such as Allen-Bradley or Modicon. If you wish to gather data about either CalAmp units in such a scenario (e.g. the temperature of the CalAmp unit), your application will require a CalAmp diagnostic driver tag.
In the unusual event that you have multiple devices attempting to use the same channel at the same time across multiple ports, you might consider adding Comm Link Sequencer Tags to serialize requests.

Related Information:

CalAmp Driver I/O Addressing
Comm Link Sequencer Tags - Serialize requests for the same communication channel across ports.

CalAmp Diagnostic Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

CalAmp Diagnostic Driver Properties: Type Tab

The Type tab of the CalAmp diagnostic driver tag properties folder is used to select the type of CalAmp unit that this tag is to represent. The choices, as shown include:

- Integra-TR
- Integra-H

- Viper SC
- Phantom II

CalAmp Diagnostic Driver Properties: Serial Tab

The Serial tab of the CalAmp diagnostic driver tag properties folder is used to select the port to which the CalAmp unit associated with this tag is connected.

The port on the Data Radio unit, from which VTScada collects diagnostic information, is a true RS-232 serial port. In order to connect to this port with VTScada, you can use one of several different hardware solutions:

- Serial port cable connected directly.
  For this case, configure and select a Serial Port tag.

- Stand alone serial modem connected to the serial port on the radio.
  VTScada has its own modem that is used to dial up the modem connected to the radio. For this option, configure and select a serial port that is using a modem.

- Ethernet terminal server connected to the serial port on the radio.
  VTScada is connected to the same network as the terminal server and can connect to it with either a TCP/IP or UDP/IP port tag

- RJ-45 Ethernet port on the radio (Viper SC and Phantom II radios only).
  Any VTScada port tag may be used, but it must match the hardware connection to the radio.

CalAmp Diagnostic Driver Tag Widgets

The following widgets are available to display information about your application's CalAmp diagnostic driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

CalAmp Driver I/O Addressing

Data can be obtained by reading from predefined addresses. The radios generate an ASCII string containing the diagnostic information.
An analog or digital tag should be configured for each item of diagnostic information. The address structure depends on the type of radio.

- For Integra-TR and Integra-H, the address format is <radio ID>:<-parameter>.
  For example, to read supply voltage on radio 3, the address is "3:VOLT".

- For Viper SC and Phantom II radios, the address format depends on whether the radio is in ROUTER or BRIDGE mode.

  - ROUTER mode: [IP address]:

  - BRIDGE mode: [MAC address]:

  For example, with a radio in ROUTER mode, a valid address might be: "[192.168.6.254]:VOLT".

For Viper SC radios only, the source and destination radio addresses must be specified for the RRSSI and LRSSI parameters.
For example, a valid address could be "[192.168.1.205], [192.168.2.1]:RRSSI"
Where the source radio is 192.168.1.205 and the destination radio is 192.168.1.2

The following parameters are available:
All radios:

TEMP	Internal radio temperature, measured in degrees C.
VOLT	The radio supply voltage.
FPOW	The forward power, measured in Watts.
RPOW	The reverse power, measured in Watts.

Integra-TR and Integra-H radios:

RRSSI	The remote received signal strength, measured in dBm.
LRSSI	The local received signal strength, measured in dBm.
GRX	The number of good data blocks received out of the last 15.
TRX	The total number of data blocks detected, up to a maximum o f 15.

Viper SC radios:

RRSSI	The remote received signal strength, measured in dBm.
LRSSI	The local received signal strength, measured in dBm.
PER	The packet error rate.

Phantom II radios:

BRSSI	The background received signal strength, measured in dBm.
LRSSI	The local received signal strength, measured in dBm.
PER	The packet error rate.
THIN	The packet thinning value

CIP Driver Tags

Not counted towards your tag license limit.

The CIP (Control and Information Protocol) driver type provides an interface between VTScada and hardware that uses the CIP or ENIP standards or both for communications.

Note: It has been reported that response time from the CIP driver has been improved by increasing the system overhead time slice setting within the RS Linx software from 20% to 70%. (Not in VTScada.) **Use this information with caution, taking your unique situation into account.**

> **Note:** Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Related Information:

CIP Driver I/O Addressing

When this driver is used in combination with a TCP/IP tag, the standard TCP port is 44818

Notes About Allen Bradley ControlLogix Tags

The ControlLogix family of PLCs use tag names to identify data objects in the PLC. The type of data object must be specified when writing to a particular tag. The ControlLogix family of PLCs supports the following atomic types:

- Boolean (BOOL)
- 8-bit Signed Integer (SINT)
- 8-bit Unsigned Integer (BYTE)
- 16-bit Signed Integer (INT)
- 32-bit Signed Integer (DINT)
- 32-bit Unsigned Integer (DWORD)
- 32-bit Float (REAL)

In addition, 2-dimensional arrays are supported, thus an array of any type listed above is possible:

- DINT[32] – a 32-element array of DINTs.

The ControlLogix family of PLCs also supports a structure. A structure is a collection of the following atomic types. For example, a Timer has the following fields:

- Timer.PRE – DINT
- Timer.ACC – DINT
- Timer.EN – BOOL
- Timer.TT – BOOL

- Timer.DN – BOOL

- Timer.FS – BOOL

- Timer.LS – BOOL

- Timer.OV – BOOL

- Timer.ER – BOOL

The VTScada driver can read and write to all of the atomic data types within the structure. The VTScada driver can also read a stream representation of a structure; it does not have any knowledge of the contents of the structure. At present, the driver cannot write an entire source.

It should be noted that even the atomic data types are themselves structures (of a special sort that can be read and written). For example, an INT is actually an array of 16 BOOLS, 1 representing each bit in the INT. It is possible to read and write a bit of an INT by addressing it correctly (i.e. INT_TAG1|INT will write to the least significant bit of INT_TAG.).

This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the Connection tab to decide whether this feature should be used in your application.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

CIP Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

CIP Driver Properties: Connection Tab

The Connection tab for a CIP driver enables you to configure the port through which communications with the hardware will take place.
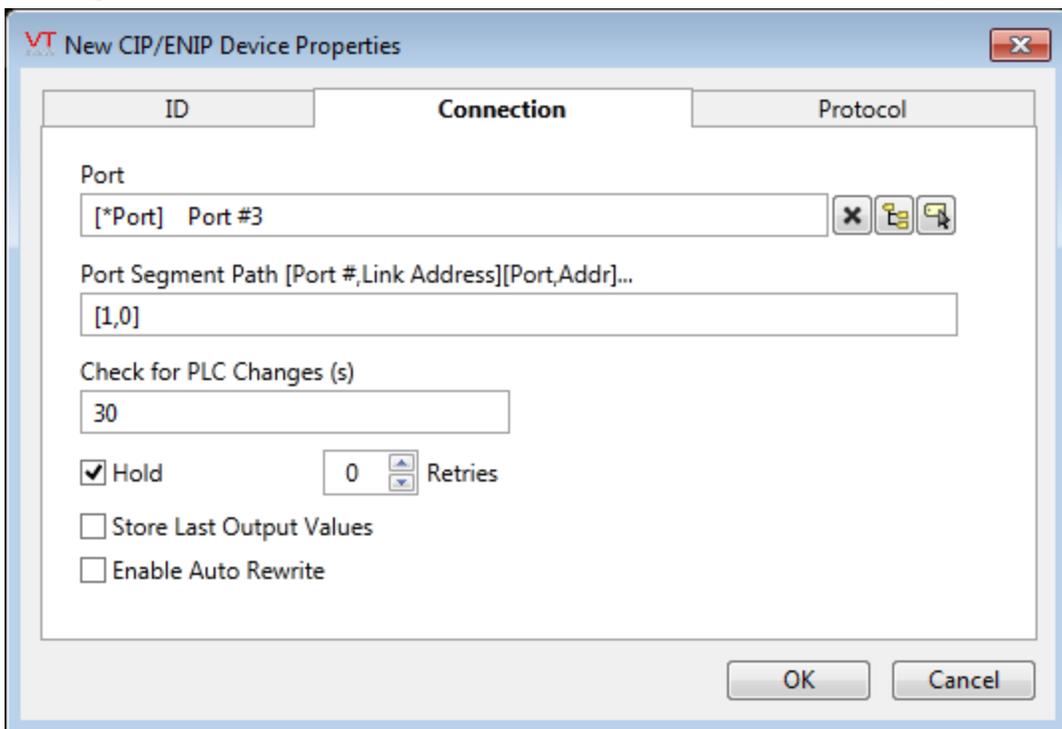
Note: The TCP/IP Port Tag should be usually be configured to communicate with the PLC on port 44818 (0xAF12). You should confirm this against your PLC's configuration.

Port

Select the TCP/IP tag that will provide the address and port to be used when communicating with the PLC.

Port Segment Path [Port #, Link Address][Port,Addr]...

The Port Segment Path format is [Port ID, Link Address], and describes the path to access the data.
   The Port ID is a number which describes the port to access the data. Port ID 1 is defined as the backplane of a PLC.
   The Link Address is the address of the item with which to connect. You may provide an IP address or URL for the link address portion of the Port Segment Path.
For example Link Address 0 is Slot 0 of the backplane. If simply connecting to an Ethernet port on a CPU card, this value parameter should be set to [1,0].
Further examples:
   [1,5]
   [1,5][2,18]
   [1,5][1,10.158.2.20]
   [1,5][1,cipplc.example.com]
You must be familiar with the hardware to which you are con-necting in order to correctly establish the Port Segment Path.

Check for PLC Changes (s)

The CIP driver reads the entire PLC tag configuration into memory before initiating any reads, allowing it to poll data by an index instead by PLC tag name. This results in considerably faster performance. The full read of the PLC tags names does take time, but is done only once and stored.
It is necessary to scan the PLC periodically to see if anything has changed. This is a single transaction that indicates whether or not another full read is required. The default of 30 seconds is recommended for most installations, but may be

extended if you are confident that the PLC configuration is not changing.

Operators may force a read at any time by using the "Read PLC Tags" button on the CIP Driver Command dialog.

Hold

The Hold check box controls whether this driver should hold data from the PLC or RTU in the event of a communications failure.

If the Hold check box is selected, the last received value from the PLC or RTU will be held. If the Hold check box is not selected, the data will be invalidated in the event of a communications error.

By default, the Hold check box is not selected.

Retries

The Retries spin box enables you to select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one. If the last output values are stored, they may be re-written by either of two methods:

- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated.

This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.

Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

CIP Driver Properties: Protocol Tab

The Protocol tab for a CIP driver enables you to configure settings related to operation timeouts.

Note: The properties are part of the communications protocol. You must be familiar with the CIP protocols in order to properly configure them. If the default values for these properties are enabling communications, don't change them.
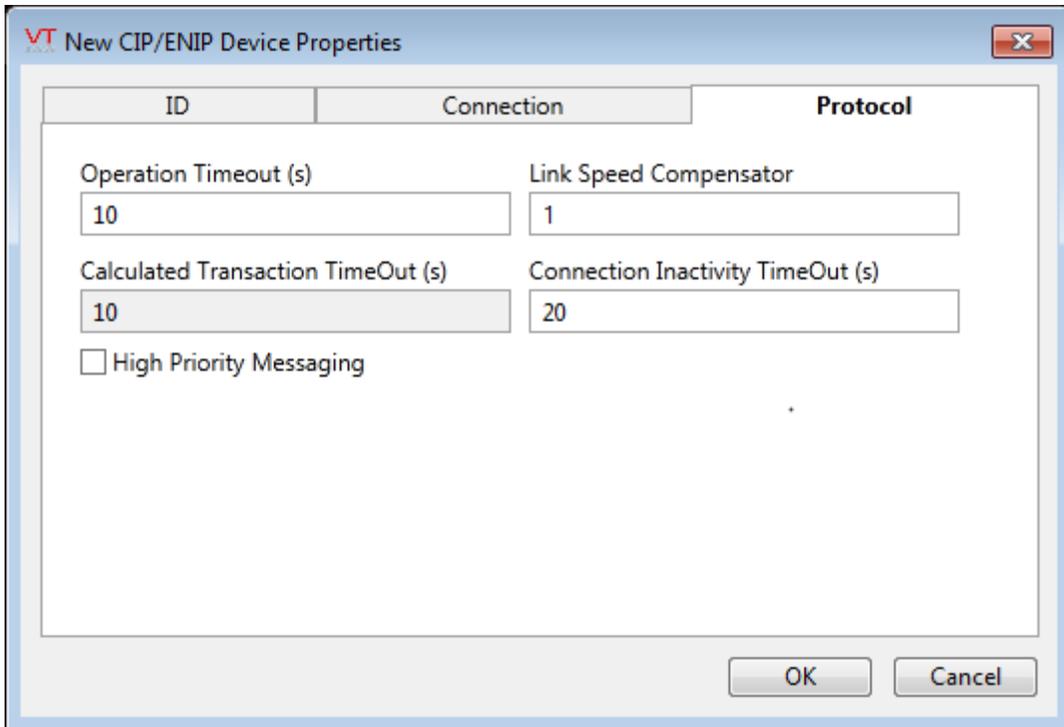
Operation Timeout

Use to specify the time (in seconds) after which the target will abort a request. The default value for Operation Timeout property is 10 (seconds).

Link Speed Compensator

VTScada has a timeout check that is similar to the Operational Timeout and matches the default value of that parameter. In a slow link environment, it may be necessary to make the VTScada timeout slightly longer to account for the transmission time on the slow link. The VTScada timeout will be the Operational Timeout parameter multiplied by the Link Speed Compensator.

Calculated Transaction Timeout(s)

This is the VTScada timeout value, calculated by multiplying

the PLC's Operational Timeout by the Link Speed Compensator.

Connection Inactivity Timeout (s)

The maximum network link inactivity time in seconds. If there is no communication between VTScada and the PLC for this duration, VTScada will close the connection to the PLC. This setting can be used to terminate connections to PLC that are polled infrequently in order to reduce network traffic. The connection is automatically re-established by the driver when the PLC polling is resumed. Note that the PLC's tag configuration must be re-read when the connection is re-established, therefore you must balance the cost of this increased traffic against the savings of closing the connection.

High Priority Message Flag

If the High Priority Message Flag check box is selected, the priority of messages is high and should take precedence over all else. Otherwise, the priority of messages will be normal.

CIP Driver tag Widgets
The following widgets are available to display information about your application's CIP Driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...CIP Control Widget

...CIP Information Widget

...CIP Statistics Widget

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

CIP Driver I/O Addressing

The VTScada CIP Address Select tool, found in every I/O tag that uses a CIP driver, will help you build addresses for your I/O tags.
If you have modified the address in a way that is not recognized by the driver, the address assist will not find that address. This applies both to addresses that include errors and to those integers and double integers that contain a bit specification (which is handled internally by the VTScada driver).

Address	Description	I/O String
Tag_Name	Reads the tag matching the provided name	Tag_Name
Tag_Name[n]	Reads element n from array Tag_Name	Tag_Name(n)
Tag_Name[n,m] or Tag_Name[n][m], etc.	Reads element m from 2D Array Tag_Name	Tag_Name(n)m
Tag_Name.Attribute	Reads Attribute from structure Tag_Name	Tag_Name Attribute
Tag_Name[n].Attribute	Reads Attribute from structure in Array Tag_Name, Element n	Tag_Name(n) Attribute

Examples:

Address	Description
SetPoint	Read the value of the tag SetPoint.
Analog_Data[4]	Read the value of element 4 of the Analog Data Array.
Timers[6].PRE	Read the value of the Preset Attribute of the 6th timer in the Timers Array.
Inputs[5,9]	Read the 9th Element of the Inputs 2 dimensional array.

CIP Driver tag Modifiers:

Earlier versions of VTScada / VTS allowed the use of tag modifiers. These are now ignored. There is no need to change existing tags that have modifiers.

Bit Addressing

You can specify a bit number in integer and double integer addresses. If you choose to use this format, note that you cannot subsequently use the address assist dialog to find that address.
To read bit number four of an element in a double-int array, add a .4 after the address. For example, DINT_Array[n].4 where [n] specifies the number of the element in the array.
Note that you cannot use this format for Boolean array elements. Everything after the closing bracket of the array element number will be ignored.

CIP Program addressing:

While VTScada normally addresses CIP controller tags, VTScada I/O tags that will use programs in the PLC can access those programs with the following address format:

```
PROGRAM:ProgramName.tagname
```

Where, "PROGRAM:" is a keyword, "ProgramName" is the name of the program in the PLC and "tagname" is the PLC tag name.

Data Flow RTU Driver Tags

Not counted towards your tag license limit.

Data Flow RTU drivers are used to provide an interface to a Data Flow RTU. Each Data Flow RTU driver provides for up to 18 installed Data Flow modules.

> Note: Data Flow RTU driver tags do not require polling drivers since they have their own polling features.

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Operators who do not have configuration privileges, but do have either "tag modify" or "manual data" security privileges can toggle the active polling state on or off if this tag is drawn as a Comm Indicator on a page. When the operator right-clicks on the Comm Indicator for the driver, a context menu will open, allowing the Active status to be changed.

When data is written to a Data Flow tag, it is immediately passed to the integrated Polling Driver tag and then on to the associated communication driver tag. Following each write, all I/O addresses will be read immediately so that feedback need not wait for the next polling cycle.

The Data Flow RTU driver can be represented on a Site Map.

Data Flow RTU Driver Properties: ID Tab

> Note: The Data Flow RTU Driver type includes one extra field on the ID tab: The Station number, also known as the RTU address.

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Data Flow RTU Driver Properties: I/O Tab

The I/O tab for a Data Flow RTU driver tag provides properties that can be used to establish a connection between this tag and the physical Data Flow RTU.
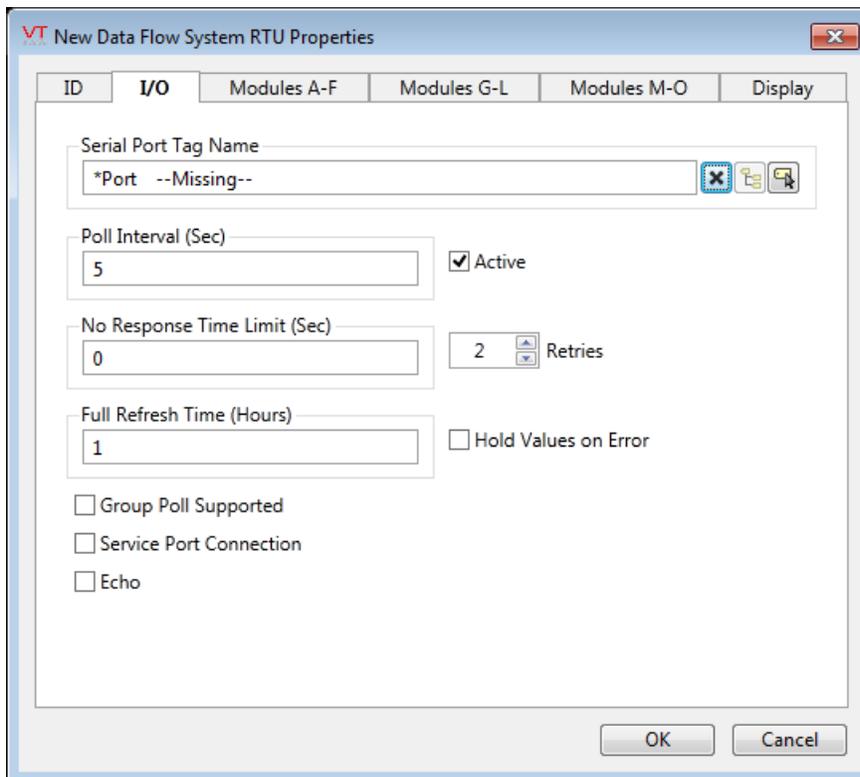
Serial Port Name

In order to connect to this port with VTScada, you can use one of several different hardware solutions:

- Serial port cable connected directly.
  For this case, configure and select a Serial Port tag.
- Stand alone serial modem connected to the serial port on the radio.
  VTScada has its own modem that is used to dial up the modem connected to the unit. For this option, configure and select a serial port that is using a modem.
- Ethernet terminal server connected to the serial port on the unit.

VTScada is connected to the same network as the terminal server and can connect to it with either a TCP/IP or UDP/IP port tag

Any type of VTScada port tag may be used, but it must match the hardware connection to the unit.

Poll Interval

Enables you to set the amount of time (in seconds or in fractions of a second) between polls of the physical Data Flow RTU.

Note: If the Active check box is not selected, the associated Data Flow RTU will not be polled, regardless of the interval assigned in the Poll Interval (Sec) field.

Active

Specify whether the designated poll interval is active or idle.

If the Active check box is selected, the associated Data Flow RTU will be polled according to the interval assigned in the Poll Interval field.

If the Active check box is not selected, the associated Data Flow RTU will not be polled, regardless of the interval assigned in the Poll Interval field.

The active status can also be toggled by the operator if the tag is drawn as a Comm Status indicator. Right-clicking on that widget will open a menu with which the operator can change the active status. When drawn as a SiteDraw, the outer ring will turn purple when the driver is not active.

No Response Time Limit

Set the amount of time (in seconds or in fractions of a second) that VTScada will wait when an attempt to poll the Data Flow RTU has been made.

If no communication is made, the system will attempt to poll the station again, according to the setting in the Retries spin box. If the system is unable to poll the Data Flow RTU, the next RTU in sequence will be polled.

Retries

Select a number indicating the amount of times that VTScada will attempt to poll an unresponsive Data Flow RTU, after which the system will move on to the next Data Flow RTU in the polling sequence.

Full Refresh Time

Enter a number indicating the amount of time, measured in hours that the system should wait before doing a full refresh of data values from the Data Flow RTU.

Hold Values on Error

Specify whether or not data should be held in the event of a communications error with the Data Flow RTU.

If the Hold Values on Error check box is selected, the last received value from the Data Flow RTU will be retained otherwise, the data will be invalidated on error.

Group Poll Supported

Select this box to specify whether or not the Data Flow RTU represented by this tag belongs to a poll group (if group polling is supported by the radio interface module).

Service Port Connection

If checked, The service port enables communication with the RTU via serial devices (such as Lantronix). It is an alternative to using the DataFlow RIM card radio. The DataFlow protocol is slightly different through the service port. Minor differences will be observed, such as all responses being marked as station '0' and the absence of CRCs on messages that would otherwise have them.

Echo

Used if the Service Port Connection option has been selected. Enable or disable according to whether your device transmits an echo for each transmit operation.

Data Flow RTU Driver Properties: Modules Tabs

The Modules tabs of the Data Flow RTU driver tag properties folder enables you to select existing (or create new) Data Flow module tags to represent the Data Flow modules installed for this RTU (see Data Flow Module Tags). You may associate up to 18 Data Flow module tags with one Data Flow RTU driver.

The modules are divided between the tabs, Modules A-F, Modules G-L and Modules M-0.
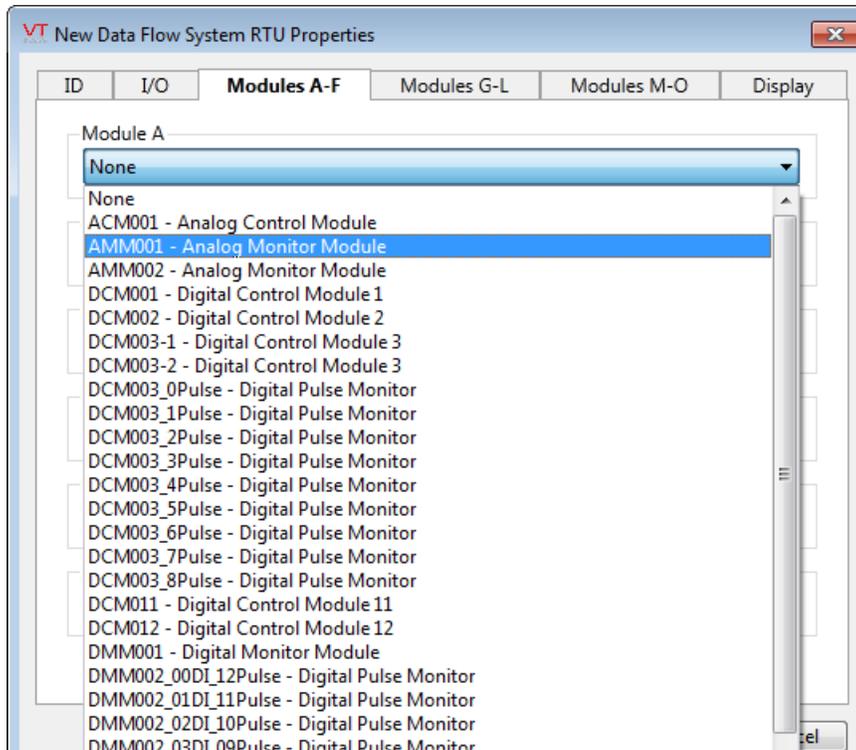
Module A, B, etc.

Use the Module fields to identify which module installed at a physical RTU by selecting a Data Flow module tags.

To associate this tag with a set of Data Flow module tags, click the drop-down field to select the appropriate Data Flow module tag for each module.

Data Flow RTU Driver Properties: Location Tab

The Location tab is used to define the placement (latitude and longitude) of the DataFlow station. Decimal values should be used rather than degrees, minutes and seconds.

You may find it easier to set the location using the map interface than to enter the latitude and longitude values here.

Latitude and Longitude

Holds location coordinates for this tag, thereby allowing it to be represented on a Site Map page.

Custom Details Page

If a custom details page has been created for this context tag, then that page should be selected here. If there is no custom details page, then operators will see the standard Site Details page upon clicking the pin for this tag in a Site Map.

Custom Map Icon

If a custom map icon widget has been defined, you can select it here. That icon will then be used instead of the standard pin when this tag is represented on a Site Map page. If there is no custom map icon, then the standard pin will be used.

Site List Display

Assuming that this tag has site properties, it will be included in the list of a Sites page. What will happen when an operator clicks on this tag in that list depends on the Site List display choice, and on whether any of this tag's children are I/O tags.

- Display as Site: A click will open the Site Details page as a pop-up.
- Display as Folder: A click will leave focus on the Sites page, but the list will now show the child tags of this site.
- Exclude: This tag should not be shown in the Sites page list.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).

Default zoom level.

Related Information:

Site Map

Data Flow System RTU Widgets

The following widgets are available to display information about your application's DataFlow RTU tags:

...Active Indicator Widget

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Data Age Widget

...DFS Station Draw

...Enable Polling Checkbox Widget

...Equipment / Status Color Indicator

...Fast Scan Widget

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Polled Station Widget

...Site Draw Widget

...Site Icon Widget

... Site Summary Widget

...Tag List Widget

...Toggle Polling Button Widget

DDE Client Tags

Not counted towards your tag license limit.
The DDE client enables tags to read from and write to a local or network DDE server.

Reference Notes:

The DDE acronym represents the term Dynamic Data Exchange; an inter-process communication system built into the Windows® operating systems. DDE enables two running applications to share the same data. For example, DDE makes it possible to insert a spreadsheet chart into a document created with a word processor; whenever the spreadsheet data changes, the chart in the document changes to match.

Note: The version bit of your VTScada installation (32-bit versus 64-bit) must match that of your operating system in order for DDE features to work.

DDE can also enable VTScada to accept data from another application, such as a chart, word processing document or spreadsheet. To do so, VTScada uses its DDE Client tag.

The application that supplies the data is called the server, while the program that accepts the data is called the client. VTScada can act as either a DDE client or a DDE server. The sections that follow provide instructions for configuring dynamic data exchange from an outside application to VTScada, and from VTScada to an outside application.

> Note: DDE is an older technology and may not be fully supported by modern operating systems or by newer features of VTScada.

DDE Client Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

DDE Client Properties: Local DDE Tab

The Local DDE tab of the DDE Client tag properties folder is used to set the application name for and a topic supported by the DDE server.
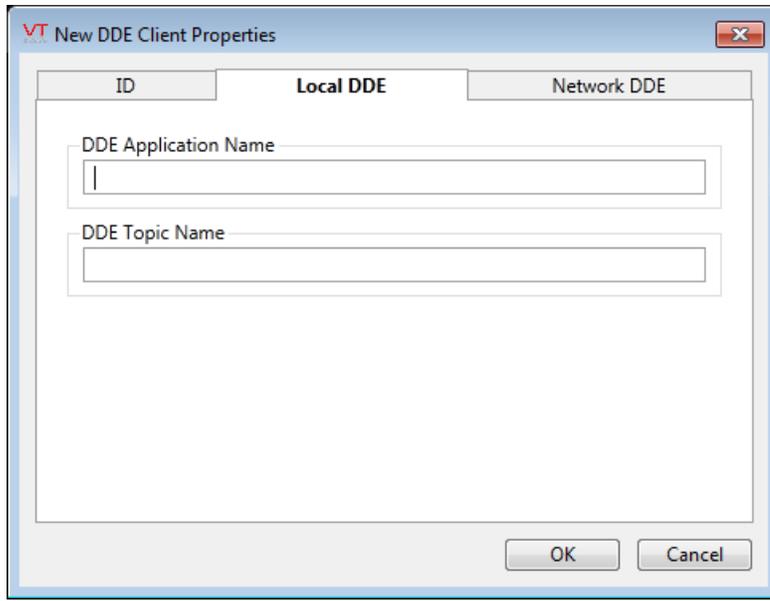
DDE Application Name

>   The DDE Application Name field enables you to enter the DDE program name. For example, "Excel".

DDE Topic Name

>   The DDE Topic Name field enables you to enter the name of a topic supported by the DDE server. If the application is Excel, this would be the name of the workbook file. For example: "MySpreadsheet.xlsx".

DDE Client Properties: Network DDE Tab

The Network DDE tab of the DDE Client tag properties folder is used to set the name of the computer on which the DDE server is running, and the share name that identifies the DDE server on the computer.

Note: DDEShare is based on NetDDE. For security reasons, Microsoft stopped distributing this product for versions of Windows following XP. If you choose, you may still be able to find and install NetDDE for your version of Windows. The instructions for doing so are beyond the scope of this document.
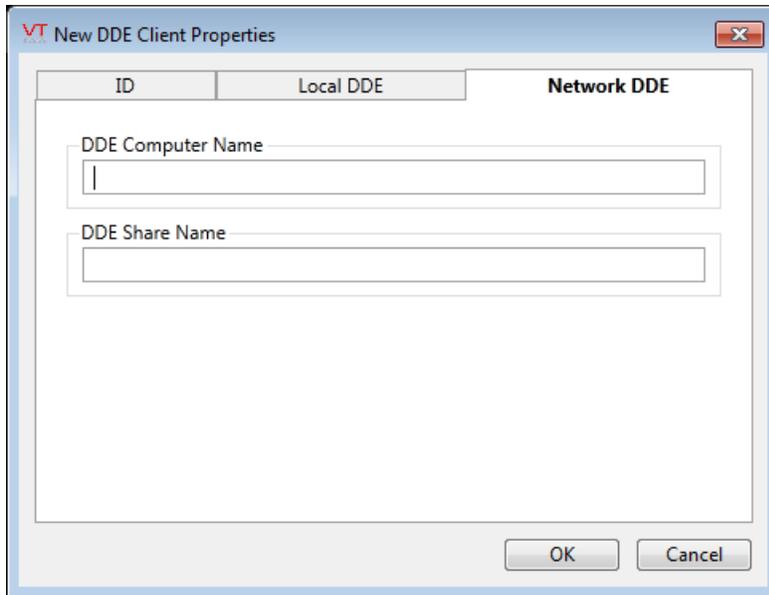
DDE Computer Name

The DDE Computer Name field enables you to enter the name of the server on which your DDE share (specified in the DDE Share Name field) was created (e.g. Station9).

DDE Share Name

The DDE Share Name field enables you to enter the name of the DDE share you created on the server specified in the DDE Computer Name field. You must add the .OLE extension to the name of the DDE share (e.g. DDEShare$.OLE).

Note: Unlike the Modicon, Allen-Bradley, and Omron Host Link drivers, the configuration folder and common routines for the DDE Client cannot be customized.

DDE Client Tag Widgets

The following widgets are available to display information about your application's DDE Client tags:

Comm Indicator Widget

Color Line Widget

Equipment / Status Color Indicator

Gradient Color Change Widget

Indicator Light Widget

LCD 5x7 Matrix Widget

LCD 7 Segment Widget

Multi-Color Widget

Multi-Text Widget

Numeric Value Widget

Plot Data Widget

Related Information:

Use VTScada as a DDE Server – How to access VTS data from a DDE client program.

VTScada as a Local DDE Client – How VTS can read from (and write to) a DDE server program.

VTScada as a Network DDE Client – Information provided for backward compatibility.

Use VTScada as a DDE Server

VTScada is a DDE server. No configuration is required.

In your client program, you must provide address information for the VTScada program, the name of the running application, the names of the tags you wish to query, and the name of the property you want to obtain from each tag (usually '\value').

The most common use-case is to configure Microsoft Excel to display VTScada tag values in the cells of a spreadsheet. This can be done using a formula for each cell, as follows.

The formula has four parts: the name of the program (VTS), the name of the application, the name of the tag, and the property of the tag to be obtained.

As entered in an Excel spreadsheet cell, the formula would follow this form:

```
=VTS|'AppName'!'TagName\Value'
```

This formula is not literal; the various components are described in the following table.

Note: If the tag name you've specified has spaces, back-slashes, or other characters that Excel might interpret as anything other than the

name, then you must surround the 'Tag Name\Property' portion of the function with single quotes as shown. It is advisable to do this for every instance, whether required or not.

Syntax	Description
=	The equals sign (=) indicates to Excel that this is a formula.
VTS	VTS is literal, specifying the name of the program to access. If VTScada is not running when the spreadsheet opens, it will be started automatically.
|	The pipe character separates the program name from the document or topic name. (In the case of VTScada, this is the application name.)
AppName	Following the pipe character is the application name. Note: If the application name includes spaces, you must use single quotes to surround the name.
!	The exclamation point character is the topic item delimiter, separating the application name from the tag to which you are referring.
TagName	Following the exclamation point character should be the name of the tag whose data you wish to be reported in the spreadsheet cell. The tag name should appear exactly as it is displayed in the name property of the tag properties folder.
\	The back slash character (\) is the scope operator, separating the tag name from the property or data to which you are referring.
Value	The name of one of the tag's properties, whose value you wish to be reported in the cell. In most cases, this will be Value, but it could also be, Name, Area, Description, etc.

Formula Examples
The following formula examples have been provided to assist you in setting up Excel spreadsheet cells to read values from VTScada tags.
Given an application named, "DDE Test" and containing a tag named "AnalogInputTag"...
To display the tag's value in a spreadsheet cell:
=VTS|'DDE Test'!'AnalogInputTag\Value'
To display the name of the same tag's associated I/O device:
=VTS|'DDE Test'!'AnalogInputTag\IODevice'

To display the description of the same tag:

=VTS|'DDE Test'!'AnalogInputTag\Description'

> Note: The name of the tag configuration variable you are requesting must be written exactly as it occurs in the tag's code. If you are unsure, use the Source Debugger to examine an instance of the tag type.

Related Information:

...VTScada as a Local DDE Client – How VTS can read from (and write to) a DDE server program.

VTScada as a Local DDE Client

When VTScada is a local DDE[1] client, data from items within a DDE server will be reported as the value of a tag in VTScada. For example, you might want a tag to mirror the value of a cell in an Excel workbook.

> Note: You can write as well as read. In the example that follows, substitute an Analog Output or Analog Control for the input tag and you will send values out, rather than read them in.
> There must be one tag for each item to be read from (or written to).
>
> DDE is an older technology and is not supported by all newer features of VTScada.

There are three tasks to complete for VTScada to read values from a DDE server:

1. Create a DDE data source.
   For example, a Microsoft Excel™ spreadsheet.
2. Configure a new DDE Client tag within your VTScada application to communicate between VTScada and the DDE server.
3. Configure a new input tag within your VTScada application to read a value from the data source.
   Continuing the example, the value might be read from a cell in the spreadsheet.

---

[1] Dynamic Data Exchange

Example: Configuring VTScada to read values from Excel spreadsheet cells:

Part 1: Excel

1. Create or select a Microsoft Excel spreadsheet, containing values that you wish to read.
2. Within Excel's options dialogs, the R1C1 reference option must be checked.
3. Save the file to your VTScada application's folder. Example: "MyS-preadsheet.xlsx"

Part 2: VTScada DDE Client tag

1. Open the VTS application that is to read values from Excel.
2. Create a new DDE Client tag, providing a name.
3. Open the Local DDE tab of the driver tag.
4. Set the name of the DDE application to "Excel".
5. Set the name of the DDE topic to the name of the spread sheet file, "MyS-preadsheet.xlsx".
6. The Network DDE tab is not required for a local spreadsheet.
7. Click OK to save the new tag.
   You may with to draw the tag in order to ensure that there is a connection. A Comm Indicator box is suggested.

Part 3: VTScada Input tag

1. Create a new Analog Input or Analog Status tag.
2. Ensure that its I/O device is the DDE Client you created in Part 2.
3. Set the address to row and column of the cell you wish to read from using the format "RnCn".
   For example, R1C3 will read from row 1 (A) and column 3.
4. Open the Scaling tab of the analog tag.
5. Set both the Unscaled and the Scaled values to range from 0 to 1.
   This results in the analog tag reporting whatever value is in the spreadsheet cell. If your purposes require scaling, adjust the values as needed.
6. Draw the tag as you wish.

Related Information:

...Use VTScada as a DDE Server - How to access VTScada data from a DDE client program.

...DDE Client Tags – Required to allow VTScada to read from a DDE server program.

VTScada as a Network DDE Client

Note: DDEShare is based on NetDDE. For security reasons, Microsoft stopped distributing this product for versions of Windows® following XP. If you choose, you may still be able to find and install NetDDE for your version of Windows. The instructions for doing so are beyond the scope of this document.

DDE is an older technology and is not supported by every newer feature of VTScada. In particular, there is no support for linking to child tag data through DDE. Only tags at the root level may be linked successfully

When VTScada is a network DDE client, data can be entered in the cell of an Excel spreadsheet, and the value will be reported as the value of a tag in VTScada.

Five tasks must be completed in order to configuring VTScada as a network DDE client.

1. Configure a new DDE share object using the DDEShare utility.
2. Configure a new DDE Client tag within your VTScada application to communicate between VTScada and the external application with which you wish to share data.
3. Select or create an output tag within your VTScada application to write data to the external application.
4. Select or create an input tag within your VTScada application to read the data that you are writing to the external application to ensure that communications are occurring as they should.
5. Configure an Excel spreadsheet of the same name specified in your DDE share object, and ensure that the path to the file is correctly configured within Excel for VTScada to locate it.

The steps for each task follow. DDE Share screen-shots were created using Windows XP, the last version of Windows that included native support for this technology. Your screens may differ.

Part 1: Configure a New DDE Share Object Using DDEShare

In order to designate VTScada as a DDE client, you must first configure the network DDE on the server PC using the DDEShare utility. The following instructions will guide you through the correct process of configuring DDEShare.

1. Obtain and install a copy of NetDDE.

2. Open the Windows Start menu.

3. Enter "DDEShare" in the Search for Programs and Files field.

4. Click OK. The DDE Share dialog opens.
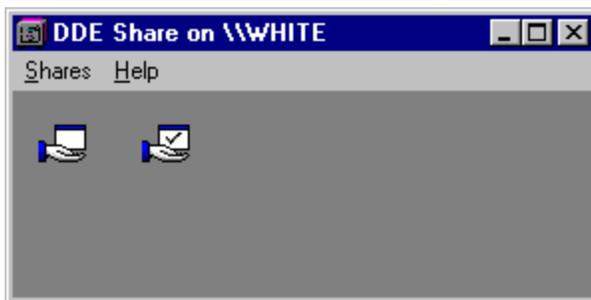
5. Double-click the icon that does not display a check mark. The DDE Shares dialog opens similar to the following:
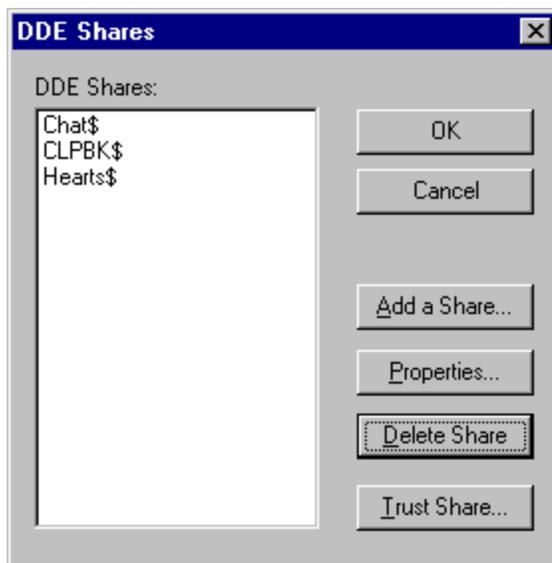
6. Click the Add a Share button. The DDE Share Properties dialog opens as shown.
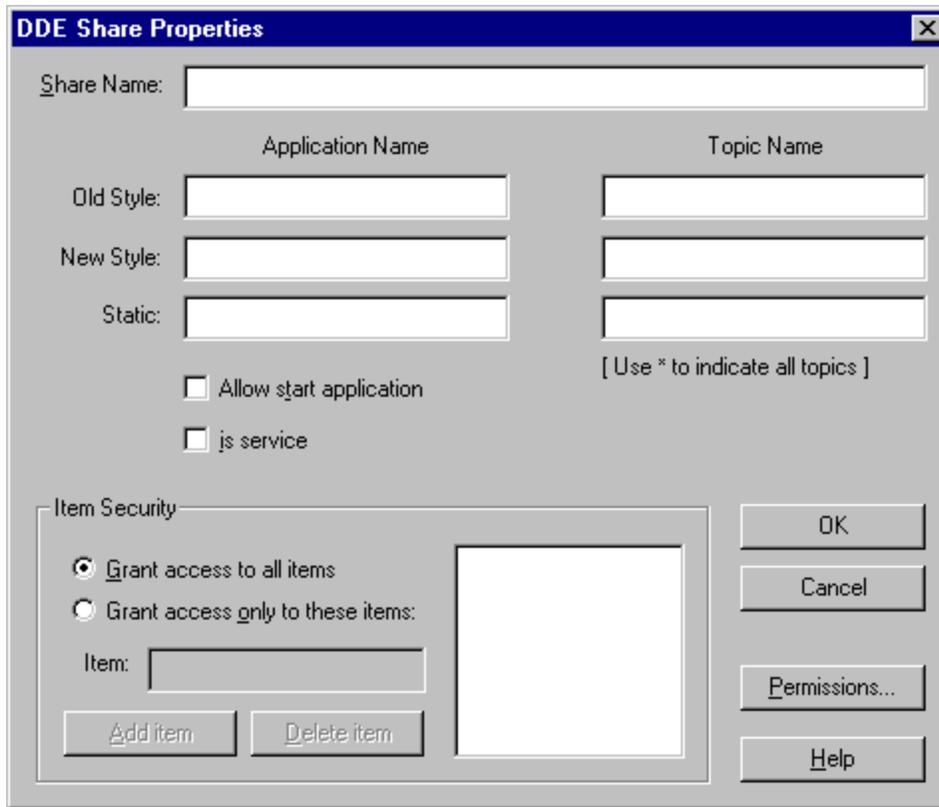
7. Enter a unique name for the share item in the Share Name field followed by the dollar sign character ($) (e.g. "DDEShare$"). (The Share Name will not accept duplicates any of the names of existing shares in the DDE Shares dialog.)

8. Enter the application name and topic name in the Old Style and New Style fields as shown in the next image.

9. Select the Allow start application check box. The DDE Share Properties should appear similar to the following:
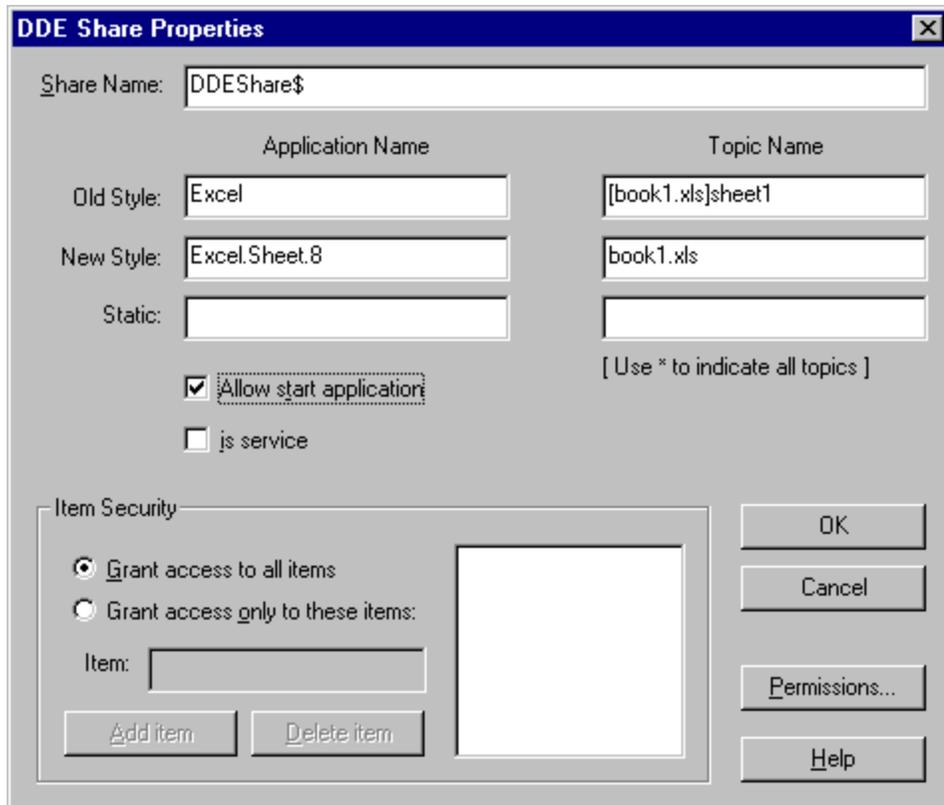
Observe the example settings shown in the DDE Share Properties dialog above: "book1.xls" identifies the name of the Excel workbook into which you wish the data from VTScada to be entered. This can be substituted for any Excel workbook name you choose; "sheet1" is the name of the Excel sheet that the DDE Share is going to access within the selected Excel work-book. You may enter any sheet name you choose, as long as the sheet exists within the specified Excel workbook; finally, select the "Allow start applic-ation" check box to enable VTScada to start Excel and open to the correct sheet in the specified workbook.

10. Click the Permissions button. The DDE Share Name Permissions dialog opens as shown, and displays the names of the current users and groups.
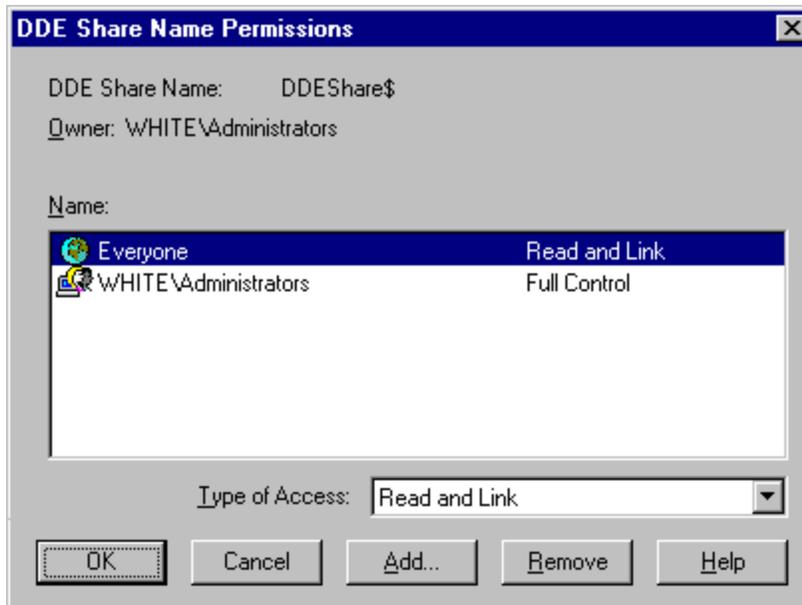
11. Select any local or domain users that you wish to be granted access to the new share, and select an appropriate option from the Type of Access drop-down list. If security is not an issue, you may wish to select "Everyone" and set the Type of Access to "Full Control". This will ensure trouble-free con-nectivity (as far as new users on the network are concerned). A second option might be to create a new group and grant the group permission to access the new share. You can later enable new users to access the share simply by adding them to the group.
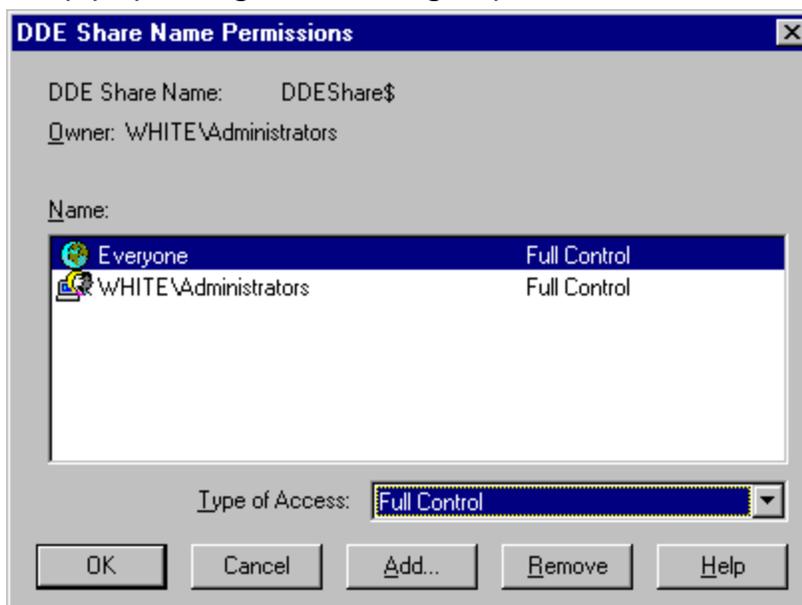
12. Click OK. You are returned to the DDE Share Properties dialog.
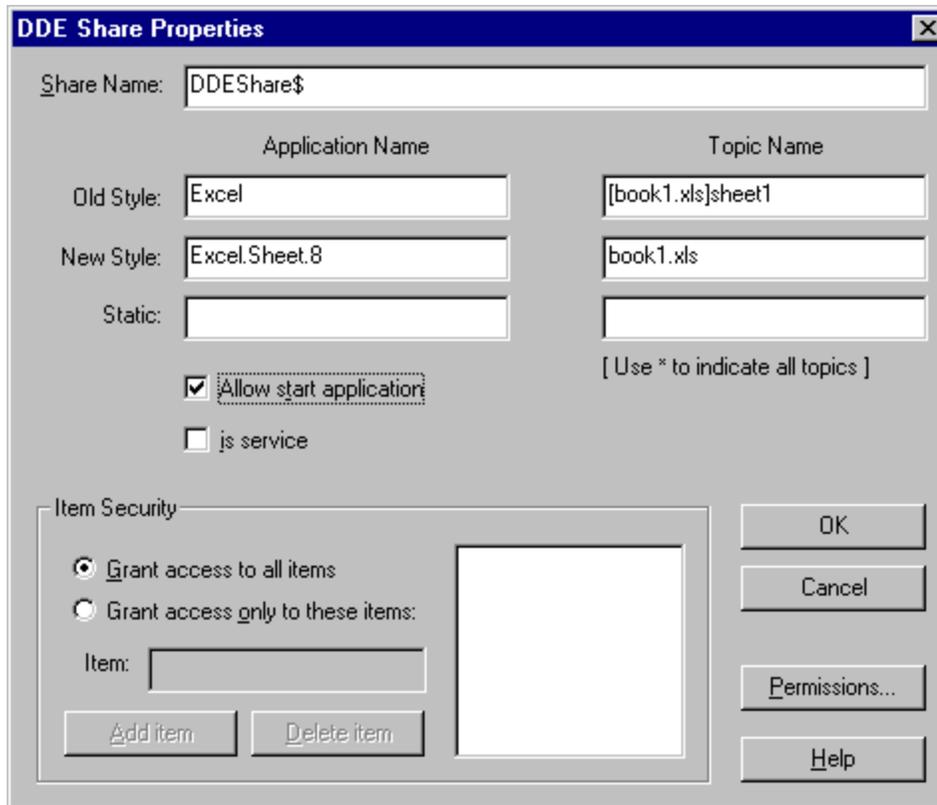
13. Click OK. You are returned to the DDE Shares dialog where the name of the new share is added to the list.
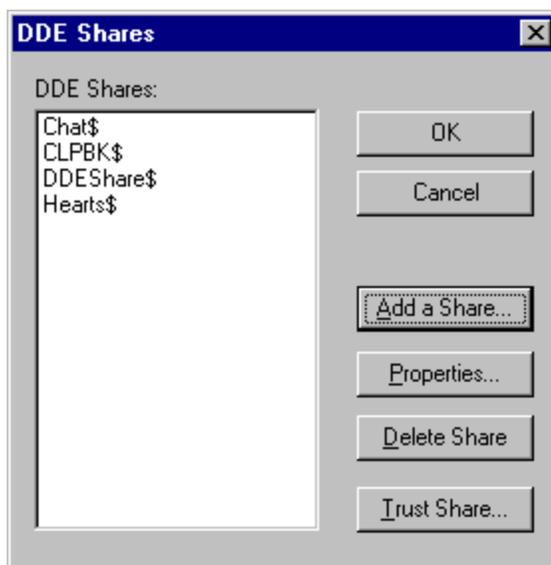
14. Select the new share in the DDE Shares list. The new share is highlighted.

15. Click the Trust Share button. The Trusted Share Properties dialog opens as shown.
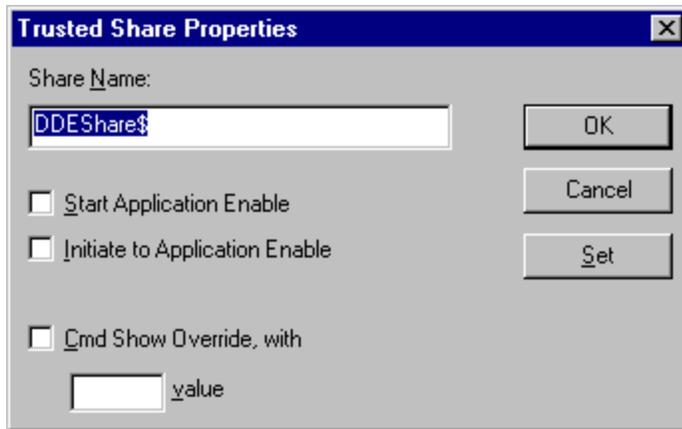
16. Select the Start Application Enable check box.

17. Select the Initiate to Application Enable check box. The Trusted Share Properties dialog should appear as shown:
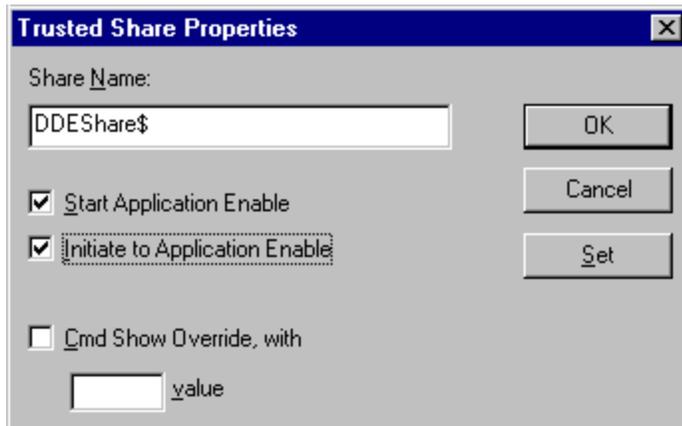
18. Click Set to set the properties for the new share.

19. Click OK. You are returned to the DDE Shares dialog.
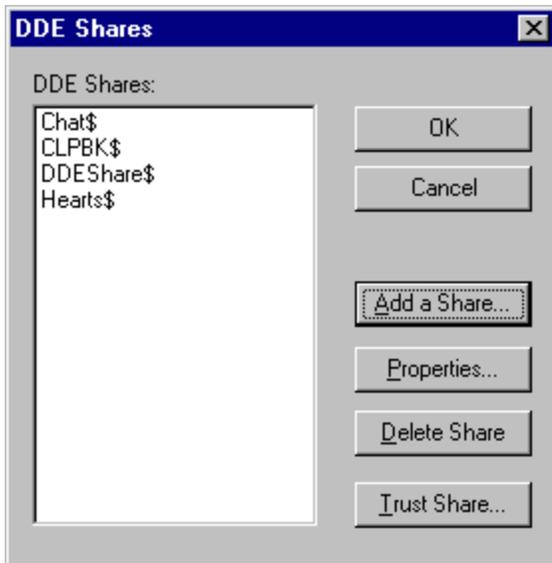
20. Click OK. You are returned to the DDE Share on WorkstationName dialog.

21. Close the DDE Share on WorkstationName dialog using the Windows close button. You can now begin to configure the DDE Client tag and input/output tags within the VTScada application on the client PC.

Part 2: Configure a New DDE Tag to Communicate between VTScada and an External Application

The next step in communicating between VTScada (the client) and an external application (such as Microsoft Excel), is to create a new DDE tag.

1. Run VTScada. The VAM opens.

2. Select the VTScada application from which you wish to output data to the external application.

3. Click the Run button. The application runs.

4. Log on to the application (if necessary) using your username and password.

5. Navigate to the page upon which you wish to place your DDE-related tag widgets.

6. Open the Tag Browser.

7. Select DDE Client from the Types drop-down list.

8. Click New. The New DDE Client Properties folder opens.

9. Enter a name for the new DDE Client tag in the Name field.

10. Enter an area for the new DDE Client tag in the Area field.

11. Enter a description for the new DDE Client tag in the Description field. An example follows:
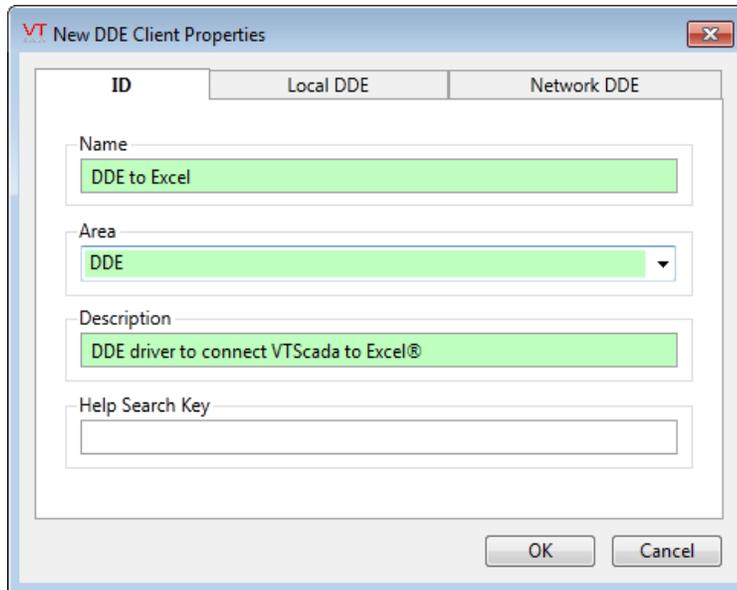
12. Click the Network DDE tab.

13. Enter the name of the server on which the DDE share created in Part 1 was created in the DDE Computer Name field.

14. Enter the name of the new DDE share created in Part 1 (in this example, "DDEShare$"), followed by the extension .OLE.

15. Click OK. The DDE Client tag properties folder closes and the information is saved in the tag properties database. You are returned to the Tag Browser where the name of the new DDE tag is displayed and selected.

16. Click the Draw button and select the Comm Indicator widget. The Comm Indicator square is attached to the mouse pointer.

17. Click to drop the Comm Indicator object on the page. The Comm Indicator dialog opens as shown.
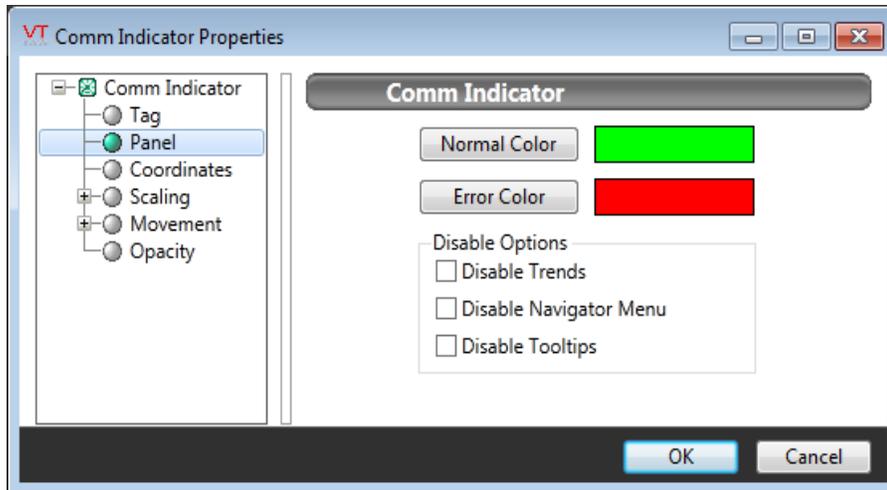
18. Click OK.

The Comm Indicator object shows the status of communications between the DDE Client and the external application, by way of a changing color square. By default, green indicates communications are occurring, red indicates that communications have failed, and gray indicates that the tag is reading invalid (as it should be).

Part 3: Configure a New Input Tag to Read Data from the External Application

The next step in setting up VTScada as a DDE client is to configure a new input tag (either analog or digital) to read data from the external application on the remote PC via the DDE tag created in Part 2. This tag will help you to confirm that the data you will be outputting (via the output tag to be created in Part 4) is being transmitted.

1. Open the Tag Browser.
2. Click New and set the type to Analog Input.
3. Enter a name for the new Analog Input tag in the Name field.
4. Enter an area for the Analog Input tag in the Area field.
5. Enter a description for the Analog Input tag in the Description field. The completed properties for the ID tab of the Analog Input tag properties folder should appear similar to the example shown:
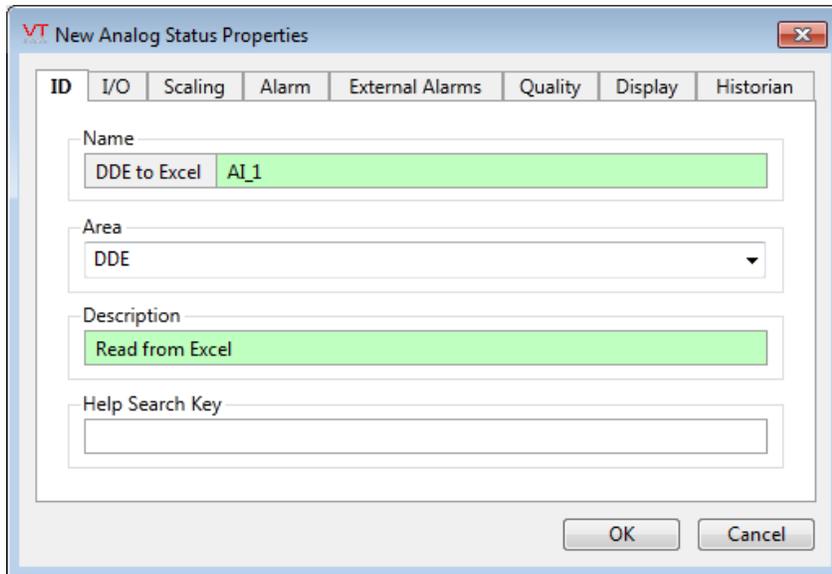
6. Click the I/O tab.
7. Select the name of the DDE Client tag from the I/O device drop-down list.
8. Enter the address of the location from which you wish to read data in the Address field. (In this example, the goal is to read data from row 1 cell 1 of the Excel spreadsheet, therefore enter "r1c1" in the Address field. The completed I/O tab should appear similar to the following image.
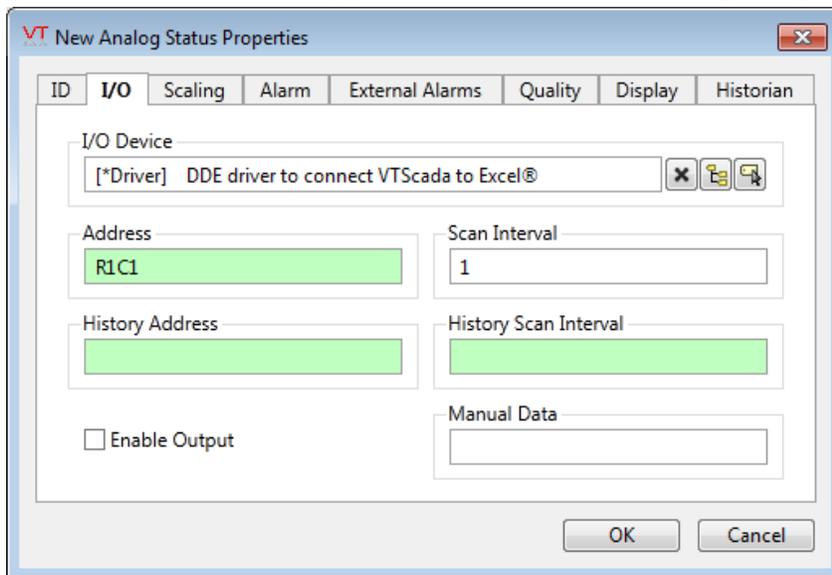
9. Click OK. You are returned to the Tag Browser where the new Analog Input tag is displayed and selected.
10. Close the Tag Browser.
11. Open the Idea Studio and select the Widgets palette.

12. Select one of the widgets that display the numeric value of the tag (i.e. Numeric Value or Meter1, Meter2, or Meter3), and drag it onto the page.

13. Link the widget to your Analog Input tag.

14. Configure the widget's display properties as you require.

Part 4: Configure a New Output Tag to Write Data to the External Application

The next step in setting up VTScada as a DDE client is to configure a new output tag (either analog or digital) to write data to the external application on the remote PC via the DDE tag created in Part 2.

1. Open the Tag Browser.

2. Click New and set the type to Analog Output.

3. Enter a name for the new analog output tag in the Name field.

4. Enter an area for the new analog output tag in the Area field.

5. Enter a description for the new analog output tag in the Description field.

6. Click the I/O tab. The properties of the I/O tab are displayed.

7. Select the name of the DDE Client tag created in Part 2.

8. Enter the address of the location to which you wish to write data in the Address field. (In this example, the goal is to write data to row 1 cell 1 of the Excel spreadsheet, therefore enter "r1c1" in the Address field.

9. Click OK.

10. Close the Tag Browser.

11. Open the Idea Studio and select the Widgets palette.

12. Expand the Analog Outputs group.

13. Select the Numeric Entry widget and drag it onto the page.

14. Link the widget to the tag you created.

15. Configure the display properties as desired.

Part 5: Configure an Excel Spreadsheet with the Name as Specified for Your DDE Share Object, and Ensure the Path is Correctly Configured Within Excel

The configuration of VTScada as a DDE client is now almost complete. The only tasks remaining are to create a spreadsheet with the same

name as was specified for your DDE share object, and ensure that the path is properly configured within Excel so that VTScada can locate it.

1. Run Microsoft Excel.

2. Create a new spreadsheet. Name this spreadsheet with the exact same name as you configured within your DDE share. For example, consider the DDE Share Properties dialog, shown in the following image. In the example DDE share object, the name of the Excel spreadsheet is given as "book1.xls", therefore, the spreadsheet must be named to match.

3. Save the new spreadsheet in your application directory.
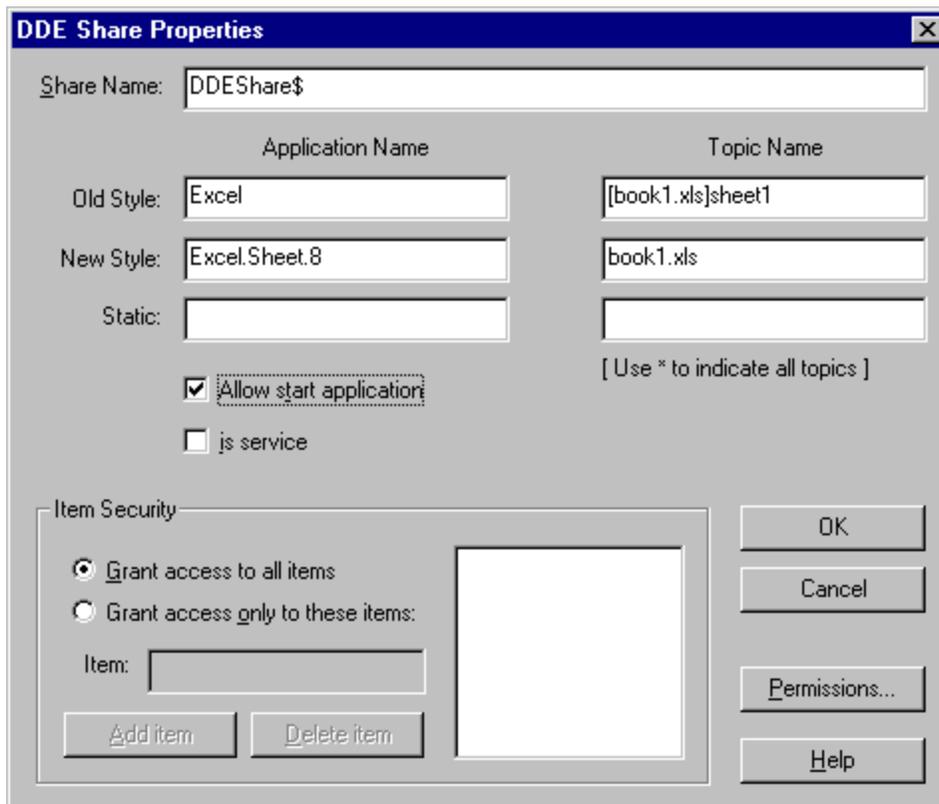
4. Select Tools|Options in Excel's menu. The Options dialog opens.

5. Click the General tab.

6. Enter the path to your application directory in the Default File Location field.

7. Click OK.

The configuration of VTScada as a DDE client is now complete. To output data values to Excel, simply enter a value in the Numeric Entry field (i.e. the widget you used for your output tag). When you press the Enter key, the value you entered in the Numeric Entry field for the analog output

tag is transmitted to the Excel spreadsheet. If you look in the Excel spreadsheet, the cell you specified to write the data to should display the output number. The widget you used for the Analog Input tag should also display the number you entered, as it reads it back from the spreadsheet.

> Note: If the value you enter in the Excel spreadsheet is not registering, try stopping both Excel and your VTScada application, compile your VTScada application, then restart it and reenter the value you wish to output.

DNP3 Driver Tags

Not counted towards your tag license limit.
This tag is a DNP3 Master, which implements all the objects defined in DNP subset level 3, except for the "Frozen Counter" type. The driver does not generate "Freeze" requests, or class enable/disable messages. The driver supports a number of other objects as listed in a following topic.

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

DNP3 Drivers may be configured to accept in-bound connections from an RTU via a configured IP Network Listener tag. In DNP3 terms this is a 'dual-endpoint' where the IP network listener allows this to be a 'listening endpoint'.
This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the Options tab to decide whether this feature should be used in your application.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Related Information:

DNP3 I/O Addressing

 DNP object types supported

IP Network Listener Tags

DNP3 Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might

also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

DNP3 Driver Properties: Serial Tab

The Serial Tab of the DNP3 driver tag properties folder is used to set the parameters used to communicate with the driver. If the configured port is using a modem for communications to a DNP3 outstation, this tag may be configured to accept incoming phone calls.
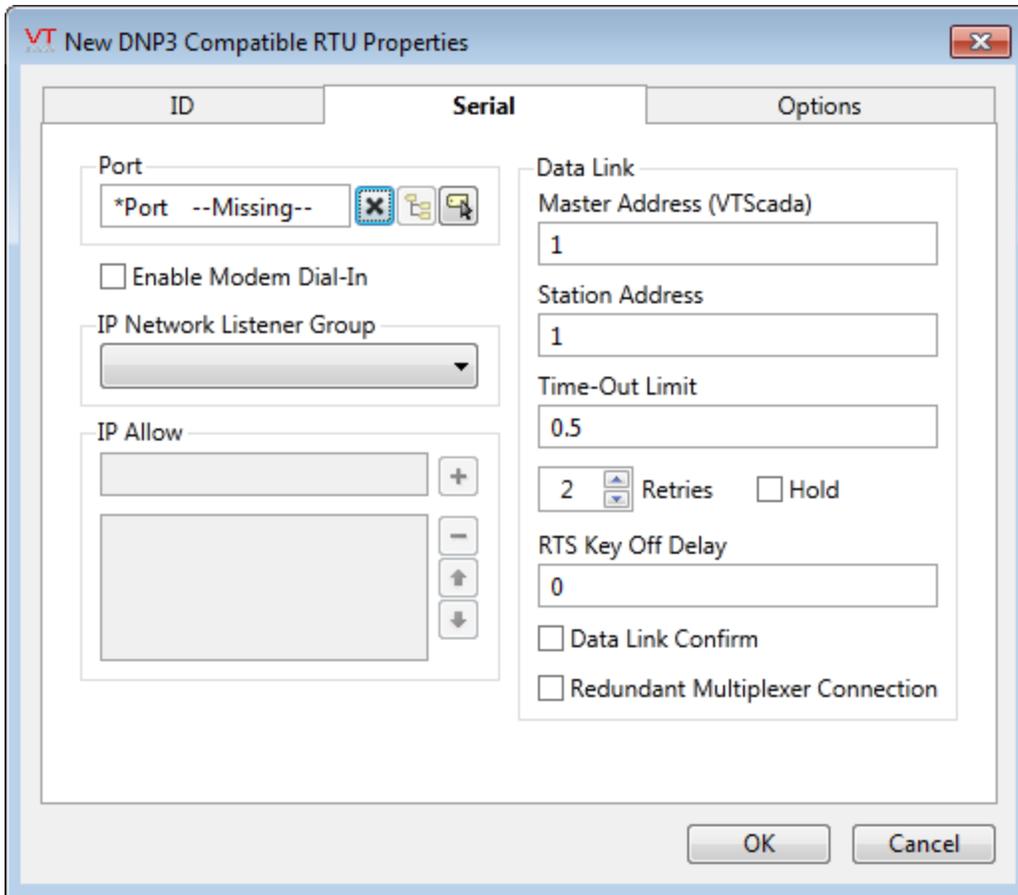
Port

Enables you to select the Serial Port tag, TCP/IP tag or UDP/IP
Port tag that you wish to be associated with this driver tag. A
Serial Port tag opens a serial port to enable communications
between VTScada and your PLC or RTU. A TCP/IP or UDP/IP tag
enables you to connect to a series of hosts, allowing you to
transmit data across a network or over the Internet. A port
must be configured regardless of whether an IP Network
Listener is also configured.

The Port field can be cleared using the X button that appears
to its right.

Enable Modem Dial-In

Should be configured only if you expect the RTU to dial in using the VTScada Modem Manager. Do not use this option if an IP Network Listener is configured.

IP Network Listener Group

Configure only if this driver is to accept in-bound TCP or UDP communications. Select an existing IP Network Listener tag using the drop-down list. While a TCP connection is open, new connections will be rejected by the DNP3 driver.
If an IP Network Listener is configured, the Port tag must be a TCP/IP configured in client socket mode, or a UDP/IP tag.

IP Allow

Optional. Misconfiguration of this field is a common source of problems.
Used in connection with an IP Network Listener tag. General IP address filtering should be set on the IP Network Listener, as this will be more efficient. The list of allowed IP addresses in the driver is intended to prevent misconfigured devices from interfering with other devices.
Both filters (that in the Network Listener and this one) can be configured at the same time. The filter in the IP Network Listener will be applied when the device first connects and the local filter will be applied after a specific driver instance has been identified.

Master Address (VTS)

The station address for VTScada. This is the "from" address in messages from VTScada to the RTU, and the "to" address in messages from the RTU to VTScada. The range is 0 to 65535.

Station Address

The station address for the RTU. The valid range is 0 to 65535.

Time Out Limit

Sets the time in seconds or fractions of a second that this driver should wait for a reply from the remote unit.

RTS Key Off Delay

Sets the amount of time (in seconds) that this driver will wait before dropping RTS at the end of a data transmission.

Retries

The Retries spin box enables you to select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Hold

Select whether this driver should hold data from the PLC or RTU in the event of a communications failure.
If the Hold check box is selected, the last received value from the PLC or RTU will be held.
If the Hold check box is not selected, the data will be invalidated in the event of a communications error.
By default, the Hold check box is not selected.

Data Link Confirm

If set, every low level data link message will be confirmed by an acknowledgment. This option is generally used in an environment where there are unreliable communications due to the extra non-data traffic sent.

Redundant Multiplexer Connection

Used only in the case that two DNP3 drivers will be subordinate to a Driver Multiplexer. If checked, this setting will prevent activity if the driver is a DriverMUX inactive subordinate.
Note: This setting is incompatible with the DriverMUX's Parallel or Alternating modes.
When in this mode (option selected, current driver is the inactive subordinate driver) the DNP3 Driver will almost completely shut down except for periodic checking of its communications link.

DNP3 Driver Properties: Options Tab

Use the properties within this tab to fine-tune the driver behavior, adjusting it for your particular hardware.
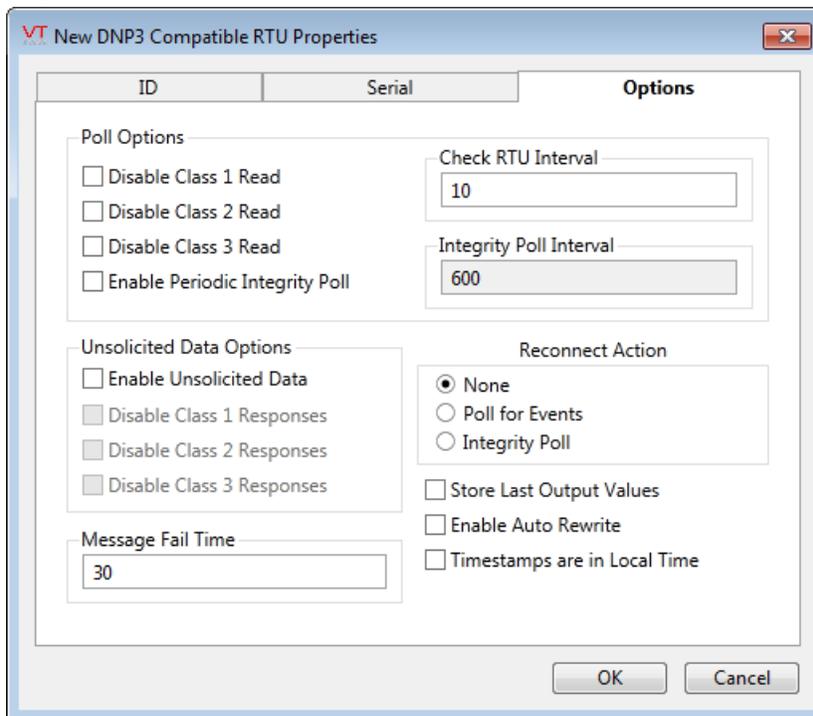
Poll Options – Disable Class x Reads

Disable Class 1, 2 and 3 Reads. Any of these three DNP3 data classes may be disabled independently. A class marked as

disabled will not be read, even if the outstation indicates that class data is available.

Check RTU Interval

The interval (in seconds) at which VTScada should check for class 1, 2 and 3 event data in the RTU. This function performs two tasks: checks that the outstation is still functioning and checks to see if the outstation has an event pending, which may be read. The outstation's configuration will determine which data changes are reported by events.
If your RTU is configured to spontaneously send values as they change (unsolicited data) you may choose to set the Check RTU Interval value to a relatively large value, thus minimizing band-width while still ensuring that values have been sent.

Enable Periodic Integrity Poll & Integrity Poll Interval

This option may not be required.
When checked, VTScada will send a request for all values every Integrity Poll Interval seconds, even though these values may not have changed.
An Integrity Poll will read all enabled Class 1, 2, and 3 events (changes) then all Class 0 (current) values. Note that Class 0 values may not include all data types – this depends entirely on the configuration of the DNP3 device.
Regardless of whether the Enable Periodic Integrity Poll option is set, there are three situations when an integrity poll will always occur:

- When the driver becomes the I/O server.
- When an RTU signals that is has re-started.
- When an RTU signals that event buffers have overflowed.

Unsolicited Data Options – Enable Unsolicited Data

The Enable Unsolicited Data option must be checked for VTScada to listen for unsolicited event messages (exceptions) from the RTU.
The configuration of the Allow Unsolicited Communications check box in the Serial tab has no effect on this behavior.

Disable Class x Responses

These options are available only when the Enable Unsolicited Data option has been set. Disable Class x Response to prevent the RTU from spontaneously sending event messages for that class. This affects only unsolicited messages from the RTU.

Reconnect Action

Choose what action to take when a connection to an RTU is re-established. The default action is "None" since the connection is usually re-established by VTScada making a Read request. No additional action is required in this case.
In rare circumstances an RTU may make the connection, then send no data. The Poll For Events option exists so that you can direct the DNP3 driver to send a request for changed data when this occurs.
In older (pre-10.0.12) versions of the driver, a reconnect would always cause an integrity poll to be done. The Integrity Poll option exists for legacy application support.

Message Fail Time

The time interval for which VTScada will delay a failed message to allow other messages on the same port to be transmitted. This enables better performance when multiple RTUs share the same port and a message or messages to one RTU are timing out, thus slowing down communications to all RTUs. Usually

set to 0 when there is only one driver using the port.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one. If the last output values are stored, they may be re-written by either of two methods:
- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.
  Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated.
This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.
Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Timestamps are in Local Time

> Controls how the driver will send and interpret received timestamps from the RTU. If checked, then timestamps are in local time. If FALSE, then timestamps are in UTC. For most applications, UTC is the proper way to work with timestamps. The local option exists for compatibility with legacy applications.
>
> The default for the parameter is controlled by the application property, DNP3LocalTimestampsDefault. Applications, upgraded from versions before VTS 10 will default to '1', but new applications will default to '0' (UTC).

DNP3 Driver Tag Widgets

The following widgets are available to display information about your application's DNP3 driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

DNP3 I/O Addressing

See also: DNP object types supported.
Within VTScada, DNP I/O addresses for all points except File Identifiers (object 70) are formatted in the following manner:
Obj/Var/Index[:NS | :LATCH | :PULSE]
Where:

- Obj is the DNP object to use. A table of supported object types can be found later in this topic.
- Var is the DNP variation of the object
- Index is the points index. For the Restart flag in the Internal Indications (object 80, var 1) use an index of 7.
- :NS is used to indicate that the point should not be scanned, as its value will be updated either by a DNP event, or the background class 0 scan. If this option isn't specified, then the point will be scanned at the rate specified in the points scan rate field. You must use :NS when reading history data from the device.
- :LATCH is used to indicate the output type is a latched control.
- :PULSE is used to indicate the output type is a pulsed control.

For outputs, if neither :Latch or :Pulse is specified then the output type is a Trip/Close control. Refer to the following notes on Binary Output.
VTScada provides the DNP 3.0 Address Select dialog to help you build these addresses. In any tag that uses a DNP3 driver, the I/O panel of the configuration dialog will provide a button beside the Address field:

Click this to open the Address Select dialog.
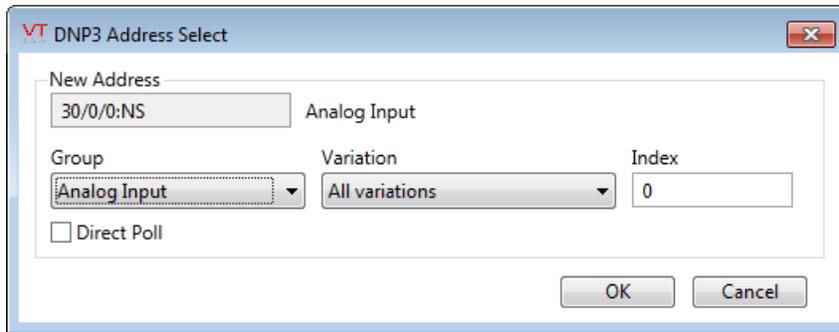
To use this dialog, start by selecting the Group designation of the I/O device.

Available groups include:

- Binary Input
- Binary Output Status
- Counter
- Frozen Counter
- Analog Input
- Analog Output Status
- Time and Date
- Internal Indications
- Octet String

If the Direct Poll option is not checked, then (for the appropriate groups) the suffix :NS will be added to the address and no variations may be applied.

Click the Direct Poll option to allow scanning and to enable the Variation selector, the content of which will vary according to the selected Group.

Binary Output:

Devices may use one of three control models:

- Activation Model. A single signal. The output is activated for the duration of the on-time regardless of the code being sent.
- Complementary Latch Model. A single output signal that is held at the commanded state.

- Complementary Two-Output. Two outputs named close and trip. One or other output is set active momentarily for a commanded on-time and off-time.

The VTScada output address suffixes, :Latch, :Pulse, or no suffix indicating a Trip/Close control, relate to these models as follows:

Note: For the following, the term "on-time" refers to the duration of the signal, not the value or control code sent in the signal.
By the DNP3 specification, timings are suggestions. The device may use the supplied timing, may limit the supplied timing to its own configured limits, or may substitute timing with its own configured values.
The suffix :LATCH should not be confused with the Complementary Latch Model of control.
To send an exact value of zero (0) or one (1), a Digital Control or Digital Output tag is recommended. To send any other value, use an Analog Control or Analog Output tag.

Trip/Close control (No suffix added to the address)

Values greater than zero send a close request (control code 0x41) and values equal to or less than zero send a trip request (control code 0x81). The absolute value being written is the duration of the on-time, measured in milliseconds. Note that a value of zero, written by the tag, sends a trip request with an on-time of one millisecond.

- Activation Model: The output is set to active for the duration of the on-time.
- Complementary Latch Model: The output is set to active for a close (value $>$ 0) or set to inactive for a trip (value $<=$ 0).
- Complementary Two-Output Model: Either the close output is set to active if a close (value $>$ 0) is sent, or the trip output is set active for a trip (value $<=$ 0), for the duration of the on-time.

Examples:

1 – sends a close request with an on-time of 1 millisecond.
0 – sends a trip request with an on-time of 1 millisecond.
1000 – sends a close request with an on-time of 1 second.
-1000 – sends a trip request with an on-time of 1 second.

:LATCH suffix

The DNP3 standard does support sending an on-time and off-time for the latch on|off codes, but this driver does not when the :LATCH suffix is in use. It always sets the times to 0, so is likely to be useful only with a complementary latch module output.

A Digital Control or Digital Output tag is recommended.

A value greater than zero (1) sends a latch-on request (control code 0x03).

A value less than or equal to zero (0) sends a latch-off request (control code 0x04).

- Activation Model: The output is set to active.
- Complementary Latch Model: The output is set active for a latch-on and inactive for a latch-off.
- Complementary Two-Output Model: The output is set active for a latch-on and inactive for a latch-off.

Examples:

1 – sends a latch-on request.
0 – sends a latch-off request.

:PULSE suffix

Neither of the complementary models are supported.

Both an on-time and an off-time can be specified as follows. Given a numeric value, "nnn.fff"

- The absolute value of the digits preceding the decimal point (nnn) specifies the on-time in milliseconds.
- The absolute value of the digits following the decimal point (fff) specifies the off-time in milliseconds.

Positive values greater than zero (> 0) send a pulse-on request (control code 0.01).

Negative values (<= 0) send a pulse-off request (control code 0x02).

- Activation Model: The output is set active for the duration of the on-time.
- Complementary Latch Model: A "NOT-SUPPORTED" status will be returned.

- Complementary Two-Output Model: A "NOT-SUPPORTED" status will be returned.

Examples:

1 – sends a pulse-on request with an on-time of 1 millisecond and an off-time of 0.

0 – sends a pulse-off request with an on-time of 0 milliseconds and an off-time of 0. This may be an error as there should be some form of on-time.

-1 – sends a pulse-off request with an on-time of 1 milliseconds and an off-time of 0.

5000.999 – sends a pulse-on request with an on-time of 5 seconds and an off-time of 0.999 seconds.

-4000.003 – sends a pulse-off request with an on-time of 4 seconds and an off-time of 3 milliseconds.

Select Before Operate (:SBO)

To set a point to use SBO rather than direct operate, append the suffix ":SBO" to the I/O address. The SBO suffix can be used in combination with the other output attribute suffixes; ":PULSE" or ":LATCH".
This suffix is available as an option in the DNP3 address assist dialog when building a binary output command.

Reads

Address examples for reading values:

To read a digital input with status (obj 1, var 2) at address 4321 use

```
1/1/4321.
```

To read a 32 bit delta counter without status (obj 20, var 7) at address 876 use

```
20/7/876
```

Outstation Directory Reads

Outstation directories may be read using an I/O address as follows:

```
70/5/DirectoryPath,READDIR
```

Where:

- DirectoryPath is the path to the directory to be read. This will vary by outstation.

The normal use for this will be from script via a Read command.

Example:

```
ReadObj = Variable("DriverName")\DriverRead
("70/5//data,READDIR", 1, &Result)
```

ReadObj will go invalid on completion.

Result will be set to an 2D array of directory entries, one row for each entry where the columns are as follows:

[0]	Entry name	Either file name or a sub-directory name
[1]	Entry Type	1 for a file, 0 for a sub-directory
[2]	Entry size	File length for a file, number of entries for a sub-directory
[3]	Timestamp	UTC
[4]	Permissions*	The permissions for the entry

* DNP3 File permissions are a set of 9 bits, indicating read, write and execute permissions for the user, group and world. Normally written in octal, the value 0777 is "all permissions for all classifications". The exact meaning of the bits is outstation-dependent.

DNP3 File Writes

Files may be written to the outstation using an I/O address as follows:

```
70/5/PathToFile[,Timestamp[,Permissions[,Append]]]
```

Where:

- PathToFile includes the path for the file. This will include forward slashes for directories. The exact path will depend on the outstation – see example.

- The optional parameter, Timestamp, is the timestamp for the file on the outstation in seconds since 1/1/1970, UTC. The default is the current time of the I/O server (in UTC).

- The optional Permissions* are the permissions for the file. The default is 0777.

- Append is a flag indicating the behavior of the write if the indicated file already exists on the outstation. If TRUE, the file will be appended to. If FALSE, the file will be overwritten by the new data. The default is false (no append)

The normal use for this will be from script via a Write command.

Example:

```
WriteObj = Variable("DriverName")\Driver\Write
("70/5//data/newdata.txt", 1, &FileStrm, Invalid, "SomeText", &Result)
```

Result will become valid once the file write has completed – TRUE for success, FALSE for failure. In addition, WriteObj will become invalid on completion.

FileStrm, the data to be written to the outstation file, can be a stream or a buffer. In the example given, the file to be written to is "/data/newdata.txt". The paths used will depend on the outstation.

DNP3 File Deletions

Files may be deleted from the outstation using an I/O address as follows:

```
70/5/PathToFile,DELETE
```

Where: PathToFile includes the path for the file to be deleted.

The normal use for this will be from script via a Write command.

Example:

```
WriteObj = Variable( "DriverName")\Driver\Write(
"70/5//data/newdata.txt,DELETE", 1, &FileStrm, Invalid, "SomeText",
&Result)
```

Note that some valid data must be provided as the data parameter (FileStrm) but it's contents are ignored.

DNP object types supported

Please refer also to the Implementation Table of the document, DNP3 Device Profile.pdf, included with your VTScada program.

Object Description	Object Type	Object Variation	Read	Write
Single bit binary input all variations	1	0	√	
Single bit binary input	1	1	√	
Single bit binary input with status	1	2	√	
Single bit binary input event all variations	2	0	√	
Single bit binary input event without time	2	1	√	
Single bit binary input event with time	2	2	√	
Single bit binary input event with relative time	2	3	√	
Single bit binary output all variations	10	0	√	
Single bit binary output status	10	2	√	
Control relay output block	12	1		√
Binary counter all variations	20	0	√	
32 bit binary counter	20	1	√	
16 bit binary counter	20	2	√	
32 bit delta counter	20	3	√	
16 bit delta counter	20	4	√	
32 bit binary counter without status	20	5	√	
16 bit binary counter without status	20	6	√	
32 bit delta counter without status	20	7	√	
16 bit delta counter without status	20	8	√	
Frozen binary counter all variations	21	0	√	
Frozen 32 bit binary counter with status	21	1	√	
Frozen 16 bit binary counter with status	21	2	√	
Frozen 32 bit delta counter with status	21	3	√	

Frozen 16 bit delta counter with status	21	4	√	
Frozen 32 bit binary counter with time	21	5	√	
Frozen 16 bit binary counter with time	21	6	√	
Frozen 32 bit delta counter with time	21	7	√	
Frozen 16 bit delta counter with time	21	8	√	
Frozen 32 bit binary counter no status	21	9	√	
Frozen 16 bit binary counter no status	21	10	√	
Frozen 32 bit delta counter no status	21	11	√	
Frozen 16 bit delta counter no status	21	12	√	
Binary counter event all variations	22	0	√	
32 bit binary counter event without time	22	1	√	
16 bit binary counter event without time	22	2	√	
32 bit delta counter event without time	22	3	√	
16 bit delta counter event without time	22	4	√	
32 bit binary counter event with time	22	5	√	
16 bit binary counter event with time	22	6	√	
32 bit delta counter event with time	22	7	√	
16 bit delta counter event with time	22	8	√	
Frozen binary counter event all variations	23	0	√	
Frozen 32 bit binary counter event without time	23	1	√	
Frozen 16 bit binary counter event without time	23	2	√	
Frozen 32 bit delta counter event without time	23	3	√	
Frozen 16 bit delta counter event without time	23	4	√	
Frozen 32 bit binary counter event with time	23	5	√	
Frozen 16 bit binary counter event with	23	6	√	

time				
Frozen 32 bit delta counter event with time	23	7	√	
Frozen 16 bit delta counter event with time	23	8	√	
Analogue input all variations	30	0	√	
32 bit Analogue input	30	1	√	
16 bit Analogue input	30	2	√	
32 bit Analogue input without status	30	3	√	
16 bit Analogue input without status	30	4	√	
Float analogue input without status	30	5	√	
Double Analogue input without status	30	6	√	
Analogue input event all variations	32	0	√	
32 bit Analogue input event without time	32	1	√	
16 bit Analogue input event without time	32	2	√	
32 bit Analogue input event with time	32	3	√	
16 bit Analogue input event with time	32	4	√	
Float Analogue input event without time	32	5	√	
Double Analogue input event without time	32	6	√	
Float Analogue input event with time	32	7	√	
Double Analogue input event with time	32	8	√	
Analogue output all variations	40	0	√	
32 bit Analogue output status	40	1	√	
16 bit Analogue output status	40	2	√	
Float Analogue output status	40	3	√	
Double Analogue output status	40	4	√	
32 bit Analogue output control block	41	1		√
16 bit Analogue output control block	41	2		√

Float Analogue output control block	41	3		√
Double Analogue output control block	41	4		√
Time and Date	50	1	√	√
Common Time Object	51	1	√	
Common Time Object – unsynchronized	51	2	√	
Time delay coarse	52	1	√	
Time delay fine	52	2	√	
Class 0 Event data	60	1	√	
Class 1 Event data	60	2	√	
Class 2 Event data	60	3	√	
Class 3 Event data	60	4	√	
File Identifier (older systems)	70	1	√	√
File Identifier (preferred object for newer systems)	70	5	√	
Internal Indications object	80	1		√
Octet String	110	*	√	√
Octet Change Object*	111	*	√	

*Event\change objects would not normally be configured as I/O addresses. They are listed to indicate that processing support is provided them. Upon receiving an event\change object, the value of the corresponding static object will be updated. For example, if 110/255/6 is configured and an event is received for 111/13/6, then the tag reading the first address will be updated with the value from the event\change as they share the same index.

Static\event object pairs that follow this rule include:

- g1 Binary Input and g2 Binary Input change
- g10 Binary Output and g11 Binary Output change
- g20 Counter and g22 Counter change
- g21 Frozen Counter and g23 Frozen Counter change
- g30 Analog Input and g32 Analog Input change

- g40 Analog Output and g42 Analog Output change
- g110 Octet Strings and g111 Octet String change

Data values returned for the objects that may be read:

Object Description	Return Value
Single bit binary input	0 or 1.
Single bit binary input with status	If status is OK, 0 or 1, else invalid.
Single bit binary input event without time	If status is OK, 0 or 1, else invalid. Refer to the note following this table.
Single bit binary input event with time	If status is OK, 0 or 1, else invalid. Refer to the note following this table.
Single bit binary output status	If status is OK, 0 or 1, else invalid.
32 bit binary counter	If status is OK, 32 bit unsigned integer, else invalid.
16 bit binary counter	If status is OK, 16 bit unsigned integer, else invalid.
32 bit delta counter	If status is OK, 32 bit unsigned integer, else invalid.
16 bit delta counter	If status is OK, 16 bit unsigned integer, else invalid.
32 bit binary counter without status	32 bit unsigned integer.
16 bit binary counter without status	16 bit unsigned integer.
32 bit delta counter without status	32 bit unsigned integer.
16 bit delta counter without status	16 bit unsigned integer.
32 bit binary counter event without time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit binary counter event without time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit delta counter event without time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this

	table.
16 bit delta counter event without time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit binary counter event with time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit binary counter event with time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit delta counter event with time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit delta counter event with time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit Analogue input	If status is OK, 32 bit signed integer, else invalid.
16 bit Analogue input	If status is OK, 16 bit signed integer, else invalid.
32 bit Analogue input without status	32 bit signed integer.
16 bit Analogue input without status	16 bit signed integer.
32 bit Analogue input event with time	If status is OK, 32 bit signed integer, else invalid. Refer to the note following this table.
16 bit Analogue input event with time	If status is OK, 16 bit signed integer, else invalid. Refer to the note following this table.
32 bit Analogue input event without time	If status is OK, 32 bit signed integer, else invalid. Refer to the note following this table.
16 bit Analogue input event without time	If status is OK, 16 bit signed integer, else invalid. Refer to the note following this table.

32 bit Analogue output status	If status is OK, 32 bit signed integer, else invalid.
16 bit Analogue output status	If status is OK, 16 bit signed integer, else invalid.
Time and Date	The current date and time in the outstation as a floating-point number representing the number of milliseconds since Jan 1 1970.
File Identifier	A buffer containing the records requested, or an individual value of the specified type from the specified offset in a specified record.
Octet String	A buffer containing the string.

Note: For events, the return value is an array of five elements:

Index	Element
0	The address of the point in the form Object/Variation/Address.
1	Value, after correction for any status flags. If the status is on-line, not restart and not communication lost, the value will be as reported in the event; else the value will be invalid.
2	The timestamp of the event.
3	The actual value before correction for status flags
4	The status flags.

For those events that return a time, the time will be as recorded in the device. For those events that do not return a time, the time will be the local time when the driver received the message.

Time values are returned as a double, representing the time as the number of milliseconds since midnight, on 1/1/1970.

The data values that may be supplied for output objects:

Object Description	Data Value
Control relay output block	32 bit signed integer. If > 0, then action will be a latch on, close or raise. The time for the raise will be the value supplied in

	units of milliseconds. If =< 0, then action will be a latch off, open or lower. The time for the lower will be the absolute value supplied in units of milliseconds. The Latch flag, if TRUE, will cause the output to be a latch type.
32 bit Analogue output control block	32 bit signed integer.
16 bit Analogue output control block	16 bit signed integer.
Float Analogue output control block	IEEE Single Precision Floating (4 bytes)
Double Analogue output control block	IEEE Double Precision Floating (8 bytes)
Time and Date	The data value is ignored. The current date and time of the VTScada system will be sent to the outstation.
File Identifier	A buffer containing the records to be written.
Internal Indications	16 bit unsigned integer. Note that only certain values are allowed and will be detailed in the remote device's profile.

Note to VTScada programmers who are using the DNP3 driver in their code:

The following flags are available in the VTScada layer of the driver and will be updated by every response from the remote station:

Flag	Meaning
IINTimeSync	TRUE when Time sync is required by the Outstation.
IINLocalDO	TRUE when some DO's in the Outstation are in local control.
IINTrouble	TRUE when an abnormal condition exists in the Outstation.
IINRestart	TRUE when an Outstation restart has occurred.
IINEventOverflow	TRUE when Outstation event buffers have overflowed.
IINConfigCorrupt	TRUE when Outstation configuration is corrupt.

DNP object types supported

Please refer to the document, VTS-DNP3-DeviceProfile.pdf, which may be found in the Examples folder of your VTScada installation.

Object Description	Object Type	Object Variation	Read	Write
Single bit binary input all variations	1	0	✓	
Single bit binary input	1	1	✓	
Single bit binary input with status	1	2	✓	
Single bit binary input event all variations	2	0	✓	
Single bit binary input event without time	2	1	✓	
Single bit binary input event with time	2	2	✓	
Single bit binary input event with relative time	2	3	✓	
Single bit binary output all variations	10	0	✓	
Single bit binary output status	10	2	✓	
Control relay output block	12	1		✓
Binary counter all variations	20	0	✓	
32 bit binary counter	20	1	✓	
16 bit binary counter	20	2	✓	
32 bit delta counter	20	3	✓	
16 bit delta counter	20	4	✓	
32 bit binary counter without status	20	5	✓	
16 bit binary counter without status	20	6	✓	
32 bit delta counter without status	20	7	✓	
16 bit delta counter without status	20	8	✓	
Frozen binary counter all variations	21	0	✓	
Frozen 32 bit binary counter with status	21	1	✓	
Frozen 16 bit binary counter with status	21	2	✓	
Frozen 32 bit delta counter with status	21	3	✓	

Frozen 16 bit delta counter with status	21	4	✓	
Frozen 32 bit binary counter with time	21	5	✓	
Frozen 16 bit binary counter with time	21	6	✓	
Frozen 32 bit delta counter with time	21	7	✓	
Frozen 16 bit delta counter with time	21	8	✓	
Frozen 32 bit binary counter no status	21	9	✓	
Frozen 16 bit binary counter no status	21	10	✓	
Frozen 32 bit delta counter no status	21	11	✓	
Frozen 16 bit delta counter no status	21	12	✓	
Binary counter event all variations	22	0	✓	
32 bit binary counter event without time	22	1	✓	
16 bit binary counter event without time	22	2	✓	
32 bit delta counter event without time	22	3	✓	
16 bit delta counter event without time	22	4	✓	
32 bit binary counter event with time	22	5	✓	
16 bit binary counter event with time	22	6	✓	
32 bit delta counter event with time	22	7	✓	
16 bit delta counter event with time	22	8	✓	
Frozen binary counter event all variations	23	0	✓	
Frozen 32 bit binary counter event without time	23	1	✓	
Frozen 16 bit binary counter event without time	23	2	✓	
Frozen 32 bit delta counter event without time	23	3	✓	
Frozen 16 bit delta counter event without time	23	4	✓	
Frozen 32 bit binary counter event with time	23	5	✓	
Frozen 16 bit binary counter event with time	23	6	✓	
Frozen 32 bit delta counter event with time	23	7	✓	

Frozen 16 bit delta counter event with time	23	8	✔	
Analogue input all variations	30	0	✔	
32 bit Analogue input	30	1	✔	
16 bit Analogue input	30	2	✔	
32 bit Analogue input without status	30	3	✔	
16 bit Analogue input without status	30	4	✔	
Float analogue input without status	30	5	✔	
Double Analogue input without status	30	6	✔	
Analogue input event all variations	32	0	✔	
32 bit Analogue input event without time	32	1	✔	
16 bit Analogue input event without time	32	2	✔	
32 bit Analogue input event with time	32	3	✔	
16 bit Analogue input event with time	32	4	✔	
Float Analogue input event without time	32	5	✔	
Double Analogue input event without time	32	6	✔	
Float Analogue input event with time	32	7	✔	
Double Analogue input event with time	32	8	✔	
Analogue output all variations	40	0	✔	
32 bit Analogue output status	40	1	✔	
16 bit Analogue output status	40	2	✔	
32 bit Analogue output control block	41	1		✔
16 bit Analogue output control block	41	2		✔
Time and Date	50	1	✔	✔
Common Time Object	51	1	✔	
Common Time Object – unsynchronized	51	2	✔	
Time delay coarse	52	1	✔	
Time delay fine	52	2	✔	
Class 0 Event data	60	1	✔	
Class 1 Event data	60	2	✔	

Class 2 Event data	60	3	✓	
Class 3 Event data	60	4	✓	
File Identifier (older systems)	70	1	✓	✓
File Identifier (preferred object for newer sys‐tems)	70	5	✓	
Internal Indications object	80	1		✓
Octet String	110	*	✓	✓
Octet Change Object*	111	*	✓	

*Event\change objects would not normally be configured as I/O addresses. They are listed to indicate that processing support is provided them. Upon receiving an event\change object, the value of the cor‐responding static object will be updated. For example, if 110/255/6 is configured and an event is received for 111/13/6, then the tag reading the first address will be updated with the value from the event\change as they share the same index.

Static\event object pairs that follow this rule include:

- g1 Binary Input and g2 Binary Input change
- g10 Binary Output and g11 Binary Output change
- g20 Counter and g22 Counter change
- g21 Frozen Counter and g23 Frozen Counter change
- g30 Analog Input and g32 Analog Input change
- g40 Analog Output and g42 Analog Output change
- g110 Octet Strings and g111 Octet String change

The data values returned for the objects that may be read:

Object Description	Return Value
Single bit binary input	0 or 1.
Single bit binary input with status	If status is OK, 0 or 1, else invalid.
Single bit binary	If status is OK, 0 or 1, else invalid. Refer to the note following this

input event without time	table.
Single bit binary input event with time	If status is OK, 0 or 1, else invalid. Refer to the note following this table.
Single bit binary output status	If status is OK, 0 or 1, else invalid.
32 bit binary counter	If status is OK, 32 bit unsigned integer, else invalid.
16 bit binary counter	If status is OK, 16 bit unsigned integer, else invalid.
32 bit delta counter	If status is OK, 32 bit unsigned integer, else invalid.
16 bit delta counter	If status is OK, 16 bit unsigned integer, else invalid.
32 bit binary counter without status	32 bit unsigned integer.
16 bit binary counter without status	16 bit unsigned integer.
32 bit delta counter without status	32 bit unsigned integer.
16 bit delta counter without status	16 bit unsigned integer.
32 bit binary counter event without time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit binary counter event without time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit delta counter event without time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit delta counter event without time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit binary counter event with time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.
16 bit binary counter event with time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table.
32 bit delta counter event with time	If status is OK, 32 bit unsigned integer, else invalid. Refer to the note following this table.

16 bit delta counter event with time	If status is OK, 16 bit unsigned integer, else invalid. Refer to the note following this table..
32 bit Analogue input	If status is OK, 32 bit signed integer, else invalid.
16 bit Analogue input	If status is OK, 16 bit signed integer, else invalid.
32 bit Analogue input without status	32 bit signed integer.
16 bit Analogue input without status	16 bit signed integer.
32 bit Analogue input event with time	If status is OK, 32 bit signed integer, else invalid. Refer to the note following this table.
16 bit Analogue input event with time	If status is OK, 16 bit signed integer, else invalid. Refer to the note following this table.
32 bit Analogue input event without time	If status is OK, 32 bit signed integer, else invalid. Refer to the note following this table.
16 bit Analogue input event without time	If status is OK, 16 bit signed integer, else invalid. Refer to the note following this table.
32 bit Analogue output status	If status is OK, 32 bit signed integer, else invalid.
16 bit Analogue output status	If status is OK, 16 bit signed integer, else invalid.
Time and Date	The current date and time in the outstation as a floating-point number representing the number of milliseconds since Jan 1 1970.
File Identifier	A buffer containing the records requested, or an individual value of the specified type from the specified offset in a specified record.
Octet String	A buffer containing the string.

Note: For events, the return value is an array of five elements:

Index	Element
0	The address of the point in the form Object/Variation/Address.

1	Value, after correction for any status flags. If the status is on-line, not restart and not communication lost, the value will be as reported in the event; else the value will be invalid.
2	The timestamp of the event.
3	The actual value before correction for status flags
4	The status flags.

For those events that return a time, the time will be as recorded in the device. For those events that do not return a time, the time will be the local time when the driver received the message.

Time values are returned as a double, representing the time as the number of milliseconds since midnight, on 1/1/1970.

The data values that may be supplied for output objects:

Object Description	Data Value
Control relay output block	32 bit signed integer. If > 0, then action will be a latch on, close or raise. The time for the raise will be the value supplied in units of milliseconds. If =< 0, then action will be a latch off, open or lower. The time for the lower will be the absolute value supplied in units of milliseconds. The Latch flag, if TRUE, will cause the output to be a latch type.
32 bit Analogue output control block	32 bit signed integer.
16 bit Analogue output control block	16 bit signed integer.
Time and Date	The data value is ignored. The current date and time of the VTScada system will be sent to the outstation.
File Identifier	A buffer containing the records to be written.
Internal Indications	16 bit unsigned integer. Note that only certain values are allowed and will be detailed in the remote device's profile.

Note to VTScada programmers who are using the DNP3 driver in their code:

The following flags are available in the VTScada layer of the driver and will be updated by every response from the remote station:

Flag	Meaning
IINTimeSync	TRUE when Time sync is required by the Outstation.
IINLocalDO	TRUE when some DO's in the Outstation are in local control.
IINTrouble	TRUE when an abnormal condition exists in the Outstation.
IINRestart	TRUE when an Outstation restart has occurred.
IINEventOverflow	TRUE when Outstation event buffers have overflowed.
IINConfigCorrupt	TRUE when Outstation configuration is corrupt.

(Characteristics available in the Table of Type Characteristics.)

Driver Multiplexer Tags

Not counted towards your tag license limit.
Driver multiplexer (DriverMUX) tags allow you to set up redundant lines of communication between I/O tags and equipment.

Reference Notes:

Possible uses for this tag include:

- Establishing a fail-over communications route in the event that one system is lost.
  In its basic configuration, the DriverMUX tag will direct all communications to the primary driver. In the event that the connection to the primary driver is lost, communication will switch to the secondary driver.
- Upgrading to new communications equipment/drivers with zero downtime.
  In this scenario, you would install the new equipment and drivers while the existing system remains in place. The new drivers would be connected to the DriverMUX tag instead of directly to your I/O devices. When you are ready to test the new system, you can direct the DriverMUX to switch to the new, secondary system. If there are problems, then communication will immediately and automatically switch back to the primary system with no loss of data.
- Load sharing between two lines of communication.

The DriverMUX tag can be directed to use both communication drivers in either an alternating or an as-ready basis. While one driver is busy with a large packet, the other can continue to send messages.

- To provide OPC Server redundancy.
  A DriverMUX tag may be configured with OPC Client Drivers as subordinates. This will allow I/O tags to exchange data with any number of OPC Servers.

Note: If a DriverMUX tag has a Polling Driver for one of its subordinate drivers, then it must have a Polling Driver as both subordinates.
A DriverMUX tag may be subordinate to a Polling Driver.

To log an event for each switch between the primary and secondary communication path, create an alarm that monitors the state of the expression, [DriverMUX Tag Name]\CurrentSubDriver. "CurrentSubDriver" is a property of this tag, identifying the subordinate driver in use.
To log an event for failure of either communication path, create an alarm for each driver. Refer to the topic Communication Driver Alarms for details.

Related Information:
DriverMUX I/O Addressing
Communication Driver Alarms

DriverMUX Properties: ID Tab
The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.Name:Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression. AreaThe area field is used to group similar tags together. By defining an area, you make it possible to:
Filter for particular tag groups when searching in the tag browserLink dial-out alarm rosters to Alarm tags having a particular areaLimit the number of tags loaded upon startup.Filter the alarm display to show only certain areas.Filter tag selection by area when building reports When working with Parent-Child tag structures, the area property of all child

tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override. To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next restart).DescriptionTag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended. Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

DriverMUX Properties: Subordinate Drivers Tab

Select the two drivers that the multiplexer will communicate with. Either

of these may be another DriverMUX tag, with its own primary and secondary device drivers.
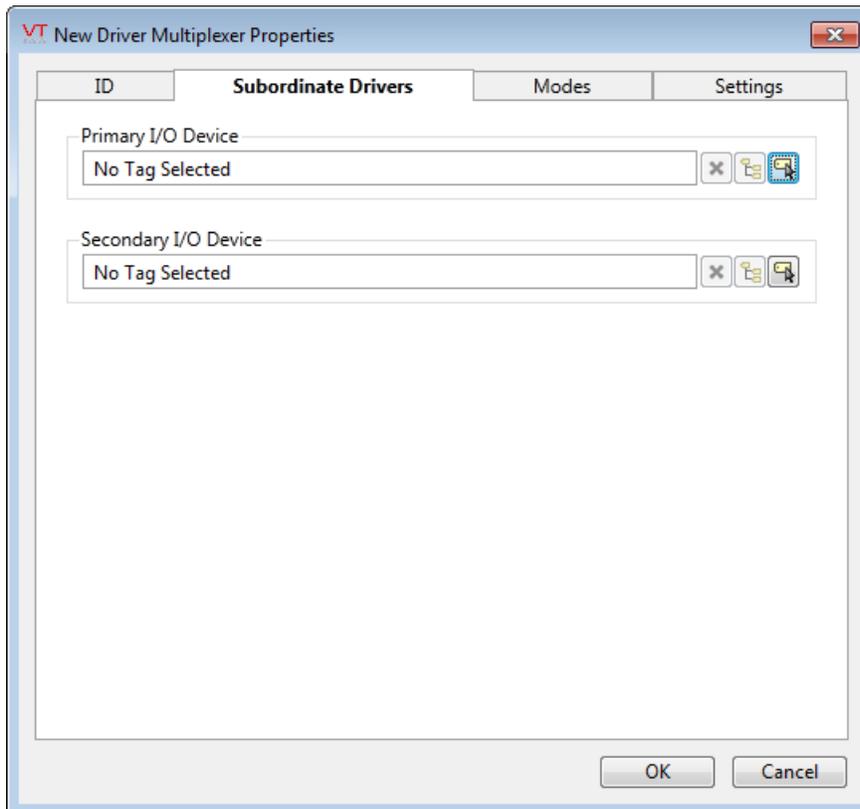
Note: If the Primary I/O Device is a Polling Driver, then the Secondary must be a Polling Driver as well.

Primary I/O Device.
   One of two device drivers that the DriverMUX tag can switch between. Depending on the purpose you intend use the DriverMUX tag for, this will normally be the device that will be used for the majority of I/O communications.

Secondary I/O Device.
   The other of the two device drivers that the DriverMUX tag can switch between. This could be any of a backup route of communication, a new system that you are switching to or even another DriverMUX tag, allowing you to configure three or more redundant lines of communication.

DriverMUX Properties: Modes Tab

Define the operation (and thereby the purpose) of the DriverMUX tag. For each of the three modes, you can select an option from a menu of

choices, or you can use a tag or expression to provide external control over the mode.

If defining a tag or expression to control a mode, note that numbering starts from 0, with the top mode in each list being the 0 mode. For example, a tag or expression controlling the Selection Mode would need to set the values 0, 1 or 2 for Automatic, Exclusive Primary and Exclusive Secondary, in that order. If the controlling tag or expression goes to invalid, then the last known good value will be used.
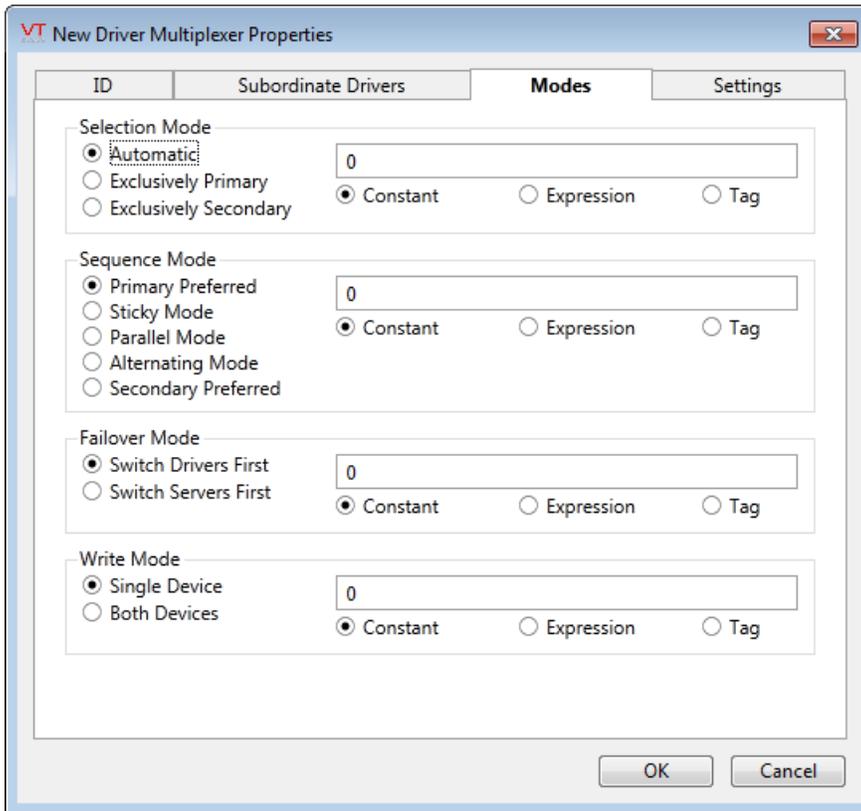
Selection Mode

Automatic enables the DriverMUX to switch between the primary and sec-ondary drivers, using rules defined in the other modes. (May be a constant, expression or tag that evaluates to 0.)

Exclusively Primary is used when only the driver defined as primary is to be used (May be a constant, expression or tag that evaluates to 1.)

Exclusively Secondary is used when only the driver defined as secondary is to be used. (May be a constant, expression or tag that evaluates to 2.)

Sequence Mode

Primary Preferred. In this mode, the primary driver is used for all communication unless it fails, at which point communication is handled by the secondary driver. When the primary driver comes back online, communication is transferred back to it. (May be a constant, expression or tag that evaluates to 0.)

Sticky Mode. Similar to Primary Preferred except that communication continues to be handled by whichever driver is in use until that driver fails. Upon failure of the active driver, communication switches to the alternate, where it remains regardless of whether communication is restored to the driver previously in use. (May be a constant, expression or tag that evaluates to 1.)

Parallel mode. In this mode, both drivers are used for communication on an as-ready basis. If one driver is busy with a larger communication packet, the other driver will be used for all packets until the first is ready for another packet. (May be a constant, expression or tag that evaluates to 2.)

Alternating mode. Both drivers are used in a strictly alternating basis. If the next driver in turn is still busy with its last communication packet, further communications will be queued until that driver is ready for another packet. (May be a constant, expression or tag that evaluates to 3.)

Secondary Preferred. In this mode, the secondary driver is used for all communication unless it fails, at which point communication is handled by the primary driver. When the secondary driver comes back online, communication is transferred back to it. (May be a constant, expression or tag that evaluates to 4.)

Failover Mode

Failover mode is of use in a remote application where backup servers are configured to handle communications in the event that a primary server fails.

- Switch Drivers First.

In the event that communication is lost with the driver in use, switch to the alternate driver. (May be a constant, expression or tag that evaluates to 0.)

- Switch Servers First.

In the event that communication is lost with the driver in use, attempt to switch to a backup server to re-establish communication with that driver before switching to the alternate driver. (May be a constant, expression or tag that evaluates to 1.)

Write Mode

By default, the driver multiplexer writes to only the active driver. If the drivers are connected to different hardware devices, then it is possible that one device might not be synchronized with regards to registers that are used as set points or other controls. This could cause unpredictable results in the case of a hardware fail over.

You have the option of forcing the DriverMUX to write to both devices.

- Single Device.

The DriverMUX tag writes to the active driver only

- Both Devices.

The DriverMUX tag sends all write requests to both the primary and secondary drivers.

DriverMUX Properties: Settings Tab

In the event of a communication failure, the DriverMUX tag will poll the failed driver at a decreasing frequency, watching for communication to be restored. This is done to allow communication to be re-established as quickly as possible in the event of a momentary outage, but to also reduce network load devoted to polling in the event of a longer outage.
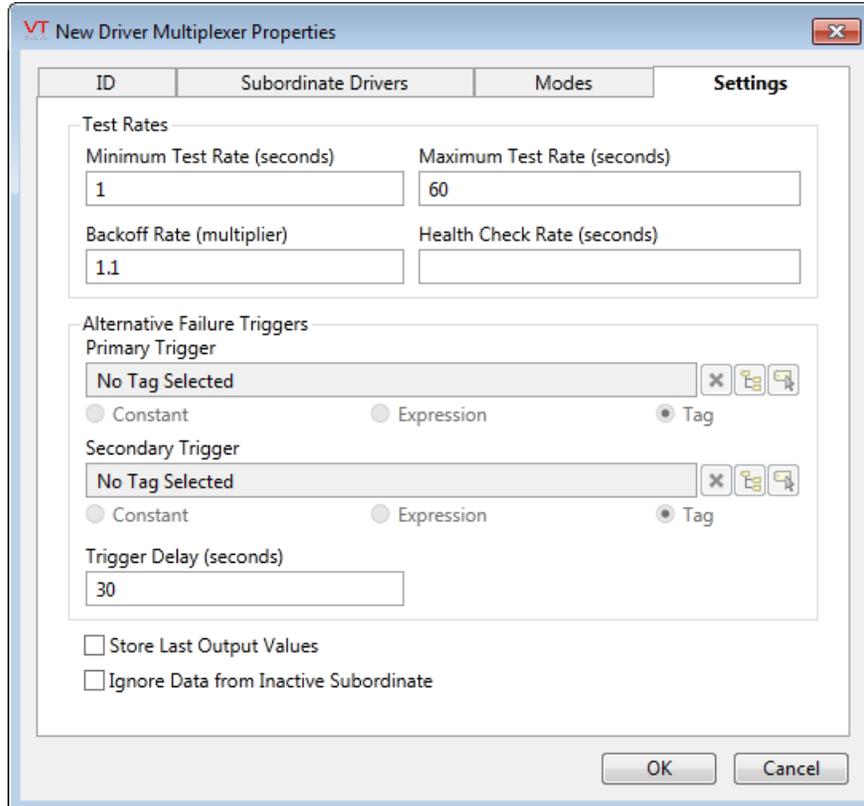
Minimum Test Rate (seconds)

The initial frequency at which checks are made to determine a failed driver's state.

Maximum Test Rate (seconds)

The longest period to wait between checks of a failed driver.

Backoff Rate (multiplier)

The time between tests is multiplied by this rate to obtain a progressively slowing frequency between tests until the Max Test time is reached.

Health Check Rate (seconds)

When the mode is set to Primary Preferred and the primary driver is in use, the secondary driver is tested at this rate to ensure its availability.

Alternate Failure Triggers

It is possible for a driver to report that it is still working, even though no data is being transferred. An example is a radio link: All devices may be in perfect working order, but if the antenna is obscured, no data will be transferred. You can use the Alternate Failure Trigger to monitor some indicator other than the driver itself to determine when the DriverMUX should switch drivers. A separate alternate trigger is provided for each driver. Use a tag or an expression that will evaluate to TRUE (1) to determine when a failure condition can be considered to have happened. An additional field, "Trigger Delay" is used to set the length of time that the alternate trigger must be true before the DriverMUX will switch drivers. For example, if monitoring for no data being received, it is necessary to wait several seconds in order to distinguish between a loss of communications and a pause between messages.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address.

If the last output values are stored, they may be re-written by either of two methods:

- Automatically, when communication is restored to the device. Auto-Rewrite must be set on the subordinate drivers that are to participate in automatic re-writing.
- Manually by way of a button press. Store Last Outputs must also be must also be set on both of the subordinate drivers. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

Ignore Data from Inactive Subordinate

Select this option to discard any data that is being received by the driver that is not the active one. The default behavior when the box is unchecked and the DriverMUX is in "Automatic" mode, is to pass all unsolicited data from both drivers to input tags.

DriverMUX Tag Widgets

The following widgets are available to display information about your application's DriverMUX tags. Note that, when the DriverMUX tag is in alternating mode, communication statistics shows only a count of switches.

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Show Statistics Button Widget

...DriverSelect Widget

...Equipment / Status Color Indicator

...Gradient Color Change Widget

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

DriverMUX I/O Addressing

The address used by an I/O tag may or may not need to change when the DriverMUX tag switches between drivers. The following three scenarios are provided to help you with I/O addressing.

Scenario 1: Two lines of communication are set up between a PLC in the field and VTS: one uses radio and the other uses a modem.

In this case, tag addressing is simple: The address is defined by the PLC device and does not change with the communication driver.
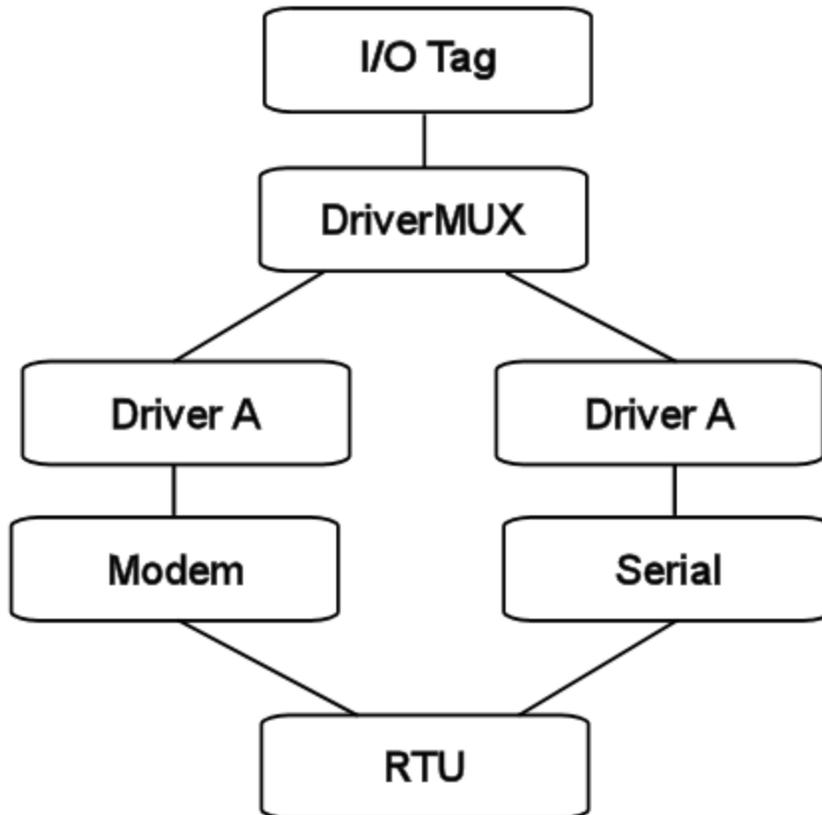
Scenario 2: Connecting to different RTUs.

For this scenario, the tag addressing used by the primary driver will be different from that used by the secondary. By way of example, perhaps one used Allen Bradley equipment and the other uses Modicon.

In this case, an I/O Tag will have to reference its intended data location by two completely different addresses. The address parameter of the I/O tag must therefore contain both addresses where each address is given

inside curly braces and are given in the order of the Primary and Sec-ondary subordinate drivers.

If (and only if) the DriverMUX is in Parallel or Alternating mode, you can make use of this address format to force an I/O tag to read from only one of the two drivers. Include both sets of braces, but leave the address blank for the driver that you do not want to read from.

Ex. {Primary Address}{Secondary Address}
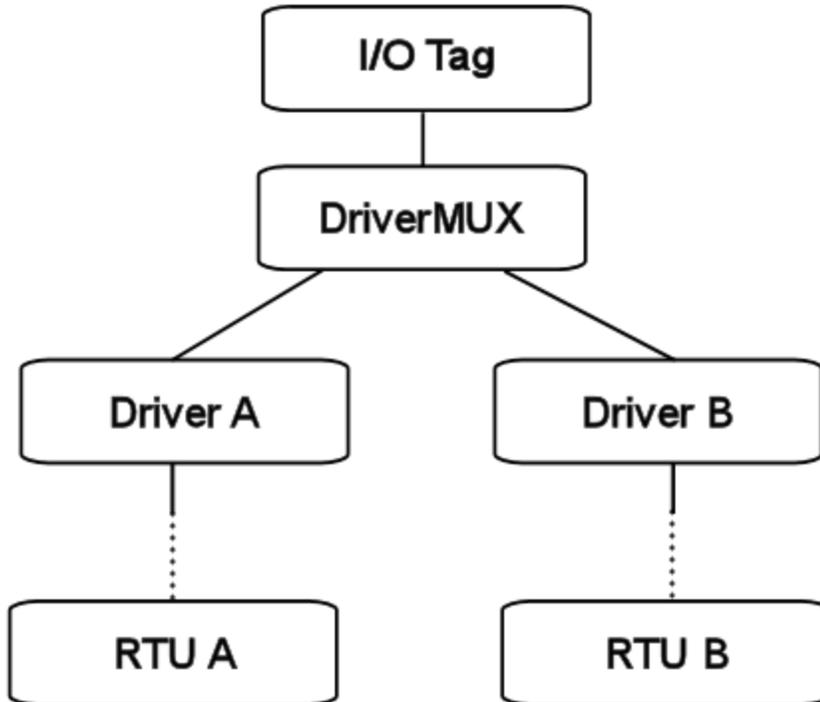
Scenario 3: A subordinate driver is also a DriverMUX tag, and each driver requires a different address for the I/O devices.

In this scenario, the addresses in each I/O tag will be nested in curly braces where the left to right order matches the primary to secondary configuration of the DriverMUX configuration:

Ex. {Primary Address}{{Secondary Address}{Tertiary Address}}

```
            ┌─────────────┐
            │   I/O Tag   │
            └──────┬──────┘
                   │
            ┌──────┴──────┐
            │  DriverMUX  │
            └──┬───────┬──┘
         ┌─────┘       └──────┐
   ┌─────┴─────┐       ┌──────┴──────┐
   │  Driver A │       │  DriverMUX2 │
   └─────┬─────┘       └──┬───────┬──┘
         │           ┌────┘       └────┐
         │      ┌────┴────┐      ┌─────┴───┐
         │      │ Driver B│      │ Driver C│
         │      └────┬────┘      └────┬────┘
   ┌─────┴─────┐ ┌───┴───┐       ┌────┴────┐
   │    TCP    │ │  TCP  │       │   TCP   │
   └─────┬─────┘ └───┬───┘       └────┬────┘
   ┌─────┴─────┐ ┌───┴───┐       ┌────┴────┐
   │   RTU A   │ │ RTU B │       │  RTU C  │
   └───────────┘ └───────┘       └─────────┘
```
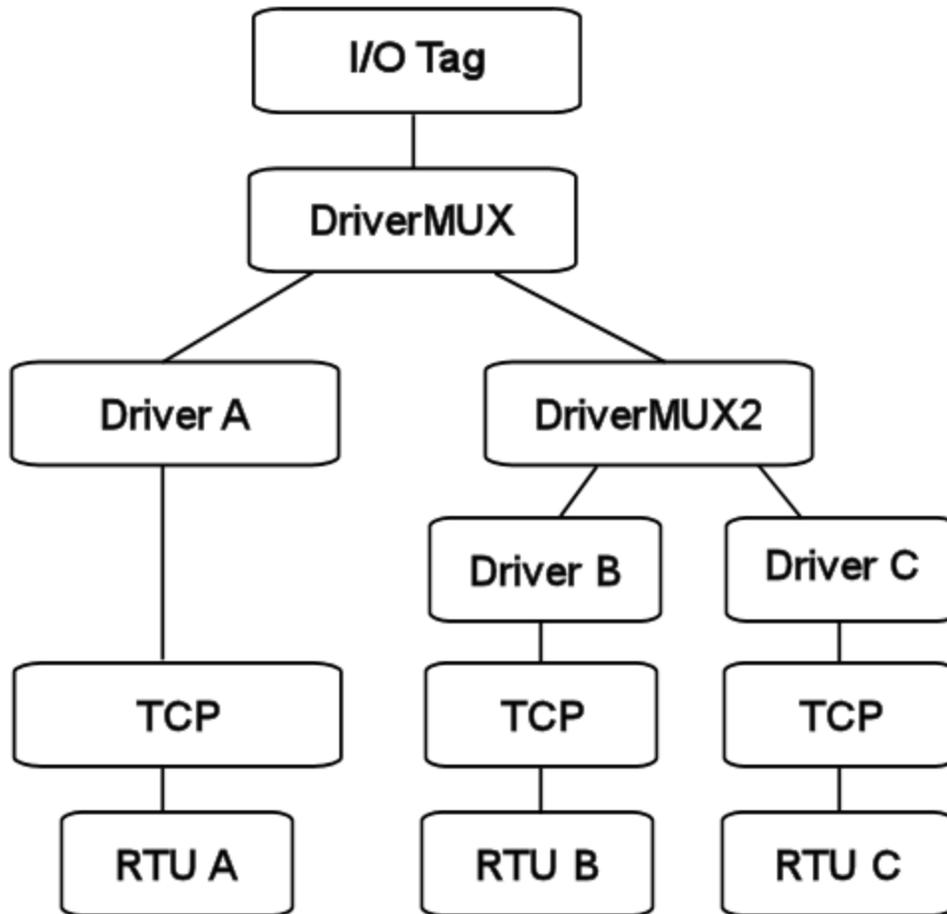
Enron Modbus Driver Tags

Not counted towards your tag license limit.

Protocol Introduction
Enron Modbus is a variation on the standard Modbus protocol that allows
for the transmission of stored historical data and alarm/event logs. The
standard read/write Modbus function codes, 01, 03, 05, 06, are sup-
ported and used to read or write real-time values. The only major dif-
ference between accessing real-time values in regular Modbus versus
Enron Modbus is the register addressing. See the addressing section for
details.

Many devices that communicate via Enron Modbus also maintain an internal historical log with periodic, hourly or daily records. Additionally, they usually have a separate log, which contains alarm and event data specific to the flow calculations that the device is monitoring. The Enron driver allows both of these logs to be read and then transferred to the VTScada historian.

Related Information:
Enron Modbus I/O Addressing

Related Application Properties:
The following property settings hold additional configuration parameters for your Enron Modbus driver:

EnronBoolMaxBlockSize – Maximum number of standard Boolean values requested in a single read.

EnronShortMaxBlockSize – Maximum number of standard short integer values requested in a single read.

EnronLongMaxBlockSize – Maximum number of standard long integer-/floating point values requested in a single read.

EnronMaxHistoryBlockSize – Maximum number of historical values requested in a single history log read.

EnronPropogateOnlyOnDataChange – If set, don't bother processing incoming data if it hasn't changed since the last read. Note that this setting does not apply to history reads or event log reads (which are al-ways processed).

EnronSharedRPC – Set to use the same RPC service for all instances of the Enron Modbus driver.

EnronContinuousHistoryRead – When set, a single history poll cycle will continue reading from the device until all new history has been read.

EnronHistoryReadStartUpDelay – Seconds to wait for Enron history tags to initialize before timing out with a comms error.

Enron Modbus Driver Properties: ID Tab
The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Enron Modbus Driver Properties: Options
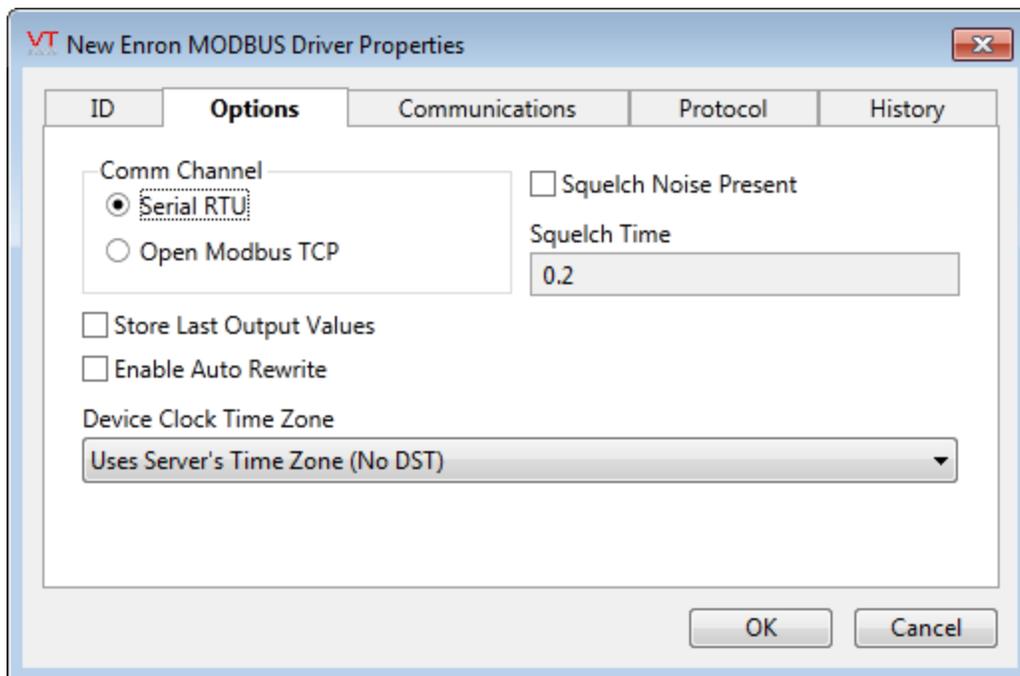
Comm Channel

Choose the format of the outgoing messages. Serial RTU sends

the messages as they are and includes a CRC[1] at the end. Open Modbus TCP prefaces the messages with a TCP header. Note that the comm channel setting can be changed independently of the port tag. For example, some devices use the Serial RTU protocol sent out over a TCP/IP port.

Squelch Noise Present

Applies when communicating over a radio modem. Check this box to enable squelching of background noise.

Squelch Time

The number of seconds to wait before clearing noise when the Squelch Noise Present option has been selected.

Store Last Output Values

If this check box is checked the last output values sent to the Enron Modbus device will be saved or stored. These can be used to refresh the output values in the event of a communications failure or a power interruption.

Enable Auto Rewrite

If checked this box causes the last stored output values to be rewritten to the device in the event of a communications loss or power failure.

Device Clock Time Zone

The time zone of the clock in the Enron Modbus device. This is used to correct the time-stamped values read from the device

---

[1]Cyclic Redundancy Check

by VTScada for storage in its historian using UTC format. Regardless of the time zone selected with this option, the driver assumes that the device does NOT adjust its clock for daylight savings time (DST).

- Uses Server Time Zone (no DST) – assumes that the device is in the same time zone as the server.
- Device Clock In UTC – the clock in the device is set to UTC
- Actual Time Zone – select the time zone where the device is located.

Enron Modbus Driver Configuration - Communications

Port

The port tag through which this driver communicates to a device, usually either a serial port tag or a TCP/IP tag.

Modbus Address

The Modbus address of the Enron Modbus device.

Time-Out Limit

The length of time, in seconds, that the driver should wait for a reply from the device.

Retries

The number of communication attempts that will be made by this driver if there is no reply to the initial message. An error will be declared after this number has been reached.

Hold

Select this option to have tags hold their last read value in the event of a communication failure with the device.

RTS Key Off Delay

Used for serial communications, this field represents the length of time in seconds that the device will wait before dropping RTS at the end of a data transmission.

Retry Delay

The length of time, in seconds, that the driver will wait between retry attempts if there is no reply to the initial message.

Enron Modbus Driver Configuration - Protocol
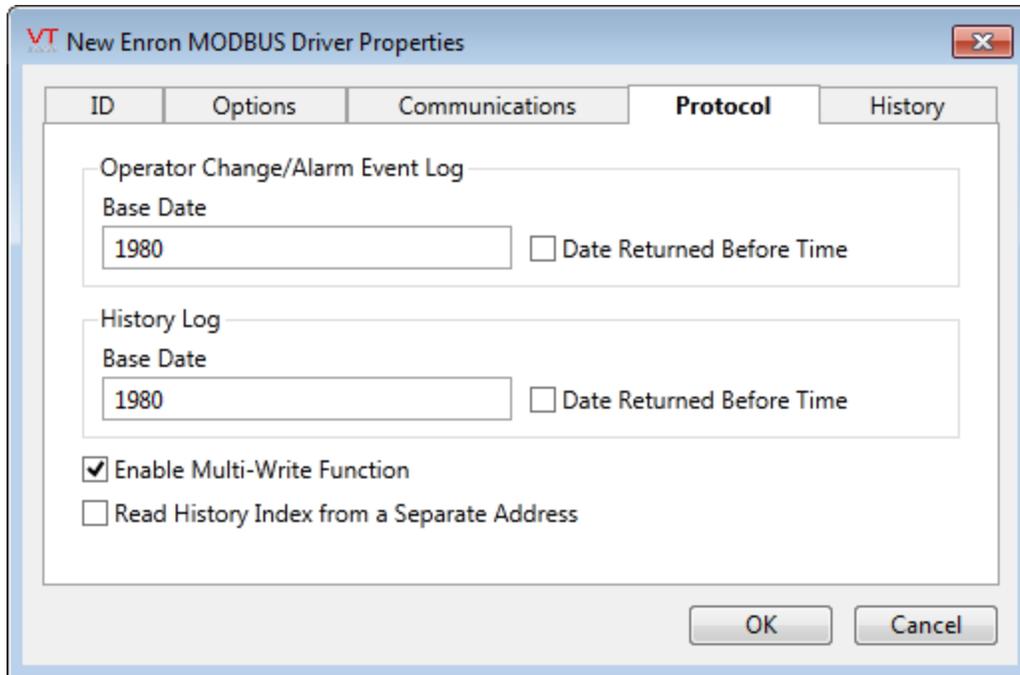
Operator Change / Alarm Event Log...

Base Date

When operator-change or alarm-event-log, time-stamped data is returned from the device, the year is provided as a two-digit number. That number is then added to this base date to produce the actual date used for the timestamp. for example, if the number returned was "34" and the base date is "1980" then the year will be recorded as 2014. This information should be available in your device's documentation.

Date Returned Before Time

Usually operator-change or alarm-event-log time-stamps from the device are returned with the time being sent before the date. If the reverse is true for your device then this box should be checked.

History Log...

Base Date

When history log time stamped data is returned from the device, the year is provided as a two-digit number. That number is then added to this base date to produce the actual date used for the timestamp (e.g. if the number returned was "34" and the base date is "1980" then the year will be re-corded as 2014. This information should be available in your device's documentation.

Date Returned Before Time

Usually, history-log time-stamps from the device are returned with the time being sent before the date. If the reverse is true for your device then this box should be checked.

Enable Multi-Write Function

Although standard Modbus multi-write function codes are not defined as part of the Enron Modbus standard, some devices can make use of them. If this is true of your device, this box may be checked.

Read History Index from a Separate Address

As described following, historical data is read from a table in the device which is ordered by an index value. By default the Enron Modbus driver will read all indices sequentially until arriving at the end of the new data (according to the datum's timestamp), however some devices track the most recent index value internally in a separate register. If this is the case with your device then this box should be checked. Note that the actual index register is set as part of the Enron History tag.

Enron Modbus Driver Configuration – History
As described in the Protocol tab – Read History Index, the Enron Modbus

protocol allows for the retrieval of historical data records stored in the device. Each record holds the value of several variables for a specific timestamp, ordered by an incrementing index value. The following table is a representation of a theoretical hourly history log.

Index	Date	Time	Temperature	Pressure	Flow
1	08/08/2014	08:00:00	21.3	87.2	48.4
2	08/08/2014	09:00:00	22.4	86.9	49.4
3	08/08/2014	10:00:00	25.1	87.3	48.7
...	...	...	...	...	...
1024	27/06/2014	19:00:00	18.9	87.4	44.0

Every hour, a new record is created inside the device. After the final history register has been filled (in this case index 1024) the next record goes back to index 1 and overwrites the record stored there.
The History tab is used to define and to track the characteristics and status of the device's history log. The "table columns" that are to be logged should be defined here. Up to 30 log tables may be defined.

Edit History Address Number

Select which of the 30 separate history addresses to configure.

Log Address

The register address in the device where the history log is stored. For the first flow run, this is usually either 701 (for hourly history) or 702 (for daily history).

Log Index Address

Some devices track the position of the index of the next record to be written. The address where this index position is stored should be entered here. This field is only used if the Enron driver's Read History Index from a Separate Address box is

checked.

Total Number of Log Indices

The total number of records stored in the history log. Once the driver has read this many indices it will start back at the first index.

Log Indices Begin at 0

Check this box if the first index of your device is 0 instead of 1.

Enron Modbus Driver Tag Widgets

The following widgets are available to display information about your Enron Modbus driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Reset Polling History Widget

...Rewrite Outputs Widget

...Next Poll Index Widget

Enron Modbus I/O Addressing

Standard Values

For standard reads and writes the address can be entered directly into the address field. Unlike with the standard Modbus driver there is no need to indicate the data type (e.g. /FLOAT) since each data type has its own predefined address range.

Register Numbers	Type	Table Name
1001–1999	Read–Write	Boolean variables
3001–3999	Read–Write	16 bit Short integer variables
5001–5999	Read–Write	32 bit Long integer variables
7001–7999*	Read–Write	32 bit Floating point variables

*Many devices allow floating point addresses above 7999.

History Logs

Note: The history logs can only be read in conjunction with an Enron History Log tag. See the Enron History Log Tag section for details on both the tag and the history storage methodology.

Analog Status tags, which store history information, should use the address structure:

```
HL:x:y
```

where x is the address of the history log (usually 701 or 702) and y is the variable's position in the returned history record (beginning with 0).

Alarm/Event Logs

An Enron Modbus device has a single log where both alarm and event information is stored. It should be noted that the alarms logged here are internal to the device and to the device's flow calculations, and as such are not necessarily related to alarms in VTScada.

Along with the timestamp, four pieces of information are returned as part of each event/alarm. Each of these pieces of information should be captured in its own analog status tag with the address:

where x is the address of the alarm log (usually 32) and y is a value between 0-3 based on the chart following.

Address Index	Description	Comments
0	Bitmap	Group of flags describing the event/alarm (see following)
1	Register	Register that was changed/that triggered the alarm
2	Data1	Previous value (event) or current value (alarm)
3	Data2	Current value (event) or not used (alarm)

The Bitmap is a group of flags whose meanings change based on whether the record is an event or an alarm. Of particular importance is bit 9, which is always a 1 for an event and a 0 for an alarm.

Bit Number	Alarm Map	Operator Event Map
0	unassigned	Fixed Value
1	unassigned	Zero Scale
2	unassigned	Full Scale
3	unassigned	Operator Entry Work Value
4	unassigned	Boolean Fixed Bit
5	unassigned	Fixed/Variable Flag
6	unassigned	Table Entry Change
7	unassigned	System Command Change
8	unassigned	unassigned
9	Operator Change Event Bit (Always 0)	Operator Change Event Bit (Always 1)
10	Lo-Lo Limit	Lo-Lo Limit
11	Lo Limit	Lo Limit
12	Hi Limit	Hi Limit
13	Hi-Hi Limit	Hi-Hi Limit

14	Rate of Change Limit	Rate of Change Limit
15	Set/Reset Alarm (1 = set, 0 = reset)	unassigned

The diagram following details the tag setup. Contrasting this with the diagram of the historical log, you can see that no intermediary tag is necessary between the driver and the status tags. However, you may still choose to add a context tag in order to group the four analog status tags together.
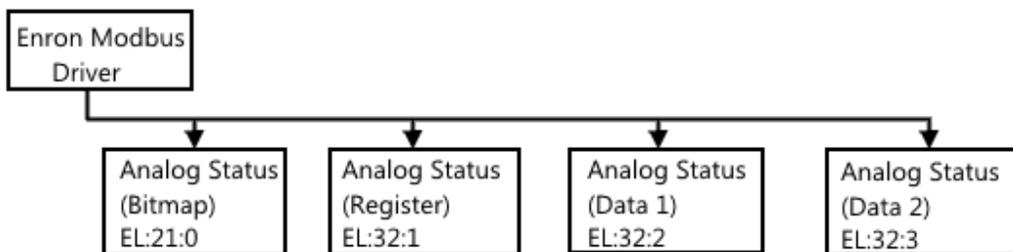
Fisher ROC Driver Tags

Not counted towards your tag license limit.
The Fisher ROC driver is used to communicate with devices that support the ROC protocol.

Related Information:
Fisher ROC I/O Addressing

Related Application Properties

ROCReadOnStartup – If true (1), the ROC driver will read all values when the system starts. Defaults to false (0).

ROCMaxParmBlockSize – Defines the maximum number of parameters that can be coalesced into a single read message when using the Parm Access accessor. Defaults to 23.

ROCDriverSharedRPC – When true (1) indicates that the same RPC

Fisher ROC Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

ROC Driver Properties: Settings Tab

The following properties are configured in the Settings tab:
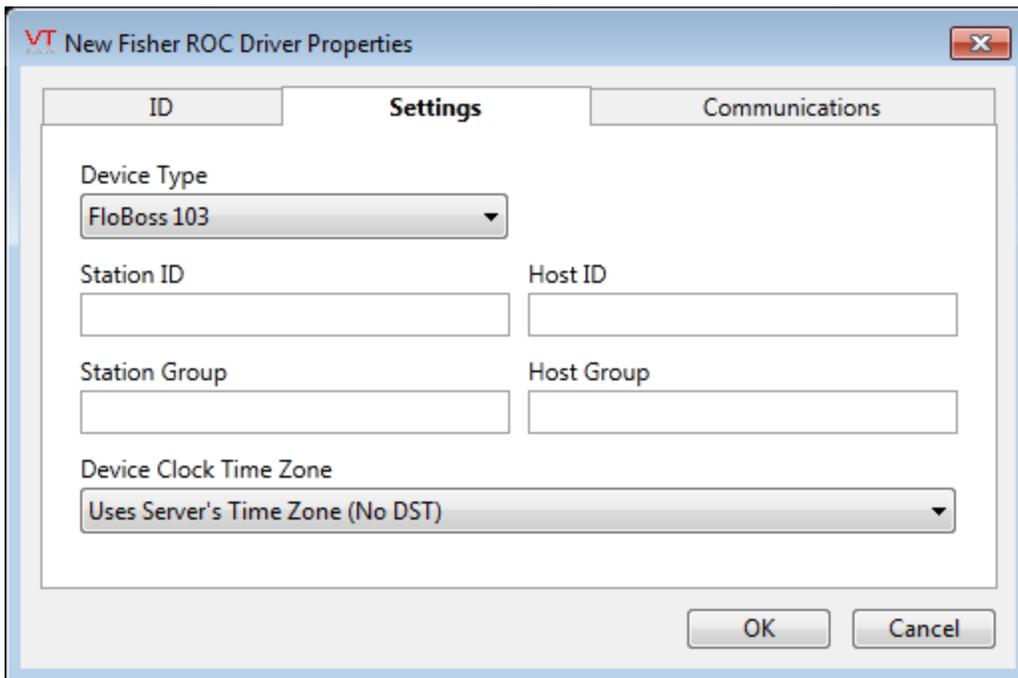
Device Type

From the drop list, select the ROC device type being

configured. Options include:

- ROC800
- FloBoss 103
- FloBoss 107
- ROC800L
- FloBoss 107 – Alternate ROCNOC

Station ID

"Station" refers to the ROC device.
The ID is the identifier used by each ROC device so that it knows which messages it is meant to respond to.

Station Group

The Group is used to further distinguish one device from another, especially in cases where there are more than 255 ROC devices on the network. For example, a message addressed to ID 100 in Group 2 will not be answered by ID 100 of Group 3.

Host ID and Host Group

"Host" refers to the VTScada application.
The VTScada driver is given an ID and a Group so that the response messages from the RTU can be delivered properly.

Device Clock Time Zone

The time zone of the clock in the device. This is used to correct the time-stamped data, read from the device by VTScada, for storage using UTC format. Regardless of the time zone selected with this option, the driver assumes that the device does *not* adjust its clock for daylight savings time (DST).

- Uses Server Time Zone (no DST) – assumes that the device is in the same time zone as the server.
- Device Clock In UTC – the clock in the device is set to UTC
- Actual Time Zone – select the actual time zone that the device is located in

ROC Driver Properties: Communications Tab
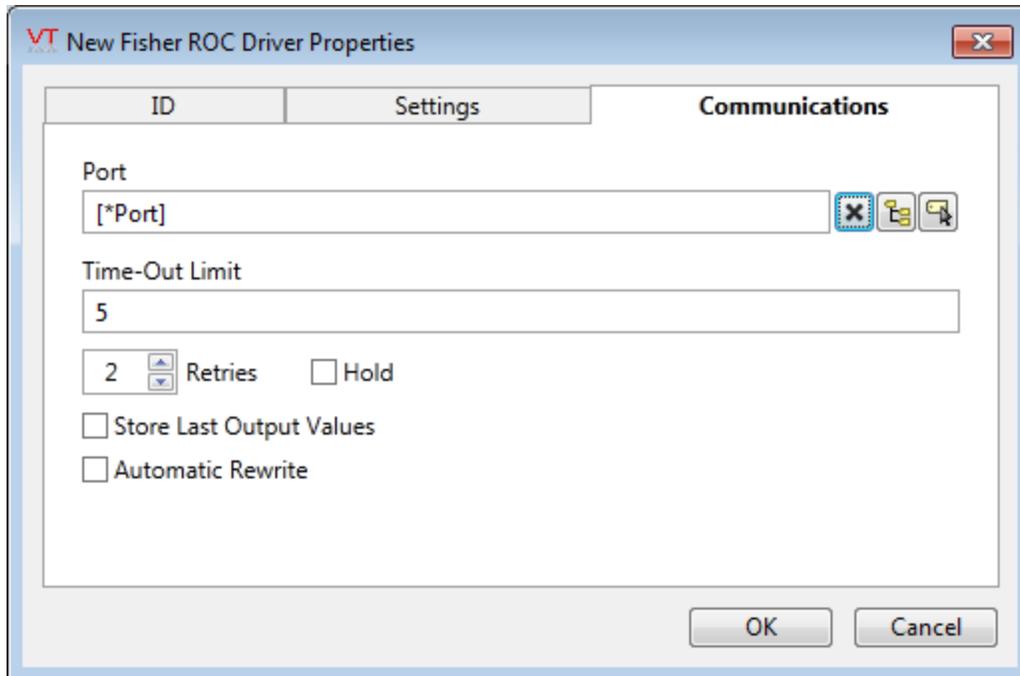
Port

The port tag through which this driver communicates with a device. This is normally a serial port tag or a TCP/IP tag.

Time-Out Limit

The length of time, in seconds, that the driver should wait for a reply from the device.

Retries

The number of communication attempts that will be made by

this driver if there is no reply to the initial message. An error will be declared after this number has been reached.

Hold

Select this option to have tags hold their last read value in the event of a communication failure with the device.

Store Last Output Values

Flag indicating that the last values written to the device should be maintained. If true, the driver will maintain the last values written for each address, allowing these value to either be re-written manually or automatically.

Automatic Rewrites

Flag indicating the driver should automatically rewrite the last value written to each address upon successful establishment of communications after a driver error occurred.

Fisher ROC Driver Tag Widgets

The following widgets are available to display information about your Fisher ROC driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

...ROC Driver History Info Button

Fisher ROC I/O Addressing

The Fisher ROC driver has an address assist module accessible from all standard I/O tags that are linked to a Fisher ROC driver. The address assist module is activated by clicking on the ROC button drawn on the I/O tab of all standard VTScada I/O tags.
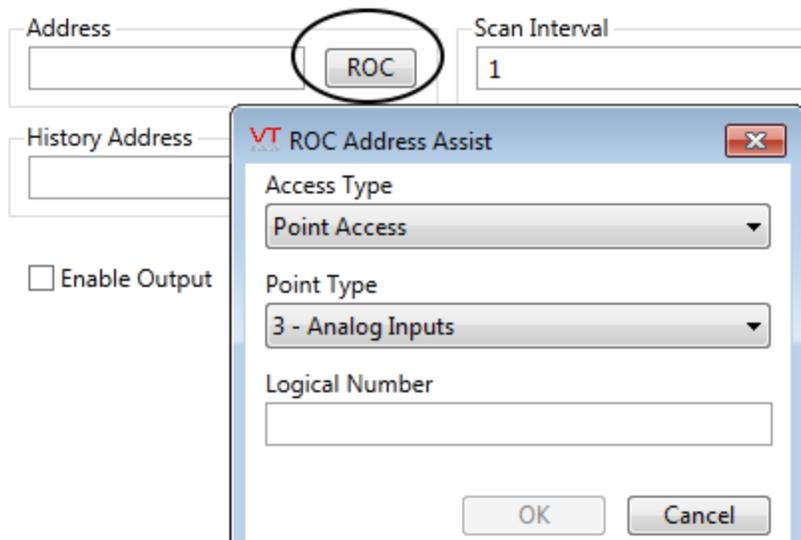
Although the Address assist module exists to help users properly select complex Fisher ROC addressing, certain circumstances may dictate that addresses be entered manually. The following table lists the access types available to I/O tags connected to the Fisher ROC Communications driver:

Access Type	Function
Point Access	Used to read individual parameters. Reads are coalesced based on point type and logic number.
Parm Access	Used to read individual parameters. Read coalescing is not ordered in any particular fashion.

Point History	Report by Exception.
Configurable Opcode Table	Used to read minute, daily, hourly, and statistical historical values form the device.
Alarm Log	Used to read the alarm log from the device.
Event Log	Used to read the event log from the device.

Point Access Address Format

Point access addresses are formatted as follows:

```
POINT:Type:Logical:Parameter
```

Where Type is the Point type, Logical is the logical number and parameter is the parameter number. As an example, in order to read AGA flow specific gravity for meter 1, the following address would be used:

```
POINT:7:0:5
```

Parm Access Address Format

Parm access addresses are formatted in the same manner as point access addresses.

```
PARM:Type:Logical:Parameter
```

Point History Address Format

The point history accessor is used to retrieve the following types of historical data:

- Minute
- Hourly
- Daily
- Statistical

Minute point history address is formatted as follows:

```
POINTHIST:MINUTE:HistoryPointNumber
```

History point number specifies the history point number whose minute data is to be accessed.

Hourly and Daily historical addressing is formatted as follows:

```
POINTHIST:DAILY:RAMArea:HistoryPoint
```

```
POINTHIST:HOURLY:RAMArea:HistoryPoint
```

RAMArea is used to specify the RAM Area, and History Point specifies the historical point number within the given RAM Area.

Statistical History addresses are formatted as follows:

```
POINTHIST:MINMAX: HistoryPointNumber:StatisticType
```

History point number specifies the history point number whose minute data is to be accessed. Statistic Type is used to specify the type of statistic to retrieve. The statistic values are as follows:

Statistic Type Address Value	Statistic
0	Current Value
1	Minimum Contract Hour Value
2	Maximum Contract Hour Value
3	Minimum Value Yesterday
4	Maximum Value Yesterday
5	Last Hour Value

Configurable Opcode Table Address Format

Configurable opcode table addresses are formatted as follows:

```
CNFOPTABLE:TableNumber:TableElement
```

TableNumber is used to specify the table number to access, and table element specifies the element number within the table to access.

Alarm Log Address Format

Alarm Log addresses are formatted as follows:

```
ALARMLOG:AlarmFieldNum
```

Where AlarmFieldNum specifies the alarm record field as shown in the following table:

Alarm Address Field Number	Statistic
0	Alarm Type
1	Alarm Code

2	Alarm tag
3	Alarm Value

Event Log Address Format

Event Log addresses are formatted as follows:

```
EVENTLOG:EventFieldNum
```

Where EventFieldNum specifies the alarm record field as indicated in the following table:

Event Address Field Number	Statistic
0	Point Type
1	Parameter Number
2	Timestamp
3	Point Number
4	Operator ID
5	Old Value
6	New Value
7	FST Number
8	Event Text
9	Event Value
10	Event Timestamp
11	Calibration Type
12	Calibration Info 1
13	Calibration Info 2

Efficient Data Collection

When accessing current parameter values, various strategies can be used in order to increase communications efficiency based on the TLP distribution of the I/O tags being accessed. the following three accessor types will be further explained in order for users to properly select the

accessor strategy to meet the particular Tag address distribution of the application.

Point Access

Will coalesce reads based on a combination of Point type and logical number. This accessor method should be used when the values of many tags, or groups of tags, being accessed have a TLP with identical point type and logical number. As an example, when accessing the following set of TLPs using the Point Access accessor:
(5:0:1, 5:0:2, 5:0:5, 5:0:6, 5:0:7, 5:0:8, 5:0:9, 5:0:10, 5:0:11, 5:0:12, 5:0:13, 5:0:14, 5:0:15, 5:1:5, 5:1:6, 5:1:7, 5:1:8, 5:1:11, 6:0:4, 6:0:5, 6:0:6, 6:0:7)
The parameters will be accessed using three separate reads.

1. Begin at 5:0:1 for 15 parameters
2. Begin at 5:1:5 for 7 parameters
3. Begin at 6:0:4 for 4 parameters

Parm Access

Will coalesce as many reads into a single message as possible. The number of parameters that can be coalesced into a single message is limited by the maximum ROC protocol message size. Given the example above:
(5:0:1, 5:0:2, 5:0:5, 5:0:6, 5:0:7, 5:0:8, 5:0:9, 5:0:10, 5:0:11, 5:0:12, 5:0:13, 5:0:14, 5:0:15, 5:1:5, 5:1:6, 5:1:7, 5:1:8, 5:1:11, 6:0:4, 6:0:5, 6:0:6, 6:0:7)
The driver will coalesce the reads into a single request. the efficiency of this accessor is lost in the fact that the TLP of each parameter being accessed must be included in the request, thus require more bandwidth in the request than simply specifying the starting TLP and the number of parameters to read. This accessor is efficient when accessing parameters which do not share common Types and logical numbers, and when the configurable opcode table has not been implemented in the device.

Configurable Opcode Table

This can be useful when the feature has been implemented in the device configuration. ROC devices allow often accessed TLPs to be assigned to particular indexes with the configurable opcode table. All values within the table can be accessed using a single read. The protocol specifies the starting index and the number of values to retrieve.

Historical Data Retrieval

Status of historical data, including alarms, events, and process data collected from the HMI is not maintained within the ROC devices. This means that the ROC device does not keep track of which data has been retrieved, and will readily allow the HMI to access all logged data within the device. In order to prevent duplicate readings within VTScada, the driver will retain the timestamp of the last data record, for each data type, and only retrieve records whose timestamps are newer than the previously retrieved record. On the initial connection to a device, the driver will have no retained timestamp and retrieve all available data.

Related Information:
Fisher ROC Driver Tags

GE Series 90 Driver Tags

Not counted towards your tag license limit.
Use this driver to communicate with GE series 90 PLCs / RTUs. Two modes of communication are supported: TCP/IP using the Service Request Transport Protocol (SRTP) or serial using the Series 90 Protocol Express (SNP-X) protocol.

Limitations
When addressing the memory in the PLC / RTU the bit reads must be byte-aligned. For example, if address %T0014 is being read a dummy tag would need to be created that would start the read at the byte boundary, %T0009. Also GE addressing starts at one, not zero.

Related Information:
GE Series 90 Addressing

GE Series 90 Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

GE Series 90 Driver Properties: Config Tab

Set the communication properties of the driver with the following fields:
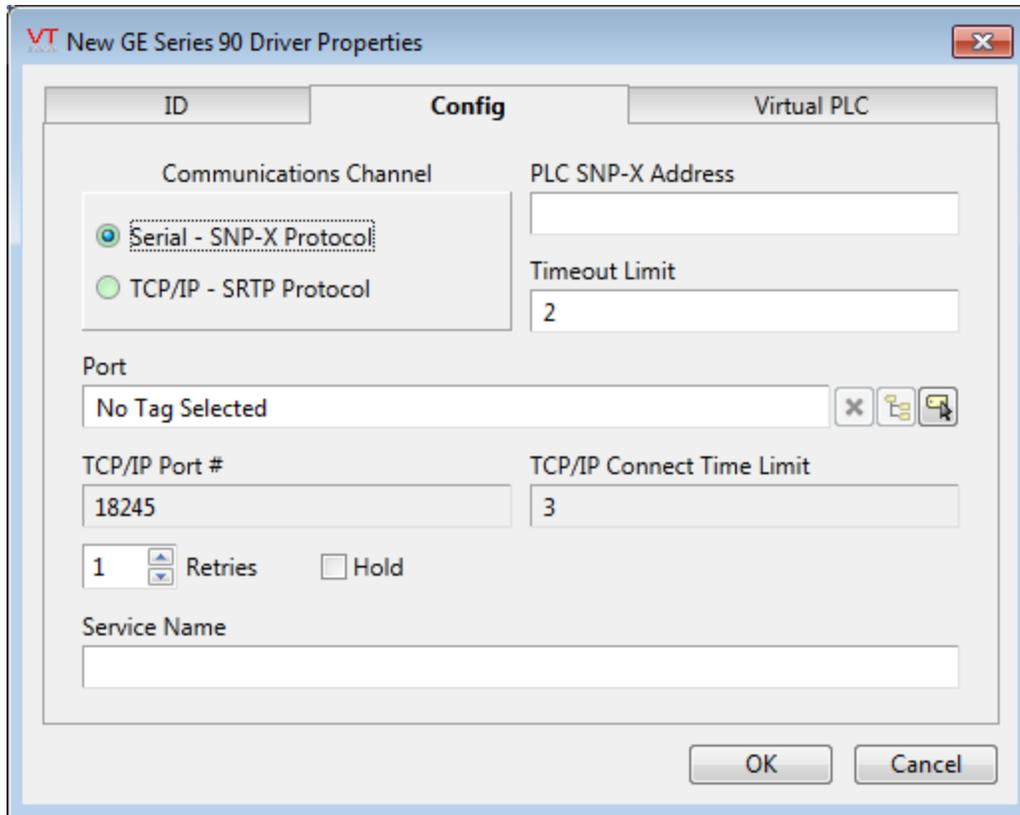
Communications Channel

Sets the type of communication the driver will use to communicate with the PLC and can be set to TCP/IP (SRTP) or Serial (SNP-X).

PLC SNP Address/PLC or TCP/IP Address

If TCP/IP communications are being used, this will be the IP address of the PLC. If Serial communications are being used this will be the station SNP-X ID.
This field cannot be left blank.

Timeout Limit

The maximum time to wait (in seconds) for a response from the remote device before retrying or generating an error.

Port

> Select the external serial or TCP/IP port for use with the SNP–X or SRTP protocols respectively.

TCP/IP Port #

> The TCP port number used by the external TCP/IP port in the case of SRTP communications.

TCP/IP Connect Time Limit

> The time limit for a successful TCP/IP connection before an error is generated.

Retries

> The number of times to retry a message before declaring an error.

Hold

> If selected, data values will be held at their last value in the event of a communication failure with the remote device. If not checked, all data read will be invalidated on error.

Service Name

> The Service Name is the name of the RPC group to which you would like your GE Series 90 driver to belong.

GE Series 90 Driver Properties: Virtual PLC Tab
The Virtual PLC tab is for setting up the driver to act like a "virtual" PLC. Specify how many memory addresses of each type to allocate to this virtual PLC.
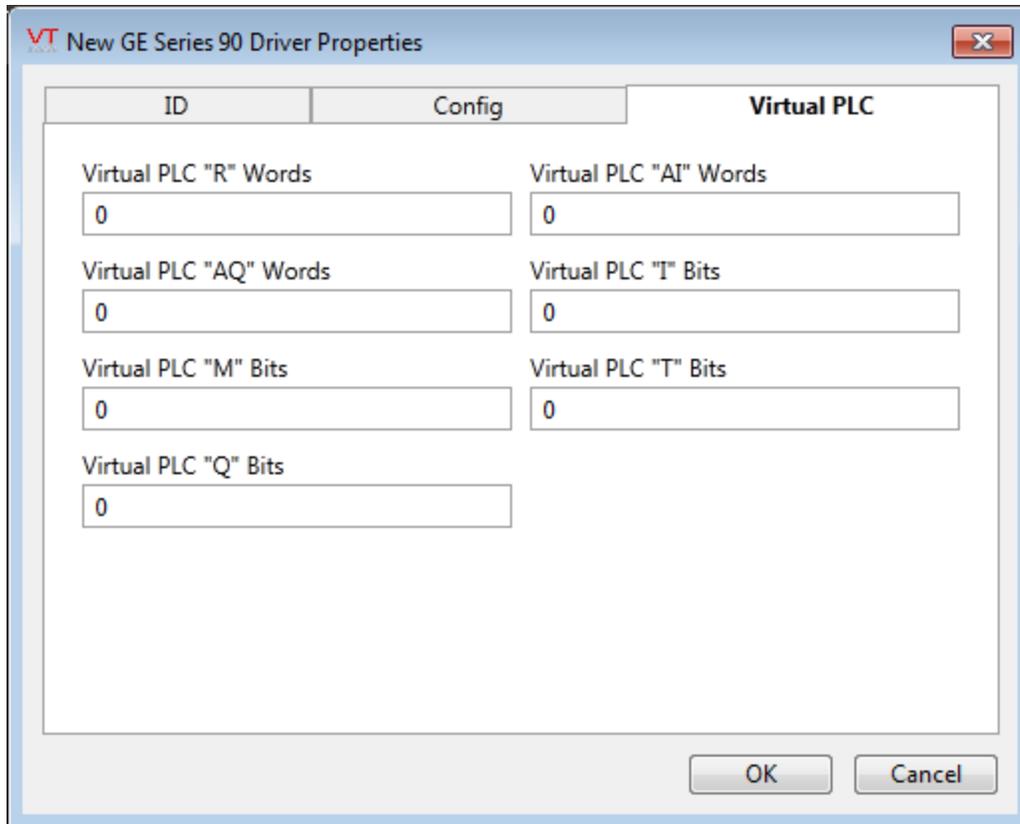
Virtual PLC "R" Words

The number of type "R" words to allocate for the virtual PLC.

Virtual PLC "AI" Words

The number of type "AI" words to allocate for the virtual PLC.

Virtual PLC "AQ" Words

The number of type "AQ" words to allocate for the virtual PLC.

Virtual PLC "I" Bits

The number of type "I" bits to allocate for the virtual PLC.

Virtual PLC "M" Bits

The number of type "M" bits to allocate for the virtual PLC.

Virtual PLC "T" Bits

The number of type "T" bits to allocate for the virtual PLC.

Virtual PLC "Q" Bits

The number of type "Q" bits to allocate for the virtual PLC.

GE Series 90 Driver Tag Widgets

The following widgets are available to display information about your Enron Modbus driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

GE Series 90 Addressing

The input / output tags addressed under the GE Series 90 Driver use the addressing system in the following table. An assist dialog is provided to help you create these addresses.
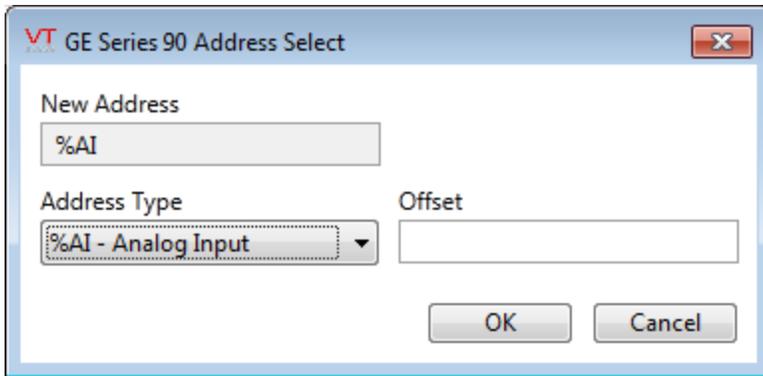
Preface the address with the register or bit type to be used. The limits on any actual addresses will depend on the configuration of the system and type of CPU used.

Address Format	Data Type
%I#	Discrete Inputs
%Q#	Discrete Outputs
%M#	Discrete Internals
%T#	Discrete Temporaries
%AI#	Analog Input
%AQ#	Analog Output
%R#	Registers
%W#	Words
%S#	Status Bits
%SA#	Status Bits
%SB#	Status Bits
%SC#	Status Bits

In this table, # represents the register or bit number, starting at one. Note that addresses configured as outputs can also be read by input tags.

Koyo Driver Tags

The Koyo driver started out as a Texas Instrument (TI) driver. Koyo purchased the technology, therefore new PLCs using this protocol bear the Koyo name. This driver works with the 200 / 300 and 400 series of PLCs.

Related Information:

...Koyo Driver Addressing

Related Application Properties

KoyoSharedRPC – Koyo shared RPC. If set, all Koyo PLCs will share the same RPC name, "KoyoDriverService". Defaults to false (0).
KoyoFailoverCount – Number of communication failures before switching to backup. Defaults to 1.

Koyo Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag

(i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Koyo Driver Properties: Configuration Tab

The Configuration Tab includes the following properties:
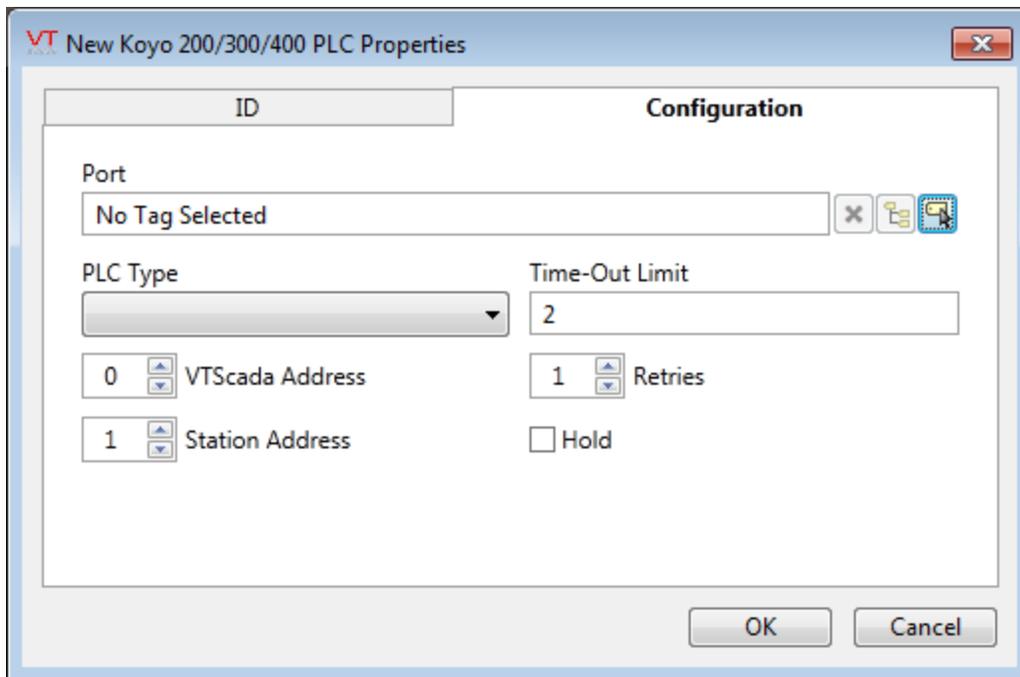
Port

This is the name of the port tag that will be used to communicate with the PLC and can be a serial port tag, a TCP/IP port tag or a UDP/IP port tag. Note that TCP/IP and UDP/IP can only be used in combination with a terminal server to convert the TCP/IP or UDP/IP to the serial communications needed to interface with the PLC at the far end. There are Koyo PLCs available that communicate directly on TCP/IP but they use a protocol that is not supported by this driver.

PLC Type

Sets the Koyo / TI PLC type with which you are communicating. Valid types selected from the drop list are: 205, 315, 315S, 320, 320S, 325, 325S, 330, 330S or 405.

Time-Out Limit

Sets the time in seconds that the driver will wait for a response

before it gives up and retries or sets an error if all retries have been attempted.

VTScada Address

Sets the address of the VTScada server that is communicating with the Koyo PLC. The valid range is 0 to 80.

Station Address

Sets the address of the PLC station with which the device will communicate. The valid range is 0 to 80.

Retries

The number of times to retry a message before declaring an error.

Hold

When checked, data values will be held at their last value in the event of a communication failure with the remote device. If not checked, all data read will be invalidated on error.

Koyo Driver tag Widgets

The following widgets are available to display information about your application's Koyo Driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Show Statistics Button Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...Koyo Driver Communication Messages Dialog

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

Koyo Driver Addressing

The input / output tags addressed under the Koyo driver use the addressing system shown here. The simple format of prefacing the address with the register or bit type to be used is utilized. The limits on any actual addresses will depend on the configuration of the system and type of CPU used. Note that bits can be extracted for word addresses by adding a "/" separator followed by the bit number. For example:
R100/0 will read bit zero of the 8 bit data register R100.

Address Type	Data Type
X	Discrete Inputs
Y	Discrete Outputs
R	8 Bit Data Registers
CT	Counter Status Bits
T	Timer Status Bits
CTR or CNT	Counter Accumulated Values
TMR	Timer Accumulated Values
SEQ	Sequencers
SR	Shift Registers
ST or S	Stages
V	Variable Memory
GX	Global Inputs

SP	Special Relays
SC	Scratchpad
D	Diagnostics

Some typical addresses would use the address types listed above followed by the register or bit number. Such as:

X10, Y20, TMR100 or V4010

MDS Diagnostic Driver Tags

Not counted towards your tag license limit.

The MDS diagnostic driver tags enable users to read statistics from any given MDS radio.

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

MDS diagnostic driver tags are required only in applications where you wish to gather data about an MDS radio. For example, a common arrangement might be an MDS radio connected to your PC, and a secondary MDS radio connected to an RTU or PLC at a remote location. In order to collect data from the RTU or PLC, your VTScada application could use an appropriate communication driver tag (e.g. Allen-Bradley or Modicon), however, if you wish to gather data about either MDS radios in such a scenario (e.g. the temperature of an MDS radio), your application will require an MDS diagnostic driver tag.

Related Information:

MDS Diagnostic Driver I/O Addressing

MDS Diagnostic Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

MDS Diagnostic Driver Properties: Serial Tab

The Serial tab of the MDS diagnostic driver tag properties folder is used to indicate the serial port to which the MDS radio associated with this tag is connected.
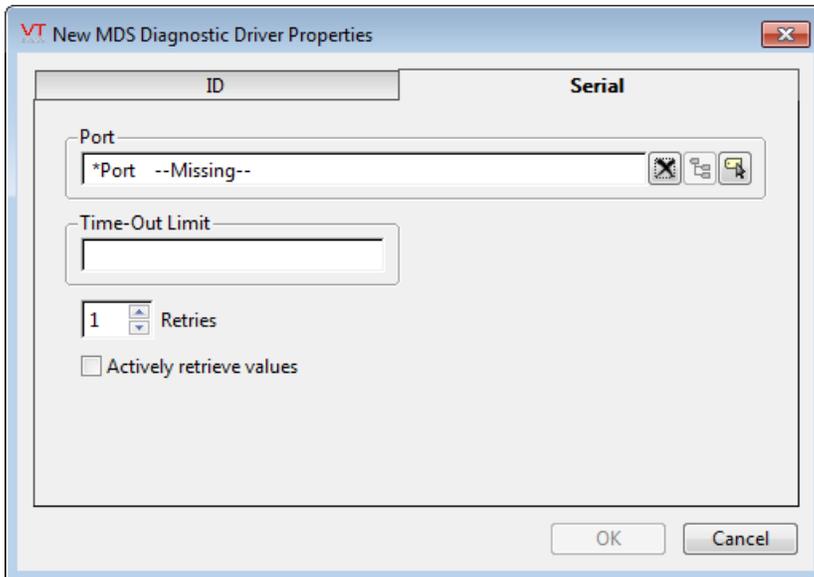
Port

The Port field enables you to select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag.

In order to connect to MDS radio port with VTScada, you can use one of several different hardware solutions:

- Serial port cable connected directly.

For this case, configure and select a Serial Port tag.

- Stand alone serial modem connected to the serial port on the radio.

VTScada has its own modem that is used to dial up the modem connected to the radio. For this option, configure and select a serial port that is using a modem.

- Ethernet terminal server connected to the serial port on the radio.

VTScada is connected to the same network as the terminal server and can connect to it with either a TCP/IP or UDP/IP port tag

Any type of VTScada port tag may be used, but it must match the hardware connection to the radio.

Time-Out Limit

Set the time in seconds or fractions of a second that this driver should wait for a reply from the remote unit.

Retries

Select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Actively Retrieve Values

Indicates that messages should be sent out in active mode, rather than passive mode.

If the Actively Retrieve Values check box is selected, messages will be sent out in active mode (i.e. a request for data from the MDS radio will be immediately sent).

If the Actively Retrieve Values check box is not selected (default), the data will be sent out in passive mode (i.e. a request for data from the MDS radio will be held until another request for data from the associated RTU or PLC is made).

MDS Diagnostic Driver Tag Widgets

The following widgets are available to display information about your application's MDS diagnostic driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

MDS Diagnostic Driver I/O Addressing

MDS radio diagnostic data can be obtained by reading from predefined addresses. Supported models are: TransNET, x790, x710, and SD series radios.
An analog or digital tag should be configured for each item of diagnostic information of interest. Note that not all radio models support all diagnostic data. Please refer to the manufacturer's manuals to find the available diagnostic data for individual models and the proper radio configuration to enable diagnostics.
The following predefined addresses are available for analog tags:

RSSI.VAL	Received signal strength, in dBm. Analog tags with this address should use the following scaling:	
	Unscaled Min	136
	Unscaled Max	206

	Scaled Min	−120
	Scaled Max	−50
SNR.VAL	Signal to noise radio, in dBm.	
TEMP.VAL	Internal radio temperature, in degrees C.	
VOLT.VAL	Primary supply voltage in mV.	
CUR.VAL	Primary supply current in mA.	
SVOLT.VAL	Secondary power supply #1 voltage, in mV.	
SVOLT2.VAL	Secondary power supply #2 voltage, in mV.	
SVOLT3.VAL	Secondary power supply #3 voltage, in mV.	
TXFPOW.-VAL	Transmit forward power, in dBm.	
TXRPOW.-VAL	Transmit reverse power, in dBm.	
LNACUR.-VAL	Low Noise Amplifier current, in mA.	
The following predefined addresses are available for digital tags (0 = OK, 1 = BAD):		
RSSI.ERR	RSSI status	
SNR.ERR	SNR status	
TEMP.ERR	Temperature status	
CUR.ERR	Primary supply current status	
SVOLT.ERR	Secondary power supply #1 voltage status	
SVOLT2.ERR	Secondary power supply #2 voltage status	
SVOLT3.ERR	Secondary power supply #3 voltage status	

TXFPOW.ERR	Transmit forward power status
TXRPOW.ERR	Transmit reverse power status
LNACUR.ERR	Low Noise Amplifier current status

The address structure is <radio unit #>:<address>.

For example, to read supply voltage on radio 3258, the address is "3258:VOLT.VAL".

Modbus Compatible Device Driver Tags

Not counted towards your tag license limit.

The Modbus Compatible Device driver provides an interface to a Modicon PLC.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Any of the following communication channels may be used:

- Serial RTU
- Serial ASCII
- SA-85 I/O card for ModBus Plus communications
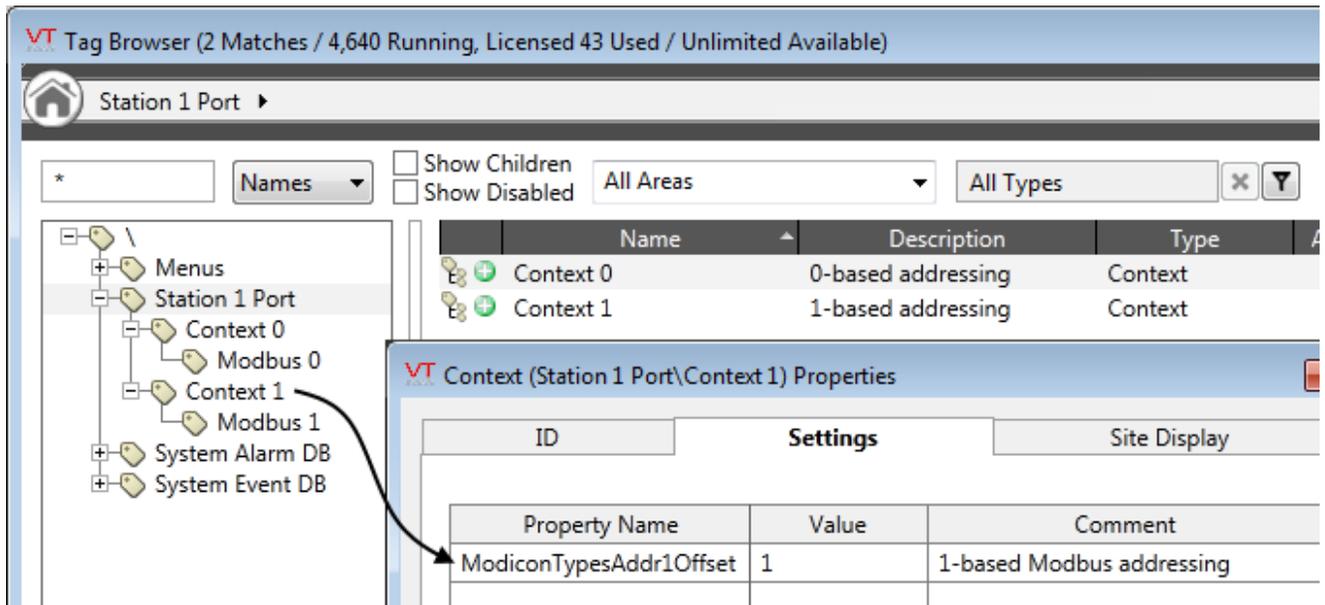- Embedded TCP/IP
- Open Modbus TCP

This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the Options tab to decide whether this feature should be used in your application. If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer

to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

The number of bytes in a Modbus message is limited by the setting: ModiconVTSMaxBlock. By default, this is 125.

When this driver is used in combination with a TCP/IP tag, the standard TCP port is 502.

By default, addressing is zero-based, as set by the application property ModiconTypedAddr1Offset. If you change this property, it will be changed for all Modbus-compatible drivers in the application. If you need some drivers to be zero-based and others to be one based, you can achieve this by creating a parent Context tag for each driver, adding a property named ModiconTypedAddr1Offset to that Context tag. The child driver tag will take the value in the parent Context tag instead of the global property value.

Modbus Compatible Device Properties

Properties Folder Tab	Property	Value
Options Tab	Comm Channel	Serial RTU
Serial Tab	PLC Address	1

Serial Tab	Time-Out Limit	0.5 (Make it shorter if Scan Interval is going to be very short)
Serial Tab	Port	COM2 (name of Serial Port tag)
Serial Tab	Retry Delay	0
Serial Tab	Hold	deselected
Serial Tab	Retries	2
Serial Tab	RTS Key Off Delay	0

Related Information:

...Modbus Compatible Device I/O Addressing

...Modbus Plus PLC Addressing

...SCADAPack History Read

...History Data for a TBox RTU

...Koyo Address Translation

...See also: ModiconVTSMaxBlock in the VTScada Admin Guide

Modbus Compatible Device: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).
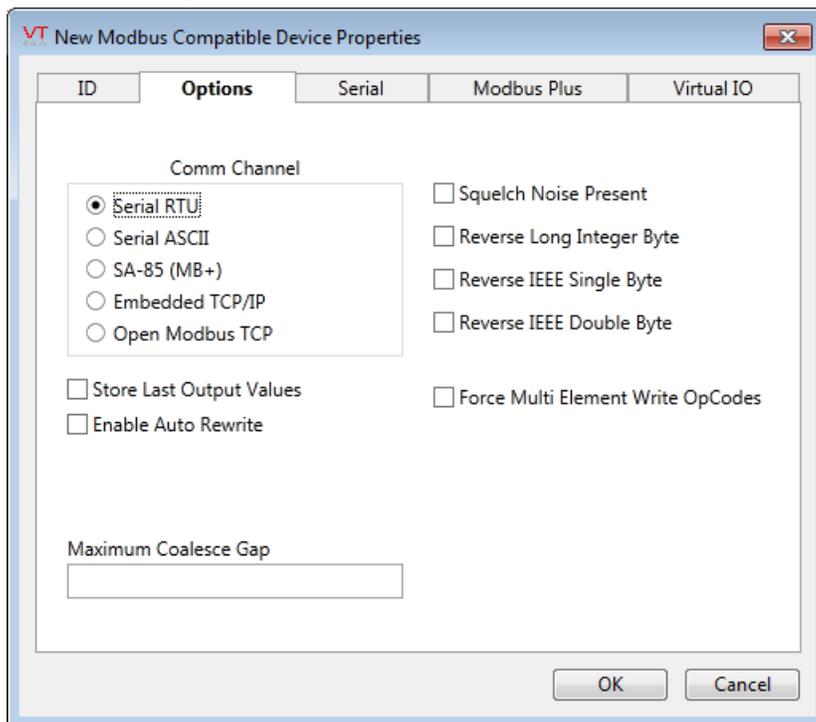
Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Modbus Compatible Device: Options Tab

The Options tab of the Modbus Compatible Device tag properties folder can be used to set byte swapping for long integer, IEEE single or IEEE double bytes. If this driver is communicating over a modem, then the Options tab is also used to indicate that squelch noise should be suppressed, if present.

Comm Channel

The Comm Channel section enables you to select a radio button indicating the communication channel and protocol to use for this Modbus Compatible Device driver. The default communications protocol is Serial RTU. This will be one of:

- Serial RTU Select to use the RTU mode of the Modbus serial communications standard.

- Serial ASCII Select to use the ASCII mode of the Modbus serial communications standard.
- SA-85 (MB+) Select to use the Modbus Plus communications protocol via a Modicon SA-85 card.
- Embedded TCP/IP Select to use the Embedded TCP/IP protocol.
- Open Modbus TCP Select to use the Modicon Open Modbus/TCP specification to receive Modbus protocol over Ethernet TCP/IP.

Squelch Noise Present

Applies when communicating over a modem. Check this box to enable the squelching of background noise. By default, the Squelch Noise Present check box is not selected.

Reverse Long Integer Byte

Select whether or not byte swapping should occur for long integer values. By default, the Reverse Long Integer Byte check box is not selected.

Reverse IEEE Single Byte

Select whether or not byte swapping should occur for IEEE single byte values. By default, the Reverse IEEE Single Byte check box is not selected.

Reverse IEEE Double Byte

Select whether or not byte swapping should occur for IEEE double byte values. By default, the Reverse IEEE Double Byte check box is not selected.

Force Multi Element Write OpCodes

Certain hardware requires data to be written using multi-element OpCodes (15 and 16) even though the actual data is

single-element (OpCodes 5 & 6). By checking this box, you can ensure that this driver instance is able to communicate with devices that require the multi-element codes.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one. If the last output values are stored, they may be re-written by either of two methods:
- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated. This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.

Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Maximum Coalesce Gap

Use only if the Modbus compatible device does not support reads from addresses that are not explicitly declared in the device. VTScada will attempt to coalesce reads from a range of addresses, which may lead to an error with these devices.
If left blank, there is no effect on driver coalescing.
Set to zero to allow for devices that cannot read from addresses that are not explicitly declared.
Set to a number greater than zero to fine-tune read coalescing. The gap is measured in units of two-byte words. Warning: when using low maximum coalesce gap values, it is important to avoid unnecessary gaps in VTScada tag addresses.

> Note: if this value is set, driver response time may be slowed due to the increased number of read blocks.

See also: VTSMaxBlock in the VTScada Programmer's Guide

Modbus Compatible Device: Serial Tab
The Serial tab for the Modbus Compatible Device Driver tag properties folder consists of attributes used to identify and establish a connection to the hardware.

Port

> Enables you to select the Serial Port tag or TCP/IP tag you wish
> to be associated with this driver tag. A Serial Port tag opens a
> serial port to enable communications between VTScada and
> your PLC or RTU. A TCP/IP tag enables you to connect to a
> series of hosts, allowing you to transmit data across a network
> or over the Internet.
> The Port field can be cleared using the X button that appears
> to its right.

IP Network Listener Group

> Configure only if this driver is to accept in-bound TCP or UDP
> communications. Select an existing IP Network Listener tag
> using the drop-down list.
> If the IP Network Listener is configured, this driver must use
> Open Modbus TCP, configured on the Options tab.

IP Allow

Optional. Misconfiguration of this field is a common source of problems.

Used in connection with an IP Network Listener tag. General IP address filtering should be set on the IP Network Listener, as this will be more efficient. The list of allowed IP addresses in the driver is intended to prevent misconfigured devices from interfering with other devices.

Both filters (that in the Network Listener and this one) can be configured at the same time. The filter in the IP Network Listener will be applied when the device first connects and the local filter will be applied after a specific driver instance has been identified.

PLC Address

The station address of the Modicon programmable logic controller. By default, the PLC Address field is set to 1.

An expression may be used to provide the address if needed. This may be required on a network where the address varies depending on the workstation where the tag is running.

Note that, to create a valid expression for an address that contains non-numeric characters, you must enclose the address in quotation marks. e.g. "31.0.0.0" See: Modbus Plus PLC Addressing.

Time-Out Limit

Set the receiver time-out limit in seconds or fractions of a second. This is the length of time that this driver should wait for a reply from the PLC or RTU.

Port

Enables you to select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag. A Serial Port tag opens a serial port to enable communications between VTScada and your PLC or RTU. A TCP/IP tag enables you to connect to a series of hosts, allowing you to transmit data across a network or over the Internet.
The Port field can be cleared using the X button that appears to its right.

Squelch Time

Sets the amount of time (in seconds or in fractions of a second) to wait before clearing noise. Enabled only when Squelch Noise Present has been selected on the Options tab.

Retries

The Retries spin box enables you to select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Hold

The Hold check box selects whether this driver should hold data from the PLC or RTU in the event of a communications failure.
If the Hold check box is selected, the last received value from the PLC or RTU will be held.
If the Hold check box is not selected, the data will be invalidated in the event of a communications error.
By default, the Hold check box is not selected.

Retry Delay

The Retry Delay field enables you to specify the amount of time (in seconds or fractions of a second) that the driver will wait between attempts to repeat a transmission of data if the previous attempt(s) have failed.
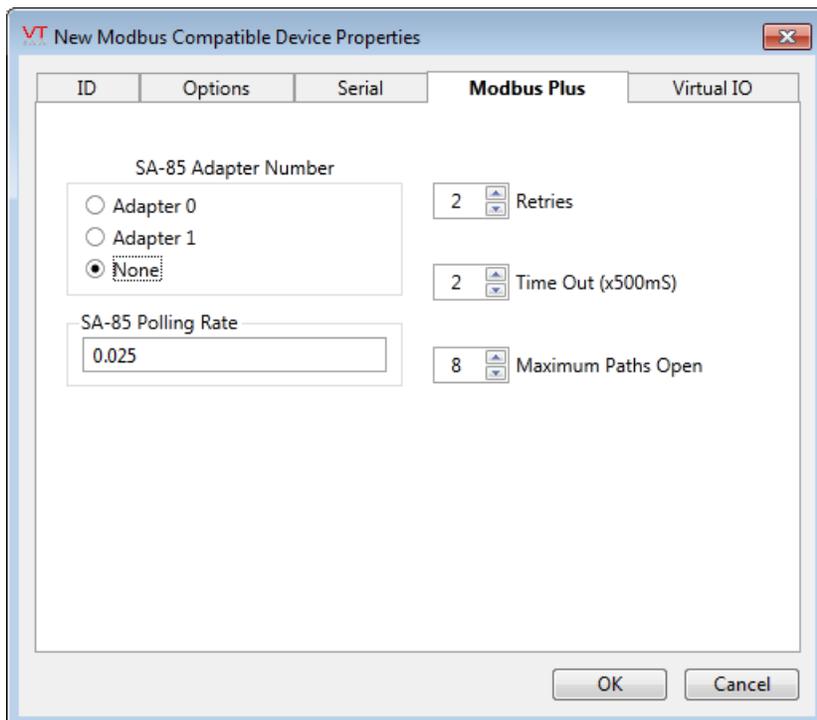
By default, the Retry Delay field is set to 0 (indicating that there should be no delay between retry attempts).

RTS Key Off Delay

The RTS Key Off Delay field represents the amount of time (in seconds) that this Modbus Compatible Device will wait before dropping RTS at the end of a data transmission.

Modbus Compatible Device Properties: Modbus Plus Tab

The Modbus Plus tab of the Modbus Compatible Device tag properties folder is used when an SA-85 I/O card for Modbus Plus communications has been configured.

SA-85 Adapter Number

The SA-85 Adapter Number section enables you to specify the number of the adapter if you have installed an SA-85 network adapter card for Modbus Plus communications networks. You can specify adapter 0, adapter 1, or none by selecting the associated radio button. The default setting for the SA-85 Adapter Number section is None.

SA-85 Polling Rate

Set the rate in seconds or fractions of a second at which you wish the SA-85 network adapter card to be polled. The default SA-85 Polling Rate setting is 0.025.

Retries

Select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Time Out (x500mS)

Specify the message response time limit that the driver will wait for a reply from the PLC. This time limit may be set from 1 to 20 increments. The default setting for the spin box is 2 (i.e. two 500 millisecond increments, or 1 second).

Maximum Paths Open

Set the maximum number of Modbus Plus sessions that can be opened simultaneously by the installed SA-85 network adapter. The Maximum Paths Open spin box may be set from 1 to 8. The default setting for the Maximum Paths Open spin box is 8.

Modbus Compatible Device: Virtual I/O Tab

These properties are used to configure a virtual PLC in the memory of

your PC.

Holding Coils 0xxxx (0-999)

Enter the number of 00000-series holding coils to be assigned to the virtual PLC.

Input Coils 1xxxx (0-9999)

Enter the number of 10000-series input coils to be assigned to the virtual PLC.

Input Registers 3xxxx (0-9999)

Enter the number of 30000-series input registers to be assigned to the virtual PLC.

Holding Registers 4xxxx (0-9999)

Enter the number of 40000-series holding registers to be assigned to the virtual PLC.

Listen on Channel x

> Select the channels of the Modbus Plus network to which the virtual PLC should listen. Please note that these do not apply to serial or TCP/IP connections.

Modbus Compatible Device Widgets

The following widgets are available to display information about your application's Modbus Compatible Device Driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

Modbus Compatible Device I/O Addressing

The allowable range of addresses is dependent upon the device with which you are communicating. For example, a PLC may only have 500 holding registers, and thus it can only address either 40001 through

40501, OR 400001 through 400501 (whichever you prefer), even though the protocol allows for addresses much higher than this.

The following table identifies the value ranges for Modicon outputs, inputs, input registers, and holding registers.

I/O Type	Text Prefix	5 digit address	6 digit address
Holding Coil	HC1 – HC65536	00001 – 09999	000001 – 065535
Input Coil	IC1 – IC56636	10001 – 19999	100001 – 165535
Input Registers	IR1 – IR65536	30001 – 39999	300001 – 365535
Holding Registers	HR1 – HR65536	40001 – 49999	400001 – 465535
Data Logger	DL1:1 – DL16:8		

Note: when using numeric addressing, the leading digit is significant as it determines the I/O type. Any 4 digit numeric address will be used as a Holding Coil address.

Some applications may be configured using the text prefix addresses that are 0-based, i.e. the first holding coil was at HC0. To permit these applications to work correctly, an application property flag (Modicon-TypedAddr1Offset) is used to set the base for text prefix addresses.

By default, addressing is zero-based, as set by the application property ModiconTypedAddr1Offset. If you change this property, it will be changed for all Modbus-compatible drivers in the application. If you need some drivers to be zero-based and others to be one based, you can achieve this by creating a parent Context tag for each driver, adding a property named ModiconTypedAddr1Offset to that Context tag. The child driver tag will take the value in the parent Context tag instead of the global property value.

Bit Addressing

An individual bit in a register (either an Input Register or a Holding Coil) may be read or written (for Holding Coils) by appending the address with "/x" where x has a value from 0 to 15 and indicates which bit is to be used. For writes, a read\modify\write cycle is used on the register as the Modicon protocol does not have a bit operation function.

If you are working with long integers, append the text "/sdword" after the bit number. For example: "40050/1/sdword".

Float Addressing

If required, you can use floating point addressing with VTScada. The process will require an adjustment to the way the address is provided in order to reach the correct register.

Modicon registers are 16 bits wide. In order to get a float address you will need to read 2 sequential registers to get 32 bits. To do this, you will need to know the lowest register address of the pair. Note that this may be odd or even and differs by device.

The address will be formed by entering the lowest address of the pair with the suffix of /FLOAT. e.g. IR9/FLOAT will read the register pair at IR9 and 10, and return the combined value as a float.

Note that driver options determine how the 4 bytes in the combined register pair are ordered to form the float as this will differ by device.

Related Information:

...Data Suffixes for Tag I/O Addressing – A list of all available data type suffixes that can be used for tag I/O addresses.

...SCADAPack History Read

...History Data for a TBox RTU

...Koyo Address Translation

SCADAPack History Read

The VTScada Modbus supports the reading of data from data loggers created by Control Microsystems RTU's that have been configured to store data using DLOG functions.

For example, you may have a Control Microsystems driver which is saving data once per minute using DLOG functions. To configure VTScada to read the data once per hour, you can use the SCADAPack History Read as described in the following notes.

Configuring the Control MicroSystems RTU

A necessary first step is to program the RTU driver to use the DLOG functions. The DLOG functions can log 8 fields from any of 16 loggers.

For each set of 8 fields, VTScada requires one to be configured as a "Date and Time" field. It does not matter which field is used for this purpose or even whether more than one is configured as "Data and Time". VTScada will simply use the first one found in numerical order.

For the other fields, VTScada supports any of the data types that Control MicroSystems uses.

Configuring VTScada

To retrieve data from history, you must use either the Analog Status tag or the Digital Status tag. These two tags are able to read both current data and historical data.

Using Analog Status tags

The format for the history address must be in the format "DLx:y" where x is the number of the logger and y is the field within that logger.

Thus, to read from field 3 of logger 1 the address would look like "DL1:3".

The history scan interval is measured in seconds and can be any integer value you wish. For example, to read every hour you would enter 3600.
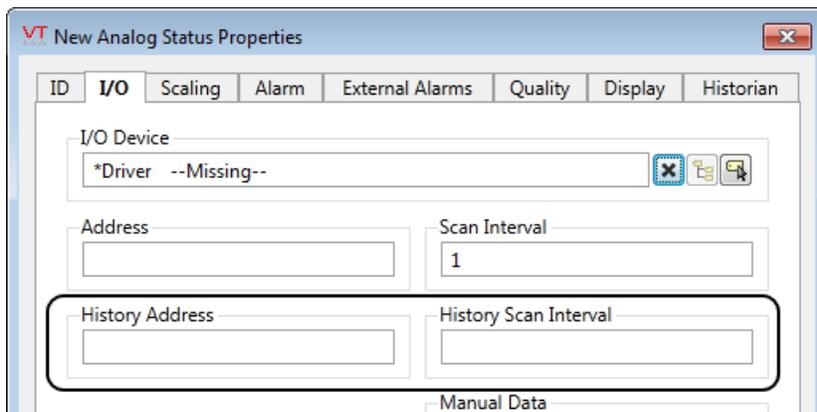
Tag Value:

If an address has been defined (standard address – not history address), the value coming from this address will always be used to set the tag value, regardless of whether or not a history address is also defined.
If only a history address has been defined, and if the value read from the data logger has a newer timestamp than the tag's current timestamp, then the value read from history will be used as the tag's value.

Varying History Scan Intervals

Should you attempt to set different history scan intervals for the different fields within one logger, VTScada will force all the tags associated with that logger to read at the same time. This time will be the smallest of the varying history scan intervals.

Scaling

A useful technique for maximizing the data being stored by the RTU is to not scale it at the logger level, leaving it as a two byte integer. Scaling will be done in VTScada after reading from the logger, using the tag's scaling properties.

Using Digital Status tags

Digital Status tags are configured the same way as Analog Status tags with the exception of the format used in the history address field. For Digital Status tags this takes the form: DLx:y/b where the /b indicates the bit number within the address. Thus, bit 0 of field 3 of logger 1 would be addressed as DL1:3/0.
By using individual bits, each field in the logger can store as many as 32 digital values, thus maximizing storage in the RTU.
You must check the Read History box in the I/O tab in order to see where to put the address.

Note that the appearance of the Digital Status tag's I/O tab will change when the Read History box is checked as shown:
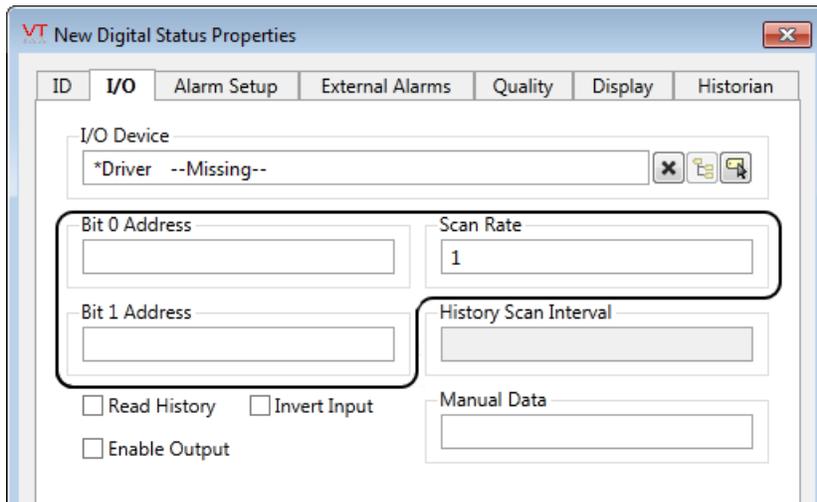
History Data for a TBox RTU

TBox RTUs, manufactured by CSE Semaphore, have the ability to store periodic time-stamped process data internally using Sample Tables. Because the standard Modbus does not have facilities for transferring time-stamped data, the TBox implements a set of custom Modbus functions to accomplish the transfer. To enable these tables to be read into VTScada tags, enhancements have been made to the standard VTScada Modbus driver specifically to access the TBox sample tables.
3 forms of history data are available: Sample Data, Digital Chronology and Analog Chronology.

- Sample data is read with a History Address of "STx", where x is the table number to read, starting at 1.
- Digital Chronology is read with a History Address of "DCxxxx", where xxxx is the address of the digital.
- Analog Chronology is read with a History Address of "ACxxxx", where xxxx is the address of the analog.

The driver will read all the available history data in a compressed mode, and will then poll at the History Scan interval for new data, reading any that is available.

The total number of simultaneous sample-table or chronology reads on a TBox RTU is ten. Because Ethernet-connected RTUs can be accessed by multiple masters, VTScada reads only one sample table or chronology at a time even if multiple tables are read at the same rate. Using this scheme, if multiple tables need to be read at the same time they are automatically read in sequence.

Up to 64 simultaneous reads are possible, but a practical limit should be lower than this, particularly for RTUs that can be accessed by multiple masters. You are advised to keep the total number of reads to fewer than ten.

Koyo Address Translation

VTScada supports the Koyo address format by internally translating to standard Modbus addresses. You do not need to perform any translation between the two formats.

A Koyo address will consist of an address key (one or two letters) followed by an octal value. For example, a Koyo address might look like "SP612".

Koyo Address Keys
X
SP
Y
C
T
CT
S
V
GY
GX

Attempting to provided an invalid octal value in the Koyo format will result in an error condition. Further information about the relation between Koyo's format and standard Modbus addresses can be found in the Koyo documentation.
Standard data type suffixes are supported

Modbus Plus PLC Addressing

When communicating using Modbus Plus, request packets may have to be routed across various BP85 Bridge Plus devices in order to reach the intended PLC. The routing information is stored as part of the PLC address.
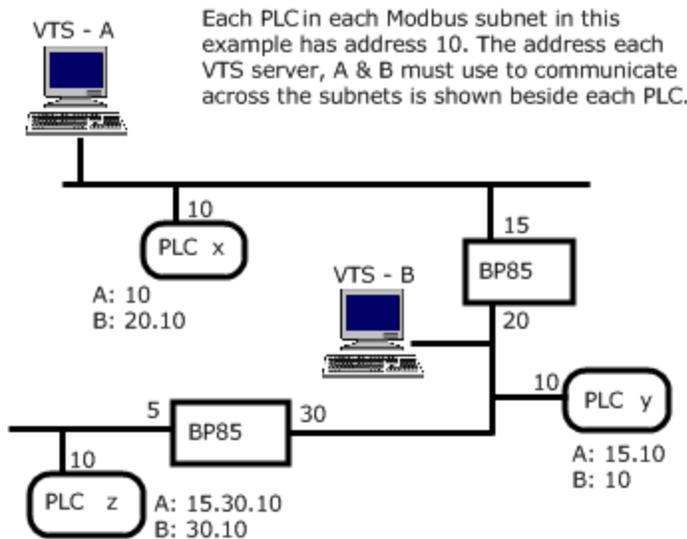
Since VTScada workstations that are located on different subnets will have to be routed through intervening BP85 Bridge Plus devices, it is therefore necessary that the PLC address of the Modbus Compatible Device tag to be resolved based on the workstation the tag is running on. The expression to use for the PLC address of PLC (x) in the above example might be as follows:

```
ToUpper(\Code\RPCManager\WkStnName) == "SERVER_B" ? "20.10" : "10"
```

This code says that, if the address is being resolved from server B, then use "20.10" otherwise use "10". The following illustration shows how this would look in your tag configuration panel.

Modbus Compatible Device Driver Tags

Not counted towards your tag license limit.
The Modbus Compatible Device driver provides an interface to a Modicon PLC.

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Any of the following communication channels may be used:

- Serial RTU
- Serial ASCII
- SA-85 I/O card for ModBus Plus communications
- Embedded TCP/IP
- Open Modbus TCP

This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the Options tab to decide whether this feature should be used in your application. If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In

this case, only the Multiplexer should be configured to re-write auto-matically.

The number of bytes in a Modbus message is limited by the setting: ModiconVTSMaxBlock. By default, this is 125.

When this driver is used in combination with a TCP/IP tag, the standard TCP port is 502.

By default, addressing is zero-based, as set by the application property ModiconTypedAddr1Offset. If you change this property, it will be changed for all Modbus-compatible drivers in the application. If you need some drivers to be zero-based and others to be one based, you can achieve this by creating a parent Context tag for each driver, adding a property named ModiconTypedAddr1Offset to that Context tag. The child driver tag will take the value in the parent Context tag instead of the global property value.

Modbus Compatible Device Properties

Properties Folder Tab	Property	Value
Options Tab	Comm Channel	Serial RTU
Serial Tab	PLC Address	1
Serial Tab	Time-Out Limit	0.5 (Make it shorter if Scan

		Interval is going to be very short)
Serial Tab	Port	COM2 (name of Serial Port tag)
Serial Tab	Retry Delay	0
Serial Tab	Hold	deselected
Serial Tab	Retries	2
Serial Tab	RTS Key Off Delay	0

Related Information:

...Modbus Compatible Device I/O Addressing

...Modbus Plus PLC Addressing

...SCADAPack History Read

...History Data for a TBox RTU

...Koyo Address Translation

...See also: ModiconVTSMaxBlock in the VTScada Admin Guide

Modbus Compatible Device: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Modbus Compatible Device: Options Tab

The Options tab of the Modbus Compatible Device tag properties folder can be used to set byte swapping for long integer, IEEE single or IEEE double bytes. If this driver is communicating over a modem, then the Options tab is also used to indicate that squelch noise should be suppressed, if present.

Comm Channel

The Comm Channel section enables you to select a radio button indicating the communication channel and protocol to use for this Modbus Compatible Device driver. The default communications protocol is Serial RTU. This will be one of:

- Serial RTU Select to use the RTU mode of the Modbus serial communications standard.

- Serial ASCII Select to use the ASCII mode of the Modbus serial communications standard.
- SA-85 (MB+) Select to use the Modbus Plus communications protocol via a Modicon SA-85 card.
- Embedded TCP/IP Select to use the Embedded TCP/IP protocol.
- Open Modbus TCP Select to use the Modicon Open Modbus/TCP specification to receive Modbus protocol over Ethernet TCP/IP.

Squelch Noise Present

Applies when communicating over a modem. Check this box to enable the squelching of background noise. By default, the Squelch Noise Present check box is not selected.

Reverse Long Integer Byte

Select whether or not byte swapping should occur for long integer values. By default, the Reverse Long Integer Byte check box is not selected.

Reverse IEEE Single Byte

Select whether or not byte swapping should occur for IEEE single byte values. By default, the Reverse IEEE Single Byte check box is not selected.

Reverse IEEE Double Byte

Select whether or not byte swapping should occur for IEEE double byte values. By default, the Reverse IEEE Double Byte check box is not selected.

Force Multi Element Write OpCodes

Certain hardware requires data to be written using multi-element OpCodes (15 and 16) even though the actual data is

single-element (OpCodes 5 & 6). By checking this box, you can ensure that this driver instance is able to communicate with devices that require the multi-element codes.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one. If the last output values are stored, they may be re-written by either of two methods:
- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated. This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.

Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Maximum Coalesce Gap

Use only if the Modbus compatible device does not support reads from addresses that are not explicitly declared in the device. VTScada will attempt to coalesce reads from a range of addresses, which may lead to an error with these devices.
If left blank, there is no effect on driver coalescing.
Set to zero to allow for devices that cannot read from addresses that are not explicitly declared.
Set to a number greater than zero to fine-tune read coalescing. The gap is measured in units of two-byte words. Warning: when using low maximum coalesce gap values, it is important to avoid unnecessary gaps in VTScada tag addresses.

Note: if this value is set, driver response time may be slowed due to the increased number of read blocks.

See also: VTSMaxBlock in the VTScada Programmer's Guide

Modbus Compatible Device: Serial Tab

The Serial tab for the Modbus Compatible Device Driver tag properties folder consists of attributes used to identify and establish a connection to the hardware.

Port

Enables you to select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag. A Serial Port tag opens a serial port to enable communications between VTScada and your PLC or RTU. A TCP/IP tag enables you to connect to a series of hosts, allowing you to transmit data across a network or over the Internet.
The Port field can be cleared using the X button that appears to its right.

IP Network Listener Group

Configure only if this driver is to accept in-bound TCP or UDP communications. Select an existing IP Network Listener tag using the drop-down list.
If the IP Network Listener is configured, this driver must use Open Modbus TCP, configured on the Options tab.

IP Allow

Optional. Misconfiguration of this field is a common source of problems.

Used in connection with an IP Network Listener tag. General IP address filtering should be set on the IP Network Listener, as this will be more efficient. The list of allowed IP addresses in the driver is intended to prevent misconfigured devices from interfering with other devices.

Both filters (that in the Network Listener and this one) can be configured at the same time. The filter in the IP Network Listener will be applied when the device first connects and the local filter will be applied after a specific driver instance has been identified.

PLC Address

The station address of the Modicon programmable logic controller. By default, the PLC Address field is set to 1.

An expression may be used to provide the address if needed. This may be required on a network where the address varies depending on the workstation where the tag is running.

Note that, to create a valid expression for an address that contains non-numeric characters, you must enclose the address in quotation marks. e.g. "31.0.0.0" See: Modbus Plus PLC Addressing.

Time-Out Limit

Set the receiver time-out limit in seconds or fractions of a second. This is the length of time that this driver should wait for a reply from the PLC or RTU.

Port

Enables you to select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag. A Serial Port tag opens a serial port to enable communications between VTScada and your PLC or RTU. A TCP/IP tag enables you to connect to a series of hosts, allowing you to transmit data across a network or over the Internet.
The Port field can be cleared using the X button that appears to its right.

Squelch Time

Sets the amount of time (in seconds or in fractions of a second) to wait before clearing noise. Enabled only when Squelch Noise Present has been selected on the Options tab.

Retries

The Retries spin box enables you to select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Hold

The Hold check box selects whether this driver should hold data from the PLC or RTU in the event of a communications failure.
If the Hold check box is selected, the last received value from the PLC or RTU will be held.
If the Hold check box is not selected, the data will be invalidated in the event of a communications error.
By default, the Hold check box is not selected.

Retry Delay

The Retry Delay field enables you to specify the amount of time (in seconds or fractions of a second) that the driver will wait between attempts to repeat a transmission of data if the previous attempt(s) have failed.

By default, the Retry Delay field is set to 0 (indicating that there should be no delay between retry attempts).

RTS Key Off Delay

The RTS Key Off Delay field represents the amount of time (in seconds) that this Modbus Compatible Device will wait before dropping RTS at the end of a data transmission.

Modbus Compatible Device Properties: Modbus Plus Tab

The Modbus Plus tab of the Modbus Compatible Device tag properties folder is used when an SA-85 I/O card for Modbus Plus communications has been configured.

SA-85 Adapter Number

The SA-85 Adapter Number section enables you to specify the number of the adapter if you have installed an SA-85 network adapter card for Modbus Plus communications networks. You can specify adapter 0, adapter 1, or none by selecting the associated radio button. The default setting for the SA-85 Adapter Number section is None.

SA-85 Polling Rate

Set the rate in seconds or fractions of a second at which you wish the SA-85 network adapter card to be polled. The default SA-85 Polling Rate setting is 0.025.

Retries

Select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Time Out (x500mS)

Specify the message response time limit that the driver will wait for a reply from the PLC. This time limit may be set from 1 to 20 increments. The default setting for the spin box is 2 (i.e. two 500 millisecond increments, or 1 second).

Maximum Paths Open

Set the maximum number of Modbus Plus sessions that can be opened simultaneously by the installed SA-85 network adapter. The Maximum Paths Open spin box may be set from 1 to 8. The default setting for the Maximum Paths Open spin box is 8.

Modbus Compatible Device: Virtual I/O Tab

These properties are used to configure a virtual PLC in the memory of

your PC.

Holding Coils 0xxxx (0-999)

Enter the number of 00000-series holding coils to be assigned to the virtual PLC.

Input Coils 1xxxx (0-9999)

Enter the number of 10000-series input coils to be assigned to the virtual PLC.

Input Registers 3xxxx (0-9999)

Enter the number of 30000-series input registers to be assigned to the virtual PLC.

Holding Registers 4xxxx (0-9999)

Enter the number of 40000-series holding registers to be assigned to the virtual PLC.

Listen on Channel x

Select the channels of the Modbus Plus network to which the virtual PLC should listen. Please note that these do not apply to serial or TCP/IP connections.

Modbus Compatible Device Widgets

The following widgets are available to display information about your application's Modbus Compatible Device Driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

Motorola ACE Driver Tags

Note: Motorola ACE features will be available only if your VTScada license includes this option.

Not counted towards your tag license limit.

The Motorola ACE driver should be used for all Motorola ACE / Moscad RTUs in your system.

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Requires a Motorola IP Gateway tag, and the Motorola table configuration file, MDLC_TYP.CFG, which lists the possible RTU table configurations.

> Note: This driver relies on the parent IP Gateway device to report whether a site is up or not. That device must be property configured *via the Motorola STS software*. Please refer to the STS Advanced Features PDF, which should be installed with STS.

The configuration file describes only the configuration for communication. It does not configure the RTU. This file should be modified to match the configuration of the RTU.

Related Information:
Motorola ACE Driver I/O Addressing
 Motorola IP Gateway Tags

Motorola IP Gateway: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.
There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Motorola ACE Driver Properties: Settings Tab

The settings tab is used to define the communication parameters to your Motorola ACE RTU. The IP Gateway device will handle retries and timeouts. No field exists in this driver to allow a retry specification. Time limit values are provided only as a fall-back in the case that the IP Gateway does not handle the time limit.

Motorola IP Gateway

The name of a configured Motorola IP Gateway tag. The first such parent found will be used automatically.

RTU ID

The identifier of the MDLC network that the RTU device was configured with.

RTU Type

Should match the RTU type, as specified in the API configuration file. This name will be found immediately after the keyword, "Type," and just before the table definitions.

RTU Status Interval

The period between RTU status polls.

Burst Refresh Interval

The burst refresh poll interval. Used to cause a read poll to occur at a regular interval while the driver is using burst mode and polling would otherwise be disabled.

Read Time Limit

The IP Gateway device configuration provides the time-outs to be used. The value in this field is only to provide a fail-safe in the event that the usual time-out limit fails to work for read operations.

Write Time Limit

The IP Gateway device configuration provides the time-outs to be used. The value in this field is only to provide a fail-safe in the event that the usual time-out limit fails to work for write operations

Hold

Controls whether this driver should hold data from the RTU in the event of a communications failure.

If the Hold check box is selected, the last received value from the RTU will be held.

If the Hold check box is not selected, the data will be invalidated in the event of a communications error.

By default, the Hold check box is not selected

Motorola ACE Tag Widgets

The following widgets are available to display information about your application's Motorola ACE driver tags:

…Alarm Priority Box

…Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

Motorola ACE Driver I/O Addressing

Note: Motorola ACE features will be available only if your VTScada license includes this option.

Addressing uses the following form, where T, R and C are zero-based integers, and B is the letter B.
 [B]T:R:C
Where...

- The letter B is optional, as indicated by the square brackets. (Do not type the brackets in an address.) If present, this indicates that the driver should wait for unsolicited data to be pushed out from the RTU, rather than the driver polling the RTU.
- T is the table number to use in the MDLC_TYP.CFG file.
- R is the row number to use in the MDLC_TYP.CFG file.
- C is the column number to use in the MDLC_TYP.CFG file.

Examples:
0:1:2

Designates Table 0, Row 1, Column 2 in polling mode, since the B is not present.

B3:2:1

Designates Table 3, Row 2, Column1 in Burst mode.

Burst mode may only be used if the RTU was programmed to communicate this way. Simply adding the B to the address will not cause the RTU to push updates.

Burst mode applies only to data reads. After an initial poll the driver will wait for the RTU to report a change. The burst option is ignored for data writes.   A burst refresh interval is provided in order to periodically perform an explicit poll of the RTU.

Motorola – Sample MDLC_Type.CFG file

Note: Motorola ACE features will be available only if your VTScada license includes this option.

The following is an example of a Motorola MDLC_Type.CFG file. It cannot be used without modifications unique to the configuration of your RTU. When modifying the configuration file, you need only setup the tables that are in use for your RTU. The number following the name of the configuration is the number of tables defined below it. Data types available for use are as follows (note that these are named differently than in the device programming software): Bit, Val, Long and Float.

```
; File Type & Version
;--------------------------
FILE_TYPE   RTU_TYPE
VERSION 1.0

; The number of RTU types defined within this file:

Number_of_Types 2

; Definition of RTU Types
;--------------------------
; Type name Number of tables

Type ExampleConfig2 9

; Table_no N_cols Col_0 Col_1 Col_2 Col_3 Col_4 Col_5 Col_6 Col_7
```

```
Table  0   1   Val
Table  1   3   Bit     Long   Long
Table  2   1   Float
Table  3   3   Bit     Long   Long
Table  4   1   Val
Table  5   2   Bit     val
Table  6   1   Val
Table  7   6   Bit     Bit    Val     Float   Bit    Bit
Table  8   6   Float   Float  Float   Val     Val    Bit

Type ExampleConfig2 2

; Table_no N_cols Col_0 Col_1 Col_2 Col_3 Col_4 Col_5 Col_6 Col_7

Table  1   3   Bit     Long   Long
Table  3   1   Val
```

Omron FINS Driver Tags

Not counted towards your tag license limit.

"FINS" stands for Factory Interface Network Service, which is a protocol for sending messages between controllers on Omron FA networks. The VTScada FINS driver takes care of selecting which of the available protocols to use and which PLC unit, node and network to communicate with.

As described in the FINS command reference, the driver can work with three different types of network: Host Link, SYSMAC NET (TCP/IP) and SYSMAC LINK.

There are differences in communications depending on the type of network in which the FINS commands are used. These differences are listed in the following table:

Item	SYSMAC NET	SYSMAC LINK	Host Link
Code [a]	Binary or ASCII	Binary Only	ASCII Only
Maximum Data Length [b]	Binary: 1990 bytes	542 bytes	542 bytes
Baud Rate	2 Mbps	2 Mbps	1200, 2400, 4800, 9600 or 19200 bps
Maximum # of Nodes	126 [c]	62 Nodes	RS-232: 1 Node RS-422: 32 Nodes

Destination Nodes	PCs, IBM PC / AT Compatible or CPU Bus Units	PCs, IBM PC / AT Compatible or CPU Bus Units	PCs Only

Notes:

a. All nodes within any one network must be set to the same code. Networks connected via gateways may use different codes. The code is automatically converted as necessary in inter-network communications.

b. This is the maximum length from command code through text for internet communications. Slightly longer communications are possible in local connections. Refer to the operation manuals for individual systems for details.

c. Not including a Line Server.

The PLC system may be configured on multiple networks, connected via a gateway (either a computer with two network link units or support boards, or a unit specially designed to work as a gateway). Using FINS commands, a given network node can communicate with other nodes on three network levels:

a. Nodes on the local network.

b. Nodes on networks connected directly via one gateway.

c. Nodes on networks separated by one other network (i.e. networks connected via two gateways). Note that a connection between a network and a host computer via a Host Link is not counted as an additional gateway for this purpose.

Related Information:

Omron FINS I/O Addressing

Related Application Properties

The following property settings hold additional configuration parameters for your Omron FINS driver:

OmronFINSServerFailTimeOut – Omron FINS server fail timeout. Defaults to 30 seconds.

OmronFINSDriverSemTimeOut – Omron FINS driver semaphore timeout. Defaults to 10 seconds.

OmronFINSSharedRPC – TRUE if all Omron FINS drivers share the same RPC service. Defaults to zero whereby the RPC service will be the Name of the driver tag. If true the RPC service will be "OmronFINSServer".

Omron FINS Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags

(note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added.

To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Omron FINS Driver Properties: Protocol Tab
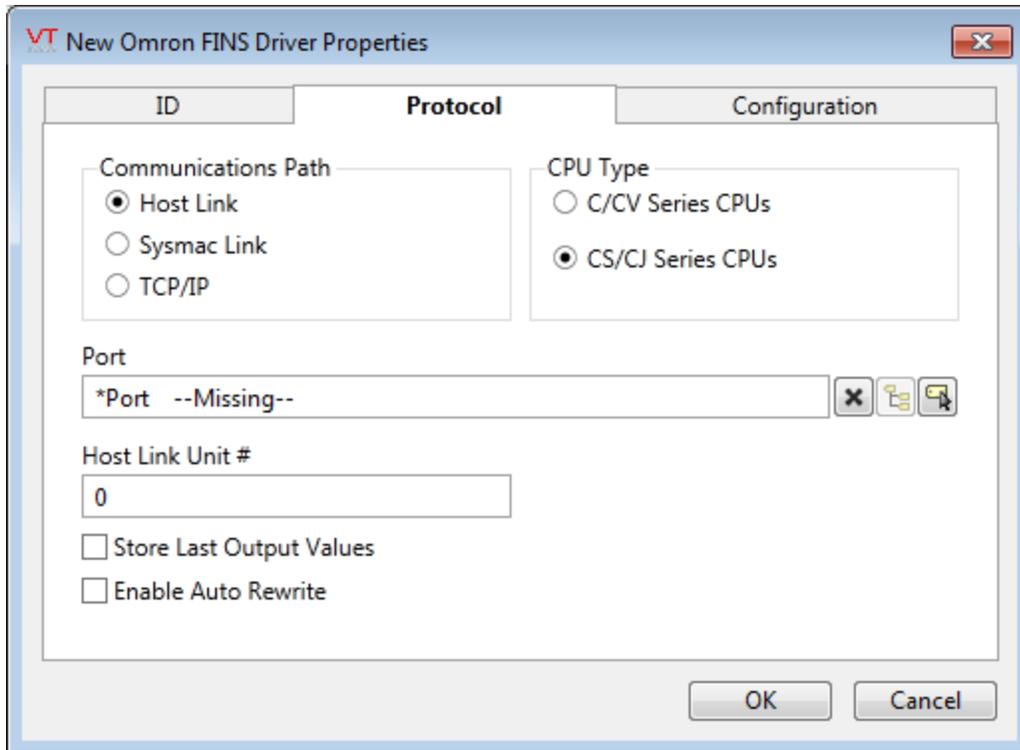
Connection Type

Choose which of the three available network types is being used by the system: Host Link, SYSMAC Link, or TCP/IP (i.e. SYSMAC Net Link).

CPU Type

The selection of CV Series CPU vs. CS/CJ Series CPU enables the driver to adjust for differences between these two types of Omron units.

Communications Port

Available only for Host Link and TCP/IP net-work types. In most cases, a COM port is used by Host Link while a TCP/IP port is required by the TCP/IP network type.

Host Link Unit

Available only when a Host Link network is configured.

Store Last Output Value

If selected, the last output values sent to the Omron FINS device will be saved or stored. Use this option if you want to refresh the output values in the event of a communications failure or a power interruption.

Enable Auto Rewrite

If checked this box causes the last stored output values to be rewritten to the device in the event of a communications loss or power failure.

Omron FINS Driver Properties: Configuration Tab

Base

Available only when the SYSMAC Link network protocol has

been selected on the Protocol tab. Specifies the base address of the network support board in the workstation.

Destination Network

In a multi-network environment, use this option to specify by number, the network with which to communication. Set 0 for HostLink. The value is configured in the PLC for SYSMAC Link or SYSMAC NET, using the CX-Integrator.

Destination Node

Specify by number, which node on the network contains the PLCs with which you wish to communicate.

Destination Unit

Specify, by number, the PLC unit on the selected node with which you wish to communicate. The CPU is unit 0 while other rack-mounted cards are the unit number + 10.

Source Network

If you have a multi-network environment, specify by number, the network with which the workstation is connected. Set as for PLC network.

Source Unit

Specify by number, which node on the network contains the workstation.

Time Limit

Responses must be received within the specified time limit, otherwise a communications error will be assumed to have

occurred.

Retries

Set the maximum number of times to attempt to send a message to the PLC

Hold

Select whether this driver should hold data from the PLC or RTU in the event of a communications failure. If the Hold check box is selected, the last received value from the PLC or RTU will be held. If the Hold check box is not selected (default), the data will be invalidated in the event of a communications error.

PLC Time Sync

When selected, the workstation time is written to the PLC every hour.

Omron FINS Driver Tag Widgets

The following widgets are available to display information about your Omron FINS driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

Omron FINS I/O Addressing

The addressing system for I/O tags consists of a memory area prefix followed by an address number. In the following table, the "#" sign should be replaced by the address number.

Address	Memory Area
IR#	Input / Output Area
WR#	Work Area
HR#	Holding Bit Area
AR#	Auxiliary Bit Area
TI#	Timer Completion Flag Area
CO#	Counter Completion Flag Area
PT#	Timer PV Area
PC#	Counter PV Area
DM#	Data Memory Area
EM#	Current Extended Memory Area

Related Information:
Omron FINS Driver Tags

Omron Host Link Driver Tags

Not counted towards your tag license limit.
The Omron Host Link driver provides an interface to an Omron PLC via a serial port.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report,

both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Note: The Omron driver requires that the PLC be in Monitor mode in order for writes to work.

This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the Serial tab to decide whether this feature should be used in your application. If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Related Information:
Omron I/O Addressing

Omron Host Link Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area
The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Omron Host Link Driver Properties: Serial Tab

The Serial tab for the Omron Host Link driver tag properties folder consists of attributes used to identify and establish a connection the Serial Port tag that the Omron Host Link driver should use to transmit data to the I/O device. The Serial tab is also used to select the communication protocol to be used in transmissions of data between the I/O device and the driver.
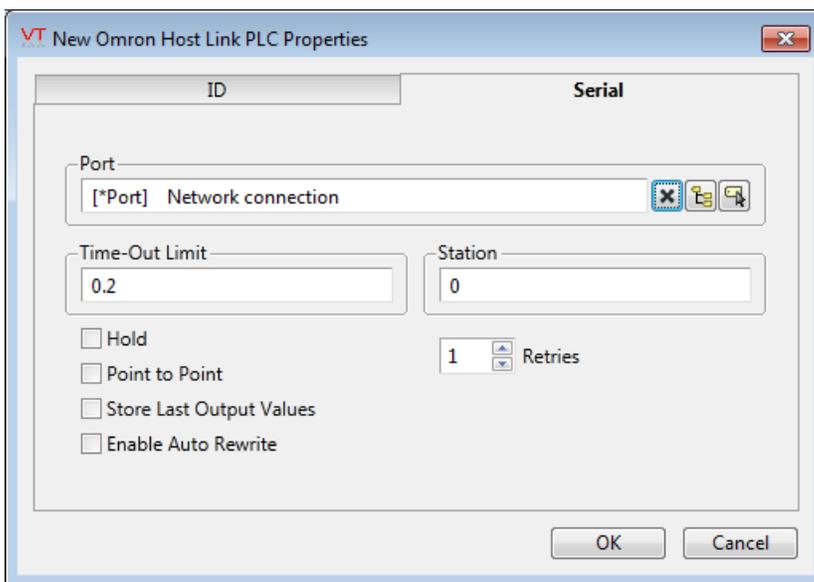
Port

Enables you to select the Serial Port tag or TCP/IP tag you wish to be associated with this driver tag. A Serial Port tag opens a serial port to enable communications between VTScada and your PLC or RTU. A TCP/IP tag enables you to connect to a series of hosts, allowing you to transmit data across a network or over the Internet.

The Port field can be cleared using the X button that appears to its right.

Time-Out Limit

Sets the receiver time-out limit (i.e. the time in seconds or fractions of a second that this driver should wait for a reply from the PLC or RTU).

Hold

Controls whether this driver should hold data from the PLC or RTU in the event of a communications failure.

If the Hold check box is selected, the last received value from the PLC or RTU will be held.

If the Hold check box is not selected, the data will be invalidated in the event of a communications error.

By default, the Hold check box is not selected.

Point To Point

Select whether the Point To Point protocol to be used by the Omron Host Link driver in communications with the PLC.

If selected, the driver will use point-to-point protocol (PPP) to communicate with the PLC.

If the Point To Point check box is not selected, the driver will use multi-drop protocol.

By default, the Point To Point check box is not selected.

Station

Enter the PLC station address in this field.

Retries

Select the number of attempts that will be made by this driver if there is no reply to a message. An error will be declared after this number has been reached.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one.

If the last output values are stored, they may be re-written by either of two methods:

- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

> If checked, the Store Last Output Values option will also be activated.
> This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.
> Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Omron Host Link Driver Tag Widgets

The following widgets are available to display information about your application's Omron Host Link driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

Omron I/O Addressing

The addressing for Omron C-Series PLCs in VTScada is made up of two parts, the memory area and the address. The address string is made up of a two-letter memory designator and an unsigned integer value as outlined in the following table. Although the Host Link protocol does not support bit reads or writes, VTScada does support bit addressing when writing or reading the word registers. Caution should be exercised when writing bits because all bits in the affected register will be written.
The addressing for Omron Host Link driver bits supports the period (.) separator, as well as the backslash character (/), enabling the use of the standard addressing format utilized by Omron.
The bit address is an optional part of the address string. A complete address for an Omron C-Series PLC would be in the following format:

`MMdddd[/BN]`

Where MM is taken from the first column in the following table, dddd is the register number, and BN is the optional bit number.
The following table shows the addressing values for Omron C-Series PLCs.

Prefix	Description	Channel Digits	Channel	Bit
AR	Auxiliary Relay	2	Yes	Yes
TC	Timer/Counter	3	Present Value	Status (don't add digits)
T	Timer	3	Present Value	Status (don't add digits)
C	Counter	3	Present Value	Status (don't add digits)
DM	Data Memory	4	Yes	No
HR	Holding Relay	2	Yes	Yes
IR	Internal Relay	3	Yes	Yes
LR	Link Relay	2	Yes	Yes

SR	Special Relay	3	Yes	Yes
TR	Temporary Relay	0	No	Yes
EM	Extended Memory	4	Yes	No

Examples of Omron Addressing:

```
DM120
HR98/2
AR120/15
```

The data types of the Omron registers are set by appending one of the following strings to the end of the address string:

```
"/UWORD"
"/UDWORD"
"/SWORD"
"/SDWORD"
"/BCD4"
```

The default data type is unsigned word integer.

Note: For a list of all available data type suffixes that can be used for tag I/O addresses, please refer to Data Type Suffixes for Tag I/O Addresses.

OPC Client Driver Tags

Not counted towards your tag license limit.

By adding OPC Client Driver tags to your application, you are able to communicate with OPC servers on your network, whether those are other VTScada applications, control equipment or other programs.

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

The OPC interface allows for two-way communication, thus you can add both input and output tag types that exchange information with the OPC server through the OPC client drivers you configure.

The ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

OPC Client Driver Properties: OPC Tab

The OPC tab for the OPC Client driver tag properties folder consists of attributes used to identify and establish a connection to the control device to be associated with this driver. The OPC tab for an OPC Client driver is also used to identify the remote server node that the driver should use to transmit data.
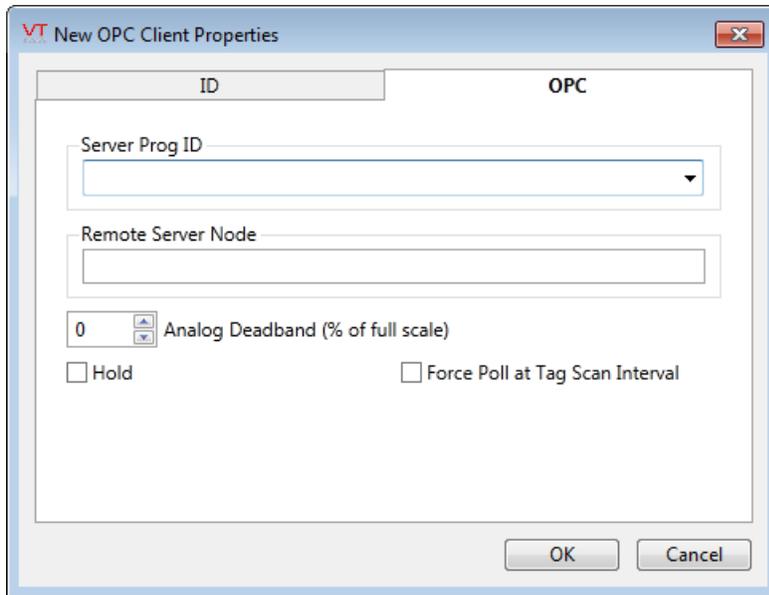
Server Prog ID

The Server Prog ID field enables you to specify the program
identifier (ProgID) for the OPC server to which you wish to con-
nect. This will be a string of the form Man-
ufacturer.OPC.ServerType.
Examples of a Server Prog ID are: Trihedral.VTSOPC.DA or Mat-
rikon.OPC.Modbus.

Note: Note: The OPC Server vendor generally supplies the Server Prog
ID; please refer to your OPC Server manufacturer's documentation for
the correct details.

Remote Server Node

Use this field to specify the name of the workstation that is act-
ing as the OPC server. The Remote Server Node field can be
left blank in the event that the local workstation is the OPC
server.

Analog Deadband (% of full scale)

Set a percentage by which analog values must change before a change in value is reported. If the percentage is 0, the updates will be event driven, occurring on each value change.

Hold

In the event of an interruption in communication, this option will cause the last good value to be held. You should use this option with caution since, unless there is another means to alert you to interruptions in communication, it will appear that a valid value is still being received.

Force Poll at Tag Scan Interval

In the default configuration, data updates are not requested from the OPC Server. Instead, it will update data on a change of value or quality. If this option is selected, then data is requested from the OPC Server at the rate of attached tag's scan interval. This may cause excessive load on the OPC Server host to service all the data update requests.

OPC Client Driver Tag Widgets

The following widgets are available to display information about your application's OPC Client Driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

OPC Client Driver I/O Addressing

The VTScada OPC Address Select tool, found in every I/O tag that uses an OPC Client driver, will help you build addresses for your I/O tags.
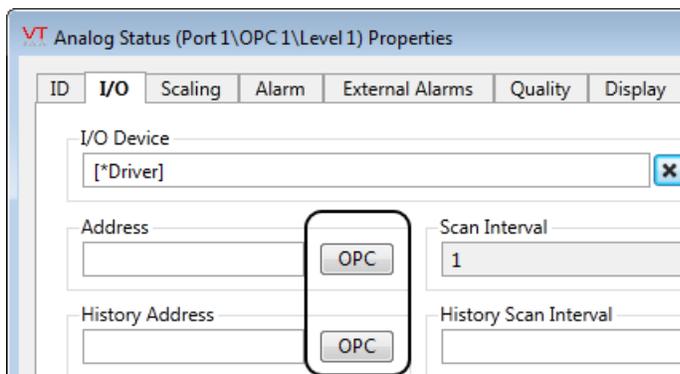
Clicking the OPC button opens the OPC Address Select tool. (Example shows a client connected to the Trihedral OPC Server – your display will vary depending on the server that your OPC Client is using.)
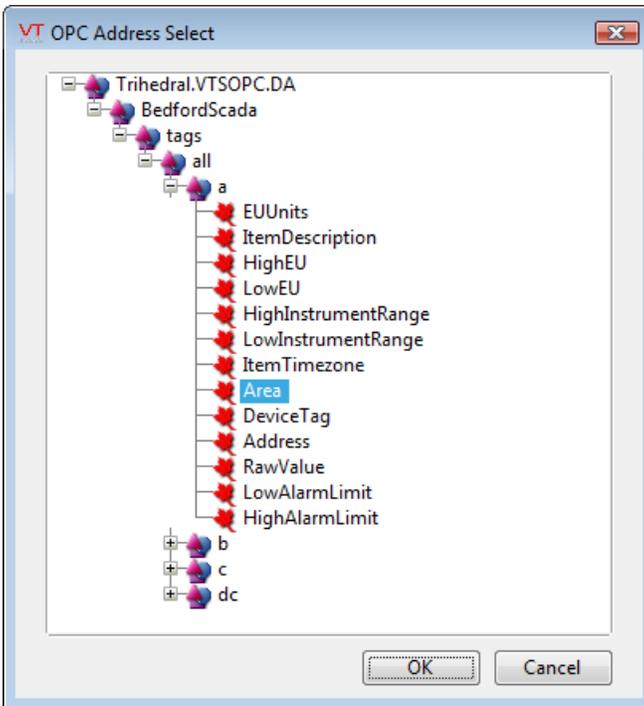
Using the Address Select tool, you can quickly navigate through the nodes to the OPC item that you wish to use.

> Note: For the VTScada OPC Server, there is no leaf for the "Value" of a tag. The tag node itself has the value attached.

The following is a basic example (not using the OPC Address Select tool) of using an OPC Client driver tag to read data from an Omron C200H PLC via an OPC Server (in this example, an INGEAR Omron OPC Server version 2.25.0.1).

> Note: There is no standard for OPC driver addressing. The following example covers only one possible configuration. For your application, the definitive guide to OPC addressing will always be the documentation for the OPC Server in use.

On the OPC Server (using the user interface provided by the manufacturer of the OPC Server):

1. Create a device named Omron.
2. Create a group named MyGroup.
3. Create two items, Tag1 and Tag2.

4. Configure the tags with the correct addressing information for connecting to the Omron PLC.

VTScada OPC Client Driver Tag Properties:

OPC Client Driver Tag Properties Folder Tab	Property	Value
OPC tab	Server Prog ID	CimQuest.IGOMOPC.1

VTScada Input Tag Properties (Analog Input or Digital Input)

Enter the address Omron.Group.Tag1 in the Address field for the VTScada input tag that will read data from the Tag1 item on the OPC Server.

Enter the address Omron.Group.Tag2 in the Address field for the VTScada input tag that will read data from the Tag2 item on the OPC Server.

Note that the addresses entered for the above VTScada input tags are based on the syntax:

```
DeviceName.GroupName.ItemName
```

Note: For a list of all available data type suffixes that can be used for tag I/O addresses, please refer to Data Type Suffixes for Tag I/O Addresses.

OPC Server Setup Tags

Not counted towards your tag license limit.

Note: Use of the OPC Server Setup tag is restricted to VTScada installations that include this feature as part of the license key. You can confirm whether this feature is enabled by opening the About VTScada dialog from the VTScada Application Manager.

Reference Notes:

Server Name: Trihedral.VTSOPC.DA

The VTScada OPC Server supports the OPC DA (Data Access) Standard. The OPC Server Setup tag is used to turn your application into an OPC Server. Minimal VTScada configuration is required on your part to do this.

> Note: OPC security is handled by Windows via user and group security permissions set using the Microsoft Component Services dialog and the COM Security tag of the My Computer dialog. Please refer to Securing an OPC Server for more information on OPC security.

Only one OPC server may be defined per application. Attempting to define a second server in the same application will result in an error dialog stating "An application can have at most one OPC Server Setup tag". Note: Use of the same name space for more than one application is strongly discouraged as it will cause significant problems.

There are no widgets for this tag. When the OPC Server Setup tag has been created, the application's OPC-supporting tags become available on the VTScada OPC server.

Related Information:

OPC Client Driver Tags

OPC Configuration in VTS – Technical information about the OPC Server configuration in VTScada.

OPC Server Setup Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

OPC Server Setup Properties: Settings Tab

An OPC Server has two configurable properties in addition to its ID information:

- The name space for this application on the Trihedral OPC Server. This is a required field.
- The Disallow Writes check box is there as a security feature. If checked, clients can query data from the server, but cannot write to it.

As a convenience, the server's Prog ID will also be shown in this screen. You may find it easiest to reuse the application's name for the server namespace. You should not use the same name space for two applications.

Polling Driver Tags

Not counted towards your tag license limit.
Polling driver tags are designed for use in telemetry applications where communication links may be costly, either in terms of time or money.

Reference Notes:

The Polling driver tag provides the following features:

- The ability to poll multiple stations or I/O devices in a pre-determined order and at a fixed time interval.
- The ability to switch to fast-polling mode of selected sites on demand. See Fast Scan Widget.
- Provides a simple grouping of I/O tags on a single display.
- Navigate to a station page using F11 or F12.

Note: Polling driver tags are not required for Data Flow RTU driver tags since those have a polling feature built-in.

Polling drivers should be configured to stand between a communications driver tag and the I/O tags. Their function is to control the order and the frequency of data transfer between the I/O tags and the remote equipment via the communication driver tag.

All input tags associated with a Polling driver will use the scan interval configured for the Polling driver rather than their configured scan rate.

When data is written from an output tag, it is immediately passed to the Polling driver tag and then on to the associated communication driver tag. Following each write, all I/O addresses will be read immediately so that feedback need not wait for the next polling cycle.

Multiple Polling drivers may be attached to a single communication driver tag, each with a separate poll rate, however the result will be that all will poll at the fastest configured rate.

Station Page Navigation

Pressing either F11 or F12 will open the Station Number dialog:
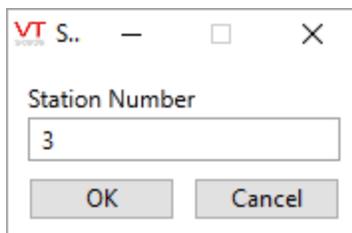
If you have set a station number for each of your polling drivers, and if you have created a page named for each station, where the name ends with the matching number, ("My Station 3"), then operators can enter the station number in this dialog, press OK or Enter, and VTScada will open the matching page. The content of that page is entirely up to you, but should be useful for the relevant station.

Note: If Modbus drivers are used in conjunction with a Polling driver in a networked application, then the application property, Modicon-SharedRPC should be set to 1. This will prevent the operation of the Polling drivers being distributed between the primary and backup servers.

Related Information:

Examples: Scan Interval and Delay After Scan Interval

How to Create a Poll Group

Polling Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Polling Driver Properties: I/O Tab

The I/O tab for the Polling Driver tag consists of properties used to identify the communication driver tag to be associated with this polling driver, and to establish the poll group, the I/O device's sequence in the poll, and the interval at which the equipment associated with the communication driver tag should be polled for data.
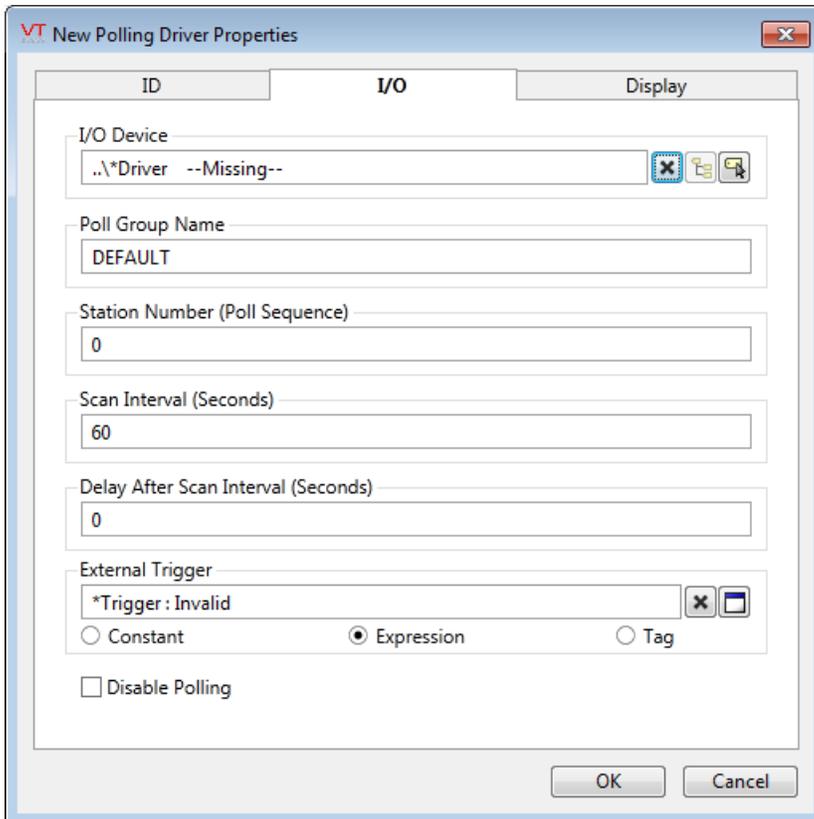
I/O Device

The I/O Device field enables you to specify the communication driver tag for which this polling driver will set the scan rate.

The I/O Device field can be used to associate this tag with a new or existing communication driver tag using the tag browser button. The button opens the Tag Browser, which displays only the existing communication driver tags for your application, and enables you to create a new communication driver tag using its New button.

Poll Group Name

The Poll Group Name field enables you to enter a name by which this poll group will be known (or the name of the existing poll group to which this polling driver should belong).

Note: Multiple polling drivers can associate multiple communication drivers with a single poll group.

For example, you can create 5 polling drivers, each of which is associated with a different communication driver, and enter the same polling group name for each of the polling drivers (e.g. "Poll Group 1," or "East Side

Stations Poll Group"). The result is that the 5 communication drivers associated with the 5 polling drivers will be polled as a single group. Instructions on configuring a poll group can be found in Setting Up a Poll Group.

Poll Sequence Number

The Poll Sequence Number field enables you to specify a number that identifies this polling driver's place in the polling sequence for the identified poll group.

For example, if you have created 5 polling drivers, each of which is associated with a different communication driver (e.g. Modbus1, Modbus2, etc.), and all 5 polling drivers have the same poll group identified (in their Poll Group Name property), you must enter a number from 1 to 5 for each of the polling drivers to determine the order in which the 5 communication drivers belonging to the poll group will be polled.

Scan Interval (Seconds)

The Scan Interval field enables you to enter a number representing the amount of time (in seconds or in fractions of a second) between requests for data from this Polling Driver tag to the I/O device with which it is associated. The Delay After Scan Interval (Seconds) property enables you to specify the amount of time that the polling driver should wait before polling the I/O device (see following).

If Scan Interval is set to 0, the I/O device is scanned for data as fast as is possible.

Note: All data that is set to the same Scan Interval is read in the same poll of the I/O device. For this reason, it is recommended that you limit the number of different scan intervals for a single I/O device to optimize the reading to a minimal number of blocks, and improve overall system update performance. For example, if you have 12 analog inputs connected to one I/O device, it is recommended that the scan intervals for these tags be set to the same value (e.g. 1 (second)).

Delay After Scan Interval (Seconds)

The Delay After Scan Interval (Seconds) property indicates the amount of time that this polling driver should wait before polling the communication driver identified in the I/O Device field.

The system works on a 12-hour clock, starting at midnight. The value entered in the Delay After Scan Interval field therefore indicates the amount

of time after 12:00 am before the system begins to scan according to the time schedule identified for the Scan Interval property.

For example, if you wish the system to begin polling at 1:00 am, and continue to poll every 2 hours (i.e. 1:00 am, 3:00 am, 5:00 am, 7:00 am, etc.), you should set the Delay After Scan Interval property to 3600 seconds (1 hour), and then set the Scan Interval to 7200 seconds (2 hours). Configuration examples are provided in Polling Driver Scan Interval and Delay After Scan Interval Examples.

Note: The Delay After Scan Interval must always be less than the Scan Interval value.

External Poll Trigger

May be linked to any tag or expression that transitions from a 0 or invalid state to a 1-state to trigger a poll. A Trigger Tag should be used here, if you want to schedule polling to occur on a given schedule or in response to an operator's command. Note that an external poll signal will not cause a poll to occur if polling is disabled, or if fast scan mode is enabled.

Disable Polling

Use to disable or re-enable the polling cycle for this poll driver. Note that this does not mean that reads or writes are disabled; only the polling cycle is affected. A Toggle Polling Button widget can be drawn in your application to allow operators to change this value at need.

Polling Driver Properties: Display Tab

The Display tab is used to define the placement (latitude and longitude) of the Polling Driver station. Decimal values should be used rather than degrees, minutes and seconds.
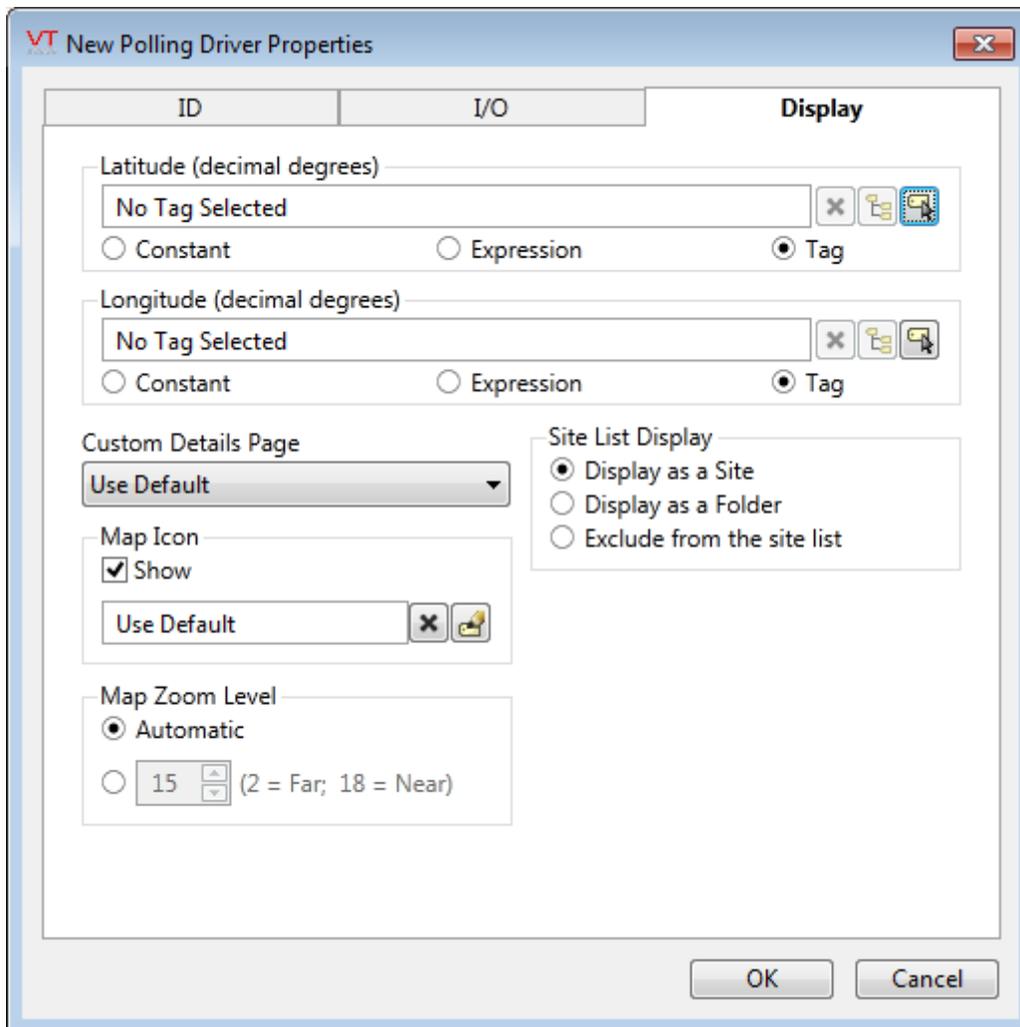
You may find it easier to set the location using the map interface than to enter the latitude and longitude values here. See: Site Map.

Latitude and Longitude

Holds location coordinates for this tag, thereby allowing it to be represented on a Site Map page.

Custom Details Page

If a custom details page has been created for this context tag, then that page should be selected here. If there is no custom details page, then operators will see the standard Site Details page upon clicking the pin for this tag in a Site Map.

Custom Map Icon

If a custom map icon widget has been defined, you can select it here. That icon will then be used instead of the standard pin when this tag is represented on a Site Map page. If there is no custom map icon, then the standard pin will be used.

Site List Display

Assuming that this tag has site properties, it will be included in the list of a Sites page. What will happen when an operator clicks on this tag in that list depends on the Site List display choice, and on whether any of this tag's children are I/O tags.

- Display as Site: A click will open the Site Details page as a pop-up.
- Display as Folder: A click will leave focus on the Sites page, but the list will now show the child tags of this site.
- Exclude: This tag should not be shown in the Sites page list.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).

Default zoom level.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).
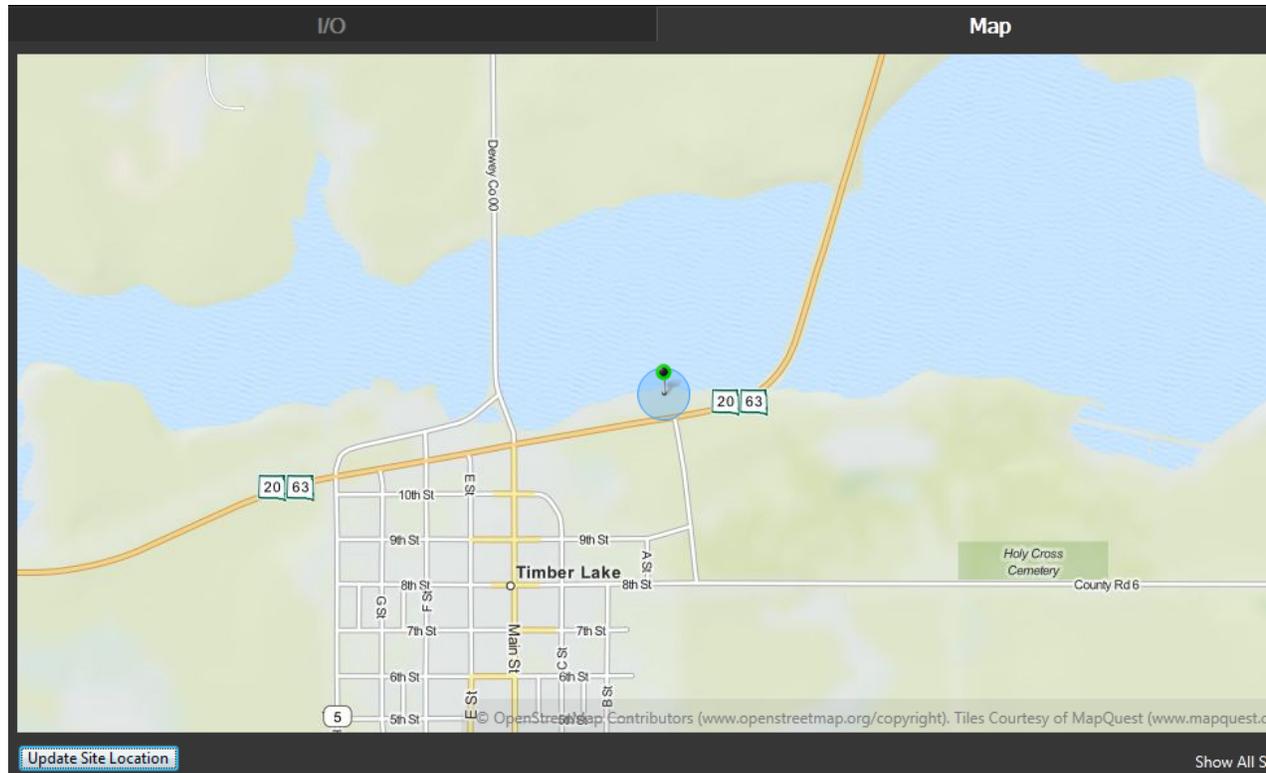
Default zoom level.

Polling Driver Tag Widgets

Polling driver tags enable users to establish groups of PLCs or RTUs that will be polled together. The widgets for a Polling Driver tag enable users to display pertinent data about the RTU or PLC with which they are associated.

The following widgets are available to Polling Driver tags:

Active Indicator Widget

Comm Indicator Widget

Comm Messages Button Widget

Show Statistics Button Widget

Data Age Widget

Draw HDV Widget

Enable Polling Checkbox Widget

Equipment / Status Color Indicator

Fast Scan Widget

Gradient Color Change Widget

Indicator Light Widget

LCD 5x7 Matrix Widget

LCD 7 Segment Widget

Data Age Widget

Multi-Color Widget

Multi-Text Widget

Numeric Value Widget

Plot Data Widget

Polled Station Widget

Site Draw Widget

Site Alarm List

Tag List Widget

Toggle Polling Button Widget

Examples: Scan Interval and Delay After Scan Interval

A Polling Driver tag can be configured to scan a communication driver tag on a regular schedule. The key to doing so is proper configuration of the Scan Interval and Delay After Scan Interval properties.

The Scan Interval property specifies the number of seconds between polls, where the polling cycle starts at midnight. The Delay After Scan Interval property specifies the number of seconds after the Scan Interval to trigger the poll. The Delay After Scan Interval property should not be greater than the Scan Interval property.

The following 3 examples have been provided to help guide you.

Example 1: Poll at 2-hour Intervals Starting at 1:00 am

If you want the poll to happen at 1:00 am, 3:00 am, 5:00 am, 7:00 am (and so forth), the Scan Interval property should be set to 7200 seconds (2 hours), and the Delay After Scan Interval property should be set to 3600 seconds (1 hour after midnight).

Example 2: Poll Every 10 Minutes Starting at Midnight

If you want the poll to happen at 12:10 am, 12:20 am, 12:30 am (and so forth), the Scan Interval property should be set to 600 seconds (10 minutes), and the Delay After Scan Interval property should be set to 0 seconds (midnight).

Example 3: Poll At 3:00 am and 3:00 pm Daily

If you want the poll to happen at 3:00 am and 3:00 pm every day, the Scan Interval property should be set to 43200 seconds (12 hours), and the Delay After Scan Interval property should be set to 10800 seconds (3 hours).

How to Create a Poll Group

A poll group can be configured by associating a polling driver with each communication driver tag that you wish to belong to the group. For example, if you wish to configure 3 communication driver tags to belong to the same poll group, you would proceed as follows:

Create a new Polling Driver tag for each communication driver tag, and configure each as follows...

1. Select the communication driver tag's name in the I/O Device field on the I/O tab of the polling driver.
   There should be one polling driver associated with each communication driver tag.

2. Enter the same text in the Poll Group Name field on the I/O tab of each polling driver (e.g. Eastern Poll Group).

3. Enter an appropriate sequential number in the Poll Sequence Number field for each of the Polling Driver tags. This will identify each Polling Driver tag's place in the poll sequence for the identified poll group. (For example, enter 1 for the first Polling Driver tag in the Eastern Poll Group, 2 for the second Polling Driver tag in the Eastern Poll Group, and 3 for the final Polling Driver tag in the Eastern Poll Group.

Related Information:

Polling Driver Tags

Siemens S7 Driver Tags

Not counted towards your tag license limit.

The Siemens S7 driver provides an interface to Siemens PLCs of the following types:

- S7-200
- S7-300
- S7-400

> Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

Communication with the PLC is via ISO over TCP.

This driver has the ability to save the last value written to each output tag, and to rewrite those values, either automatically when lost communications are restored, or manually by the press of a button. Carefully review the information in the [Options](#) tab to decide whether this feature should be used in your application. If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

When this driver is used in combination with a TCP/IP tag, the standard TCP port is 102.

Related Information:

Siemens S7 Driver I/O Addressing

Siemens S7 Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Siemens S7 Driver Properties: Options Tab

The Options tab for the Siemens S7 driver tag properties folder consists of attributes used to identify and establish a connection to the control device to be associated with this driver.
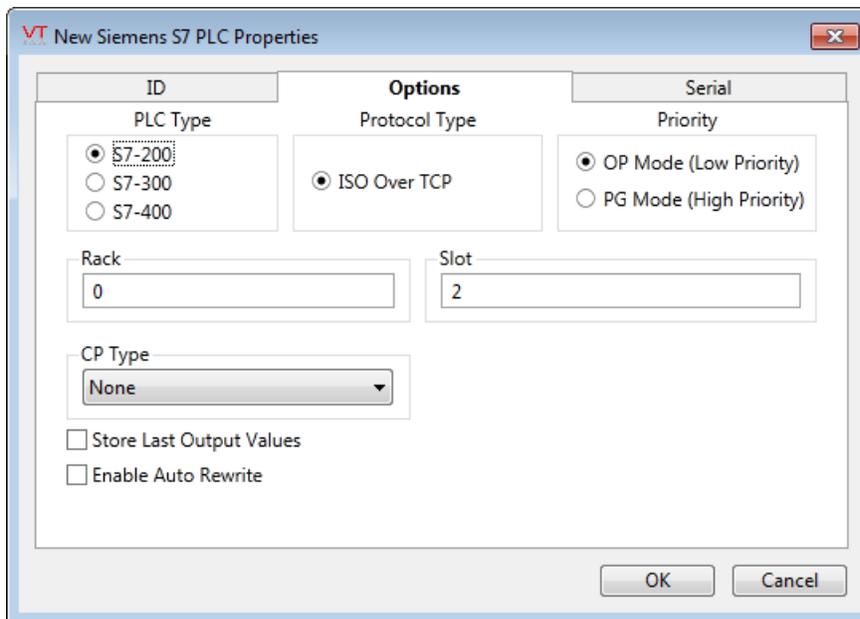
PLC Type
The S7 Driver is able to communicate with 3 Siemens devices: the S7-200, S7-300 and S7-400.

Protocol Type
ISO over TCP is the only protocol available.

Priority

Two priority levels are available: OP Mode (low priority) and PG Mode (high priority). Note that there may be a limit on the number of PG mode connections which are available.

Rack and Slot

Specify the CPU position. These can be determined from the hardware configuration. Note that these fields do not apply to the S7-200 CPU.

CP Type

If you are connecting to a Siemens PLC through a communications processor, select it here, otherwise select None.

Options include: CP-243, CP-343 and CP-443.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one.

If the last output values are stored, they may be re-written by either of two methods:

- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

If this driver is being used in conjunction with a Driver Multiplexer, then configure the Driver Multiplexer to store the last values, not the drivers connected to the Multiplexer. In this case, only the Multiplexer should be configured to re-write automatically.

Enable Auto Rewrite

If checked, the Store Last Output Values option will also be activated.

This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.

Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

Siemens S7 Driver Properties: Serial Tab

The Serial tab for the Siemens S7 driver tag properties folder consists of attributes used to identify and establish a connection to the control device to be associated with this driver.
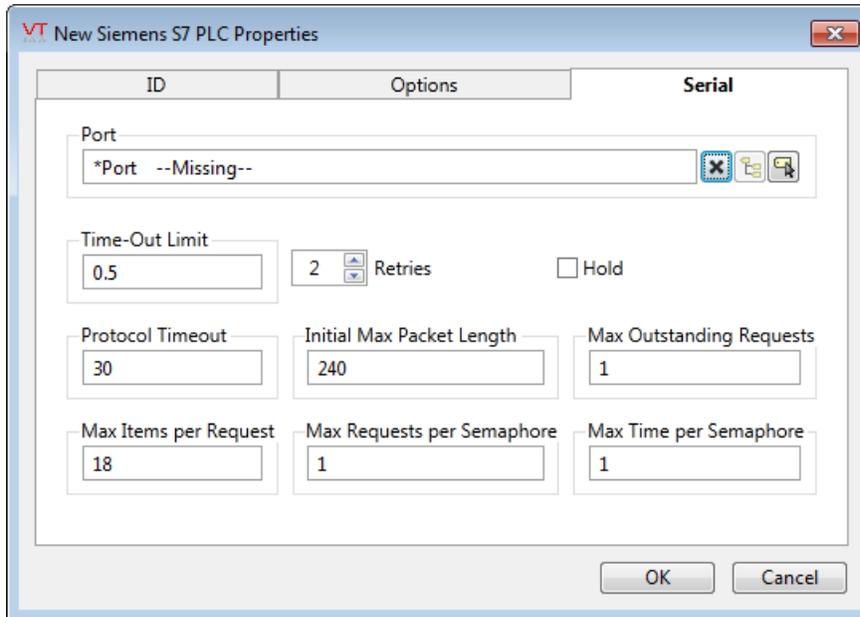
Port

> Since the only protocol available is ISO over TCP, you should configure and select a TCP Port.

Time-Out Limit and Retries

> A half second time out limit with one retry is recommended as a reasonable value for a Siemens PLC.

Hold

> In the event of an interruption in communication, this option will cause the last good value to be held. You should use this option with caution since, unless there is another means to alert you to interruptions in communication, it will appear that a valid value is still being received.

Protocol Timeout

> Timeout period, after which a lack of protocol traffic is considered an error. Default: 30 seconds.

Initial Max Packet Length

> Maximum PG/OP packet length used for initial communication with the PLC. This value will not be used after length negotiation with the PLC is complete. Default: 240.

Max. Outstanding Requests

> Defines the maximum number of PG/OP packets that may be outstanding.
> Default: 1 outstanding request.

Max Items Per Request

> Defines the maximum number of item addresses to include in a request.
> Default: 18 items.

Max Requests per Semaphore

> Defines the maximum number of requests that can be sent in one sem-
> aphore acquisition. Default: 100 requests.

Max Time per Semaphore

> Defines the maximum amount of time to hold the semaphore. Default: 1
> second.

Siemens S7 Driver Tag Widgets

The following widgets are available to display information about your
application's Siemens S7 driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

Siemens S7 Driver I/O Addressing

The syntax of the S7 memory addresses and the legal ranges for each of the supported PLCs are as follows:
The syntax for memory addresses takes one of 3 forms:

- [Area][Byte Address].[Bit Index]
- [Area][Data Size: B, W or D][Starting Byte Address]
- [Data Block Number DBx].DB[Data Size: B, W, X, or D][Starting Byte Address]

Memory Area	Area Prefix	S7-200 Bytes	S7-300 Bytes	S7-400 Bytes
Input	E(I)	16 (128 bits)	8K (64Kbits)	16K (128Kbits)
Output	A(Q)	5 (40 bits)	8K (64Kbits)	16K (128Kbits)
Memory	M	32	4K	16K
Special Memory	S(SM)			
Variable Storage	V			
Timer	T	512	1K	4K
Counter	Z(C)	512	1K	4K
Data Blocks	DBx DBy		2048, 64K each	8192, 64K each
Peripheral Input	PE(I)			
Peripheral Output	PA(Q)			
Analog Input	AE(I)		(n/a)	(n/a)
Analog Output	AA(Q)		(n/a)	(n/a)
Local	L			

SNMP Driver Tags

Not counted towards your tag license limit.
The SNMP driver can be used to communicate with any device that uses the Simple Network Management Protocol (versions 1, 2c or 3).

Note: Driver Errors: To learn more about the cause of an error condition, refer to the Driver Summary Report and the Driver Error Report, both of which are available in the Reports page. The Show Stats button will also provide current and last error messages: Show Statistics Button Widget

Reference Notes:

When using version 3, authentication may be done using either HMAC-SHA-96 or the MD5 protocol. Encryption uses either DES or 128-bit AES [1]. Choice of protocol is done using the following application properties, found in Settings.Dynamic:

- SNMPv3DefaultUserAuthProtocol (Set "0" for MD5 and "1" for SHA-1)
- SNMPv3DefaultUserPrivProtocol (Set "0" for DES and 1 for AES128)

[1] If circumstances require that multiple users or security levels be configured, then multiple SNMP drivers can be instantiated.

The following security levels are supported: NoAuthNoPriv, AuthNoPriv, and AuthPriv. There can be only one level per driver instance. When SNMP version 3 is selected, the contents of the Agent tab of the configuration folder change to allow selection of the security level and parameters for the User name and Context Name. The following application privileges are used to store the configuration: SNMPv3DefaultUserAuthKey, SNMPv3DefaultUserAuthProtocol, SNMPv3DefaultUserPrivKey, SNMPv3DefaultUserPrivProtocol.

I/O tags that are configured to use an SNMP driver will have an address selection feature. A button labeled "MIB" (management information base) will be added beside the Address field of the I/O tab. Clicking this will open an SNMP Address Select window, from which you can choose the correct Object Identifier (OID) address.

Note: The SNMP driver requires more configuration than other drivers. You will need to obtain the MIB library for your device from the manufacturer and you will need to tell VTScada where to find this information. See SNMP Addressing for details and configuration steps.

Related Information:

SNMP I/O Addressing

SNMP Agent Configuration is described in the VTScada Programmer's Guide

SNMP Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags

(note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

SNMP Driver Properties: Communications Tab

Use the Communications tab to select the port through which this driver will communicate as well as options for handling communication errors. In addition to the port parameters, this tab provides a location for you to specify the name of the SNMP group to which this driver instance will belong.
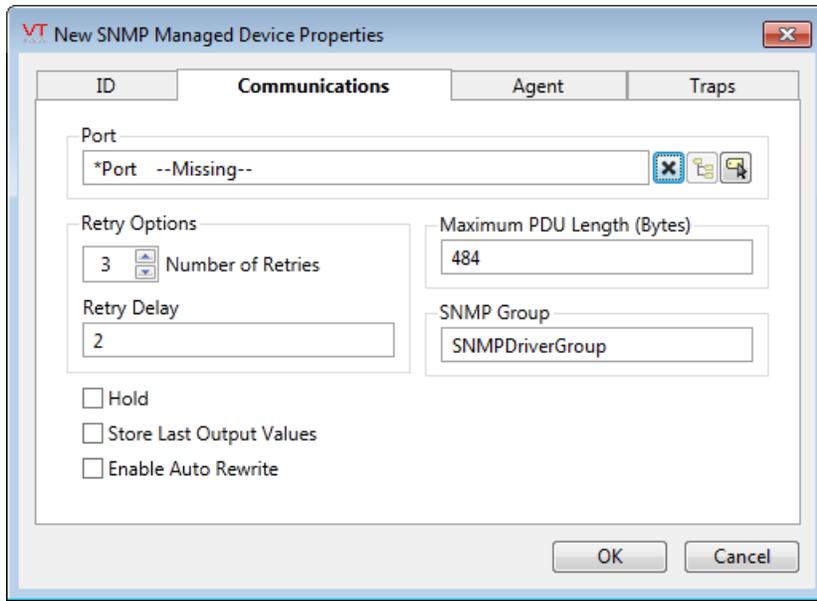
Port

Use the tag browser button to select the TCP/IP or UDP port tag to which the remote equipment is attached. The X button can use used to clear the current selection.

Retries

In the event that a connection cannot be made to the remote equipment on the first attempt, this field sets the number of communication retries that are to be attempted before the driver gives up with an error message.

Retry Delay

Sets the delay, measured in seconds, between retry attempts.

Maximum PDU Length (Bytes)

Use to specify the maximum length, as measured in bytes, of the Protocol Data Unit. This value need only be changed from the default if a different value is required by the equipment

with which you will be communicating.

SNMP Group

The name of a VTScada RPC service to which this driver instance will belong. Alternate RPC services need only be created if your application uses multiple servers on separate subnets. In most applications you should leave the default value for this field.

Hold

In the event of an interruption in communication, this option will cause the last good value to be held. You should use this option with caution since, unless there is another means to alert you to interruptions in communication, it will appear that a valid value is still being received.

Store Last Output Values

When checked, the driver will maintain a record of the last value written to each output address. This may be useful in at least two situations:

- For hardware that does not maintain its state during a power loss and must be restored to that state when re-started.
- When failed hardware is replaced by a new device and you would like to start that device with the values last written to the old one.

If the last output values are stored, they may be re-written by either of two methods:

- Automatically, when communication is restored to the device.
- Manually by way of a button press. See, Rewrite Outputs Widget for details.

Changing this value from checked to unchecked will cause all stored values to be erased immediately.

Enable Auto Rewrite

> If checked, the Store Last Output Values option will also be activated.
>
> This option causes the driver to automatically rewrite the last value written to each output, in the event that communications are lost, then restored.
>
> Use this option only if you are certain that you want the last values to be rewritten automatically after an interruption in driver communications.

SNMP Driver Properties: Agent Tab

Provides configuration options related to the SNMP device with which this driver will communicate.
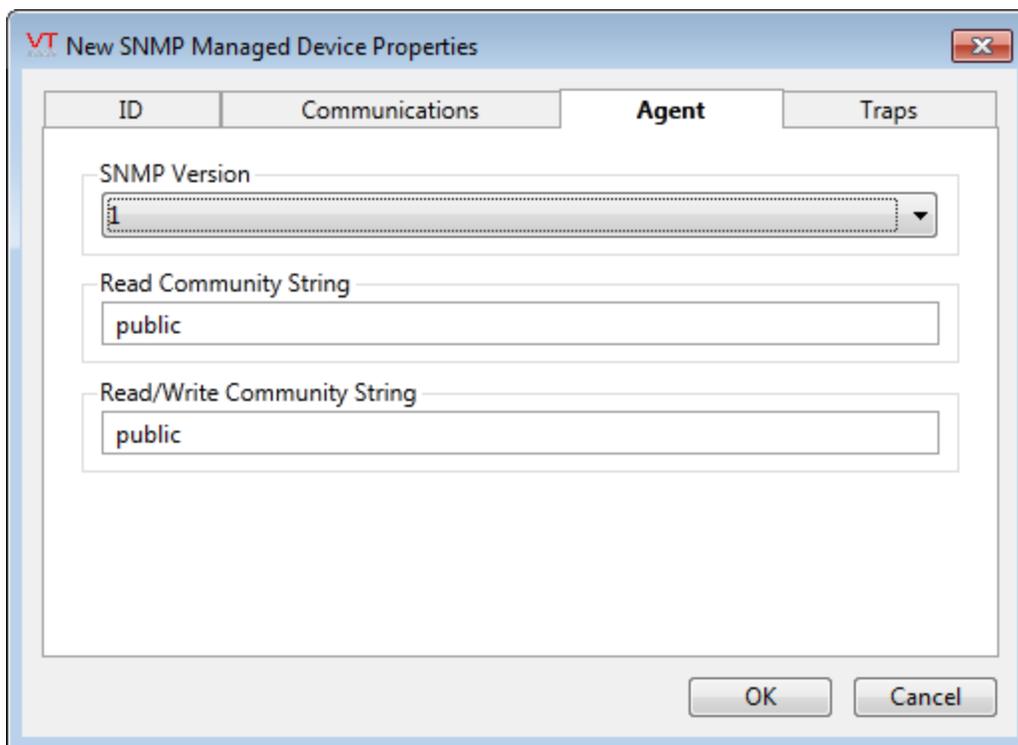
Format for SNMP 1 and 2c.

Format for SNMP 3

SNMP Version

Select the simple network messaging protocol version that the agent is using. Supported versions include 1, 2c and 3. Defaults to 1. Selecting version 3 will cause the available configuration parameters to change as shown

Read Community String (Versions 1 and 2c)

Used for authentication of the device. The default value of "public" should have been changed on the device in the interest of improved security.

Read/Write Community String (Versions 1 and 2c only)

Similar to the Read Community String, this value is used to authenticate an SNMP device that will be used to both read and write information. Like the former, this value should be

changed on the device from the default of "public".

User Name (Version 3 only)

The name required for authentication.

Context Name (Version 3 only)

A context is a collection of information accessible by an SNMP entity. A SNMP Engine may contain one or more contexts, where the default context has an empty name. Information is not constrained to a single SNMP Engine ID or context; the same information can be available via different means. A context is used to define information subsets. For example, an SNMP Engine exposed to this driver might contain multiple components, each of which implements MIB items that may have conflicting OID info. The concept of a context was introduced to avoid this ambiguity. For simple devices the default context is sufficient.

SNMP Driver Properties: Traps Tab

Note: In general, it is recommended that the Allow Traps option be disabled unless explicitly required.

Used to specify the port on which the driver will listen for unsolicited (Trap) messages. The intended purpose of Traps in SNMP is to notify the network management service so that it can investigate as required. They are not intended to set data, since doing so represents a weak point in security (although, many devices use Traps to do exactly that). Traps typically contain messages such as a timestamp and an event OID such as a device restart.
The port number defaults to 162. Should not match the local or the remote port number.

You may set the port number to zero (0) to prevent the SNMP driver from accessing the port. This may be useful in the case that the driver should not open a port or receive traps.
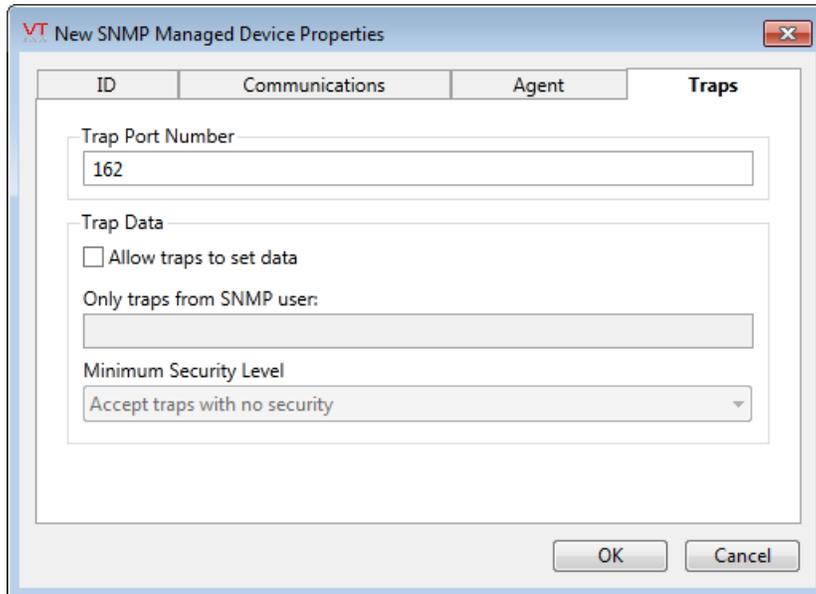
The following notes are for SNMP version 3 only:

Additional configuration options, within the Trap Data block will be enabled when SNMP v3 is in use.

When filtering for a specific SNMP user, check that the minimum security level is set as required. Note that the user/security–level/context setup on the Agent tab is for commands (Get/Set/GetBulk) and has no effect on traps.

Security levels available when SNMP v3 is in use include:

- Accept traps with no security
- Only accept authenticated traps.
- Only accept authenticated and encrypted traps.

SNMP Driver Tag Widgets

The following widgets are available to display information about your application's SNMP driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Comm Messages Button Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Rewrite Outputs Widget

SNMP I/O Addressing

SNMP addresses are built using a hierarchy of OID[1] values. These can be long. The VTScada SNMP Address Select tool, found in every I/O tag that uses an SNMP driver, will help you build addresses for your I/O tags. Before using the Address Select tool, you must first tell VTScada where to find the MIB[2] file for your device. You will need to obtain a .MIB file from your equipment manufacturer.

It is possible to create a VTScada SNMP Agent service, to serve tag values over SNMP.

SNMP Agent Configuration is described in the VTScada Programmer's Guide

SNMP Address Configuration

---

[1]Object IDentifier. A part of the SNMP driver addressing system.

[2]Management Information Base. A hierarchy of the information available to an SNMP device, organized by numbered Object Identifiers (OIDs).

There are two parts to address configuration. The first is to obtain the required .MIB file(s) and import them into your application. The second is to build the address using the built-in SNMP Address Select dialog (shown later in this topic).

Importing a .MIB

1. Obtain a .MIB file from your equipment manufacturer.
2. Save this file to a folder on your VTScada server computer.

Note: All .MIB files that will be used by an application will be stored a folder named "MIB" within your application directory. You can create this folder yourself and copy the files to it. If using the Import MIB button, VTScada will create the folder and copy the selected files to it.

3. Import the file into the application's version control system.
   There are two ways to do this:
   a. Click on the application's Import File Changes button in the VAM.
      Or:
   a. Create and configure an SNMP driver tag in your application.
   b. Select that driver in an I/O tag, such as an Analog Input.
      The address field of the I/O tag will now have a MIB button.
   c. Click the MIB button – the SNMP Address Select dialog will open.
   d. Click the Import MIB… button.
   e. Select the file.

Note: If the .MIB file depends on other .MIB or .MY files, you will get an error message in the Address Select tool (step d. above). Hovering over the error will open a tool-tip window, display the name of the missing file. Locate and copy this file to the MIB folder, then use the Import File Changes button in the VAM.

Building an address.

The following instructions assume that you are configuring an I/O tag that uses an SNMP driver and have reached the stage of building the address.
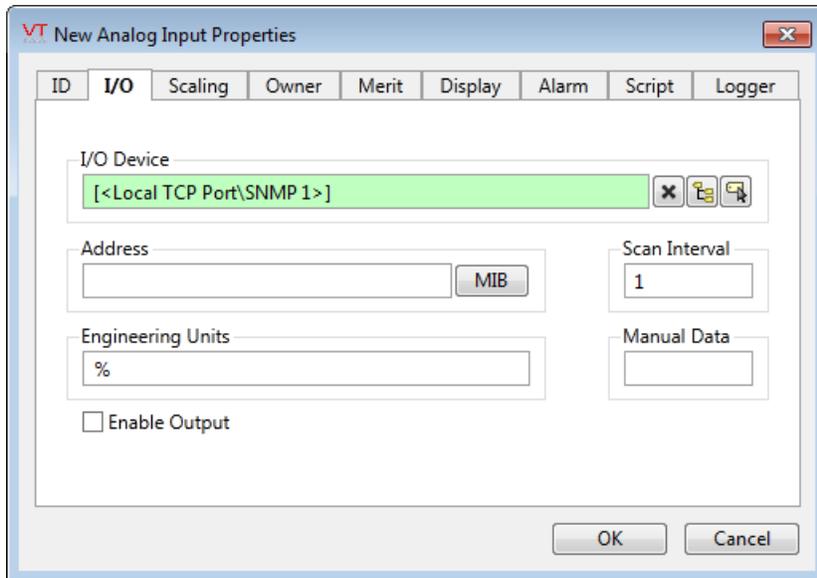
A MIB button can be found beside the address field when an SNMP driver has been selected as the I/O Device.

1. Click on the MIB button.

2. Navigate through the tree to select the device you wish to address.
   Note that you can hover over a node to see extra information.

3. If required, enter the Table Index number, specifying the row in the SNMP table.
   The Table Index is the last number in the SNMP address string.

4. Click OK.
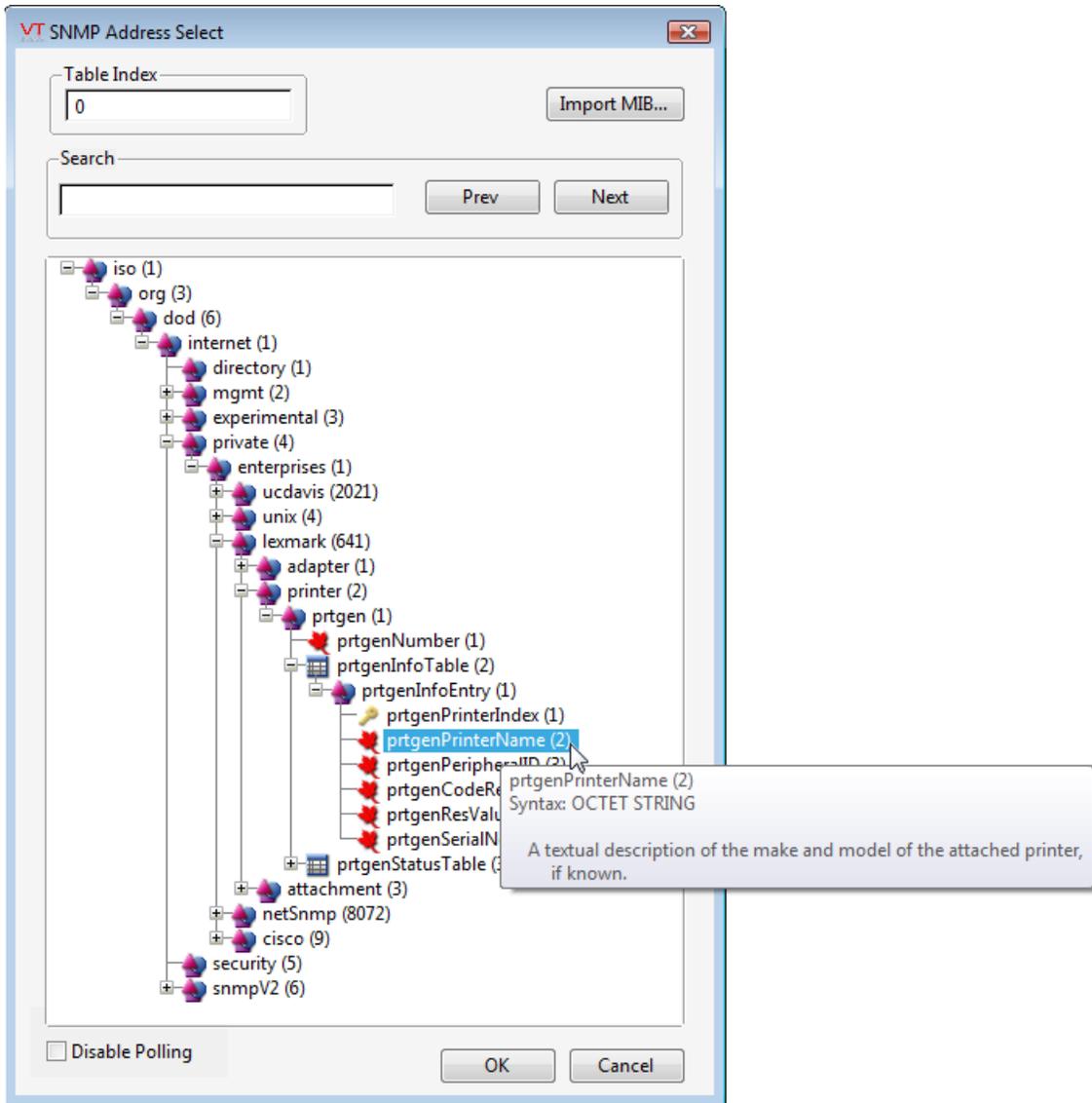
The result will be similar to that shown here:
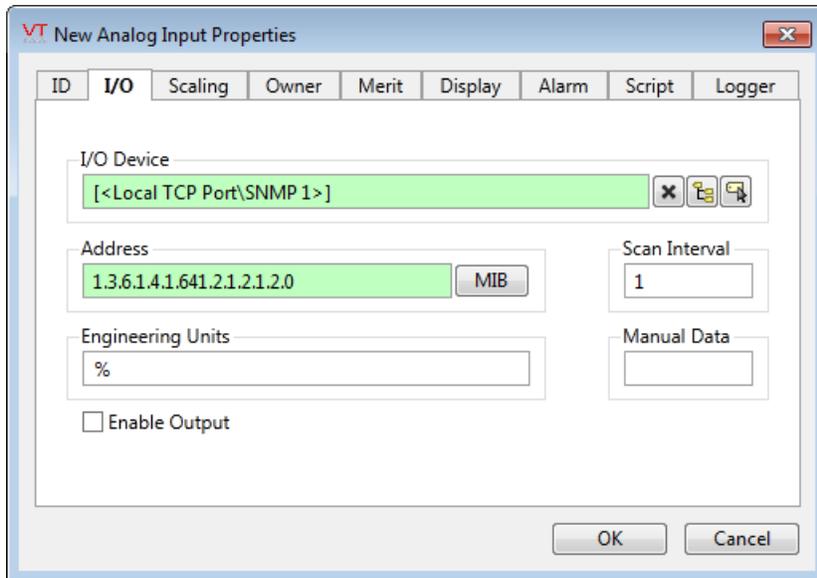
SQL Data Query Driver Tag

The SQL Data Query driver provides a means for a VTScada application to capture data from an ODBC-compliant database and to place that data into the application's Analog Status, Digital Status, Analog Input, or Digital Input tags. The driver allows time-stamped historical data in the database, or current values, or both to be read into VTScada tags and sent to the VTScada historian.

To use the driver, do the following steps in order:

1. If you plan to use an ODBC data source name for the connection, create this using the Microsoft ODBC administration tools. Choose the 32-bit or 64-bit ODBC administrator as appropriate for the data source. VTScada can connect to either.

2. Create and configure one or more SQL Data Query Driver tags.
   At least one tag will be required for each ODBC source used. Additional tags may be required if more than one SQL Query template is required.

3. Create the I/O tags that will receive data from the driver tags.
   Use String I/O tags if retrieving text values.
   If using Analog Status tags, ensure that you set the scaling parameters appropriately. In most cases, the Unscaled Data Process Max and the Scaled Data Process Max values should be identical so as to avoid scaling the retrieved value.

4. Draw the SQL Data Query Driver tag using the built-in drawing widgets. The Show Stats button and a Communication Indicator widget are both recommended.

Related Application Properties:

The following property settings hold additional configuration parameters for your SQL Data Query driver:

SQLDataQueryDriverDefaultDBType – The default database type is 2 (Oracle).

SQLDataQueryDriverDefaultTableName – The default table name is "RealAnalog".

SQLDataQueryDriverMaxTagsPerQuery – Limit on the number of tags per query. Defaults to 25.

SQL Data Query Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

SQL Data Query Properties: Database Settings Tab

The Database Settings Tab holds the properties required to connect to the database that you will query.
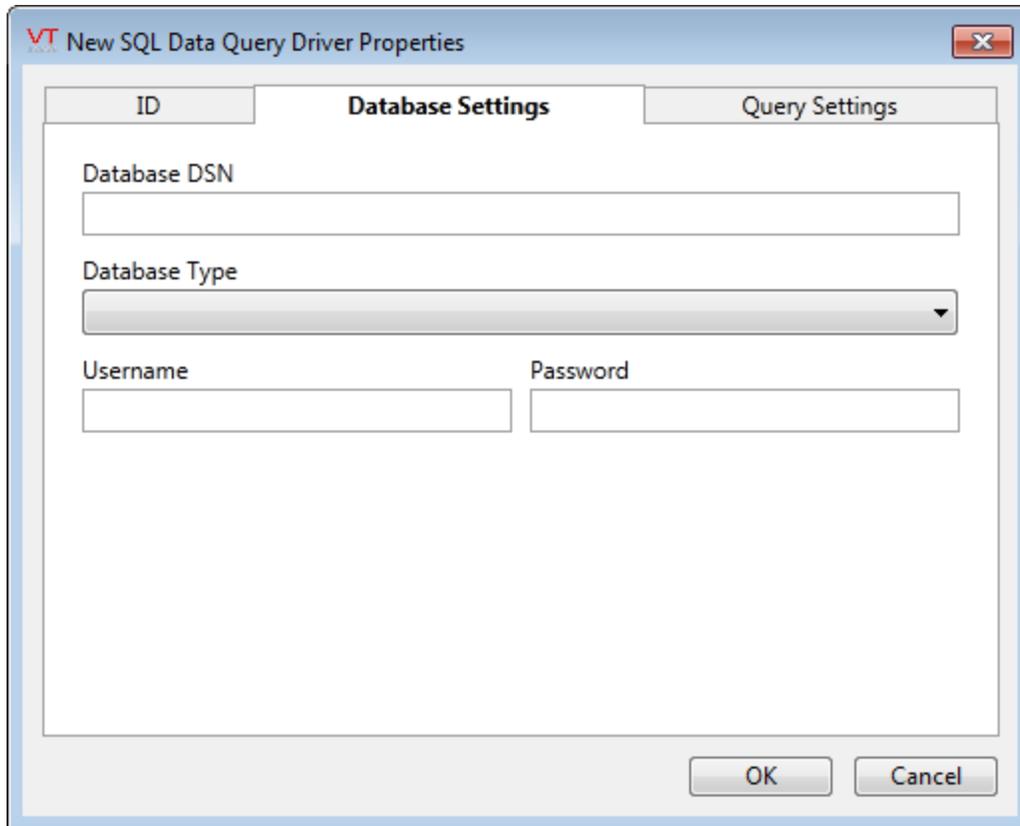
Database DSN

The Data Source Name (DSN) is configured using the Microsoft ODBC Administrator tool. It is used in place of a connection string in order to access a database via an ODBC-compliant driver.

Database Type

In addition to the usual types supported by VTScada: Oracle, MySQL, and MS-SQL Server, you can select from Access, PolyHedra or SyBase.

Username and Password

Provide the credentials that are required to connect to your database program.

SQL Data Query Properties: Query Settings tab
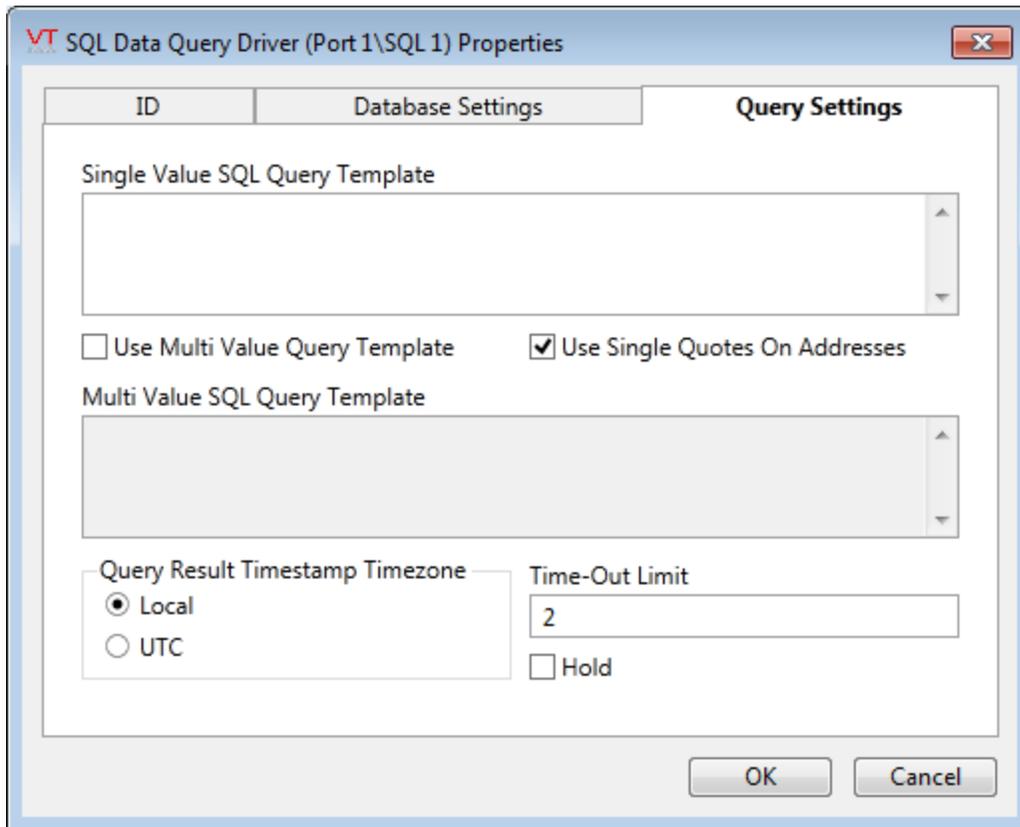
Single Value SQL Query Template

Use this if either of the following is true:

- The data will be queried in a format of one query per address (input tag) or
- Only one tag address is supplied (attached) to the driver.

Templates are the SQL selection queries that you want to execute. The table name and where-clause values to query are written as replaceable parameters, with values coming from the I/O tags using this driver. For more information, refer to

the section, "Creating Queries for the SQL Data Driver", found within this topic.

Multi Value SQL Query Template

The tag query template to use if multiple tag addresses will be returned by a single query.

Use Multi Value Query Template

Force the use of the "Multi Value SQL Query Template" if more than one tag is attached.

Use Single Quotes on Addresses

Check this box to apply single quotation marks to all tag addresses / where-clause parameters, when they are inserted into a query.
Select this option if querying based on text values. Deselect this option if querying based on numeric values.

Hold

Check this to have I/O tags attached to the driver hold their last value in the event of a communication failure with the data-base. If not checked, tags will have their value set to invalid on a communication failure.

Query Result Timestamp Time Zone

Using these radio buttons you can display the timestamp within your results using either the local time zone or UTC. Note that the VTScada Historian stores all data using UTC, but this might not be the case for the database that you are query-ing.

Time-Out Limit

Time limit in seconds for queries to the database server. Queries that exceed this time will be declared as a communication failure and data for associated tags may be invalidated, depending on the state of the "Hold" setting.

Creating Queries for the SQL Data Driver

The SQL Data Query Driver relies on user-defined query templates in combination with parameters supplied by the address fields of the associated tags to create configurable queries that extract data as needed. Use this to create generic query template to be created at the driver level that are then modified by the contents of the attached input tag address fields through the use of replaceable tokens. This enables the driver tag to read a database table containing data for multiple tags with a single query definition.

A user defined query template can contain the following tokens:

- ^A – ID value token
- ^T – Table name token

Several examples following show how these can be used in conjunction with the tag addresses to create queries to extract database table values into VTS tags. Note that query templates may contain modifiers as appropriate for the database being queried, such as "LIMIT" (MySQL, Oracle), "TOP" (Access, MS-SQL), "ORDER BY", etc.

Required Query Result Format

All queries returned from the database must be structured in a multi-field format following one or the other of the formats described following:

- Option 1 – ID and Value

  The data will be returned as two columns with the first field containing a text string or numeric value that matches the ID portion of the address field of the tag for which it is destined.

The second column will contain the data value for the tags that match the address ID in the first field. For example, if an Analog Status tag is created with its address set to "ABCDE", a query resulting in the following returned data set would assign a value of 36.7 to the raw value of the tag:

Field 1 – ID	Field 2 – Value
ABCDE	36.7
AAAAA	56.8
GHGHG	99999

If two other tags are created with their addresses set to "AAAAA" and "GHGHGH", their raw values would be set to 56.9 and 99999 respectively by the same query result. In this example, the timestamp of the Analog Status tags would be set to the current time as a result of the query result as there are no timestamps in the data set returned from the database.

- Option 2 – ID, Value, and Timestamp

An optional third timestamp field may also be returned from the database and if present will be automatically detected by the driver and used as the timestamp for the tags. For the above example, if the table to be queried was instead formatted as shown here:

Field 1 – ID	Field 2 –Tag Value	Field 3 – Timestamp
ABCDE	36.7	2014-06-02 12:34:56
AAAAA	56.8	2013-06-02 13:44:16
GHGHG	99999	2012-06-02 14:54:06

Then the Analog Status tags with addresses "ABCDE", "AAAAA", and "GHGHGH" would be assigned values 36.7, 56.8, 99999, and timestamps "2014-06-02 12:34:56", '2013-06-02 13:44:16", & "2012-06-02 14:54:06" respectively.

Using Query Templates

The driver tag's query templates function by allowing the user to create a generic query that will have some of its content replaced by data extracted from the address fields of the associated I/O tags through the use of the ID token ^A or the table token ^T. The examples following show several ways in which data can be read from a database using the query templates in conjunction with the tokens.

Example #1 – Single Tag Value from a Table
This example reads a single value from a specific table.
Given a query template configured as:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = '^A'
```

And, an Analog Status tag with its address set to AIT0045,
Then the resulting query would be:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = 'AIT0045'
```

The result is a two-field response where the first field is a match for the address field of the attached Analog Status tag.

An additional setting of the driver that makes building queries of this type easier is the "Use Single Quotes on Addresses". If checked, then the tag addresses are automatically delimited by the single quotation marks so the query template can be changed to:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = ^A
```

And, the resulting query will be the same. This option should be used only when querying based on text values, not numbers.

Example #2 – Multiple Tag Values from a Table Using Multiple Queries
If multiple tags need to be extracted from a single table, then the template shown in example #1 will function by executing one query for each connected tag. For example, if three tags are connected to the driver with addresses, AIT0045, PT0679, and FIT0098, then three queries are generated by the driver:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = 'AIT0045'
```
```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = 'PT0679'
```

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName = 'FIT0098'
```

This example is contingent on the following settings being made in the driver:

- The template is entered in the "Single Value SQL Query Template" setting
- "Use Multi Value Query Template" is NOT checked

Note that the setting "Use Multi Value Query Template" enables the user to select which template to use. If not checked, the "Single Value SQL Query Template" is always used. If checked, the "Single Value SQL Query Template" is used only if there is 1 tag attached to the driver, while the "Multi Value SQL Query Template" is used if there are multiple tags attached to the driver.

Example #3 – Multiple Tag Values from a Table Using a Single Query

As noted above in Example #2, a query template can be used to query data for multiple tags in a single query, provided that the query result correctly correlates the tag addresses in the first field with the data in the second field. If, for the above example, the "Use Multi Value Query Template" setting is checked, then the following template entered in the Multi Value SQL Query Template field:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName In (^A)
```

would result in a query structured as:

```
SELECT TagName, TagValue FROM DataLogTable WHERE TagName In
('AIT0045','PT0679','FIT0098')
```

The dataset returned from this query would contain multiple records and the driver will assign the data to the appropriate tags based on the contents of the TagName field returned as the ID.

Example #4 – Using Table Name in the Address

If each tag requires data to be extracted from a different table, then a table name may be added to a tag's address and from there used to populate the table name field of a query automatically. Using this template:

```
SELECT TagName, TagValue FROM ^T WHERE TagName = '^A'
```

Where used with tags having the following addresses:

AIT0098/DataHourly

PT1111/DataDaily

FIT0108/DataMonthly

Then, the resulting queries would be in the form:

```
SELECT TagName, TagValue FROM DataHourly WHERE TagName = 'AIT0098'
```

```
SELECT TagName, TagValue FROM DataDaily WHERE TagName = 'PT1111'
```

```
SELECT TagName, TagValue FROM DataWeekly WHERE TagName = 'FIT0108'
```

Note that this example assumes that all three tables contain the columns, 'TagName' and 'TagValue'.

Example #5 – ID Does Not Exist In Database

In some instances, there may be a need to extract data from a database table where no unique ID exists to identify the data other than the field names in the table. For example, if a database contains a table named "LastReadings" that contains only 1 record with the following fields:

    ReadingTime
    Temperature
    Pressure
    FlowRate

If you wish to extract the most recent of these three readings into three individual tags, then the following query template could be used:

```
SELECT '^A', ^A, ReadingTime FROM LastReadings
```

By setting the tag addresses to "Temperature", "Pressure", and "Flow", the following queries will be created:

```
SELECT 'Temperature', Temperature, ReadingTime FROM LastReadings
```

```
SELECT 'Pressure', Pressure, ReadingTime FROM LastReadings
```

```
SELECT 'FlowRate', FlowRate, ReadingTime FROM LastReadings
```

Note the use of the ID in two places in the list of fields to select. By placing the first of these inside of single quotation marks, the query result will return this field value as a text string, which will then meet the driver requirements to have the first field match the ID portion of the address

set by the tag. The second field returned will contain the corresponding data required for the tag from the named field itself.

Example #6 – Multiple columns of readings per row

In many databases, there will be many values per row in a table. For example, consider the following table named FieldReadings which has temperature, pressure and flow, all identified by ID:

ID	Temperature	Pressure	Flow
1001	50.6	100.4	20.5
1002	53.7	120.3	15.5

One approach is to define three separate driver tags, with the following query templates:

```
SELECT ID, Temperature FROM FieldReadings WHERE ID IN (^A)
```

```
SELECT ID, Pressure FROM FieldReadings WHERE ID IN (^A)
```

```
SELECT ID, Flow FROM FieldReadings WHERE ID IN (^A)
```

This benefits from driver read coalescing, in that the driver can efficiently retrieve data for multiple tags with a minimum of calls to the database. However, this comes at the cost of more complex tag configuration, where I/O tags must be configured to point to different drivers that all reference the same table.

Another approach is to override the use of the ^T table placeholder to instead refer to the column from which to retrieve the data.

The query template for the single instance of the driver tag would look like the following:

```
SELECT ID, ^T FROM FieldReadings WHERE ID IN (^A)
```

The tag addressing would look as follows:

```
1001/Temperature
```

```
1001/Pressure
```

```
1001/Flow
```

This has the benefit of reducing the number of driver tags to one and reducing configuration complexity across the system. However, due to the addressing scheme used, the driver can not take advantage of read

coalescing, which will result in a database query for each tag value requested. This approach would be of use in a smaller system with a limited amount of tags that wish to query a fast database platform at spaced intervals. This approach may not be suited for large-scale systems querying data at high rates.

SQL Data Query Driver Widgets

The following widgets are available to display information about your SQL Data Query driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Color Line Widget

...Show Statistics Button Widget

...Gradient Color Change Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

Related Information:

...SQL Data Query I/O Addressing

SQL Data Query I/O Addressing

Input tags using the SQL query driver should use the following format in their address fields:

```
ID/Table:HIST
```

Where:

ID – Denotes a text string or numeric value that will uniquely identify the tag to which it is associated. This is used to match data returned from the database with the tag.

/Table – The optional table name.

:HIST – An optional suffix on the address to force the driver to configure the data returned from the database in a format that is compatible with the "History Address" settings for the analog status and digital status tags. This format is typically used when retrieving logged historical records from a database into the VTScada historian.

Examples of valid addresses:

Address	Description
AIT09876	Simple text ID (In the driver configuration, select 'Single quotes should be specified on addresses')
9087	Simple numeric ID
PIT9087.PV/DataLog	ID "PIT9087.PV" including a table named "DataLog".
PIT9087.PV/DataLog:HIST	ID "PIT9087.PV" including a table named "DataLog" used in history retrieval mode
9087:HIST	ID "9087" used in history retrieval mode

Related Information:

...SQL Data Query Driver Tag

...SQL Data Query Properties: Query Settings tab

Input Tags

Input tags are used to read data from an I/O device, such as a programmable logic controller (PLC), remote terminal unit (RTU), or an I/O board. These tags permit the handling of either digital (discrete), or analog (continuous floating point) values.

All input tags require a driver tag, which will provide the communication protocol required for the device. In general, it is best to configure a driver tag first, then add input tags later as children of that driver.

The following input tags are included with VTS:

Analog Input Tags   Analog Status Tags

Digital Input Tags   Digital Status Tags

Pulse Input Tags   Pump Status Tags

String I/O Tags

Related Information:

...Communication Driver Tags – Driver tag reference

...Working With Tags – The Tag Browser – Using the Tag Browser to create and edit tags.

Comparison of Analog Input Versus Analog Status Types

Analog Status and Analog Input tags are used for similar purposes. Both are used to monitor analog values and, with only one exception, the same widgets are available to each. Either could be used for the same PLC I/O address and when drawn on the screen, there is nothing to indicate which tag has been used.
A similar relationship exists between Digital Input and Digital Status tags.
A common question then is, "which type should I use?". The following comparison of features should help you choose. Any tag attribute not listed in the following table is the same for both.

Analog Input	Analog Status
No built-in logger	Logger built in, logging on change
No deadband option	Deadband option – used to set a minimum change required before logging a new value.
No built-in alarm	High and low alarms built-in.
Does not support history reads	Supports history reads when used with drivers that have this feature.
Not drawn on VTScada Station Pages	Drawn on VTScada Station Pages
Does not include the "Draw" widget	Includes the "Draw" widget
Can be linked to an "Owner" tag.	Does not have the "Owner" option.

The feature comparison shows that Analog Status tags have several advantages. There are more built-in features, reducing the need to use extra tags to provide logging and alarms. Analog Status tags also support history reads and are displayed on Station Pages. If you plan to use any of these features, then an Analog Status tag is probably your best choice.

Why then might you ever use an Analog Input tag? First, the dedicated Logging or Alarm tags that you add to an analog input will have more features and therefore more control than the built in logging and alarms of the Analog Status. Secondly, if you do not need the extra features of the Analog Status tag, the smaller Analog Input is a more efficient choice. Finally, for applications that have been coded to make use of tag aggregation, the Owner option of the Analog Input is an essential feature.

When you do add Analog Status tags to your application, there is one extremely important configuration option to remember: You should always set a value for the deadband. The automatic logging feature of these tags will add a new record each time the value changes, regardless of how small that change is. To avoid filling the log file with system noise, you should set the deadband to a small value, thus causing the tag to ignore any change less than this amount. For example, on a input that changes from 0 to 100, you might set the deadband between 0.1 and 1.

Related information that you may need:

Analog Input tag reference. Analog Input Tags
Analog Status tag reference. Analog Status Tags

Analog Input Tags

Counted towards your tag license limit.
The role of Analog Input tags is to represent incoming analog data values from equipment processes, and provide widgets to display the data in numeric or graphic form as a bar, gauge, or number.
Analogs can have text values if the driver returns a text value and if all four of the scaling values are set equal. Standard practice is to set all four scale values to 1.

Analog Input Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
> The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Analog Input Properties: I/O Tab

The I/O Tab contains the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU). This is done by specifying the driver tag that communicates with the physical I/O device, the address at the physical I/O device from which this tag is to read its data, and the rate at which the I/O device should be scanned for data.

If supported by the communication driver, the Analog Input tag may also be used to write data.

I/O Device

> Select the communication driver tag from which data will be read. By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Address

> Provide the address within the I/O device from which this tag is to read data. This value must match the configuration of your PLC or RTU hardware. Refer to the Addressing topic for your particular device driver for guidance. For some drivers (SNMP and the OPC Client) an address browser is provided to assist you.

Scan Interval

> Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Engineering Units

Provide units of measure that the input data represents. Possible values for this field include "rpm" "degrees C", "%", etc.

You can add a degree symbol to the engineering units by holding down the Alt-key while typing 0176. Release the Alt-key before typing the C or F.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Enable Output

If checked, this input tag may also be used to write data to the specified address of the communication driver. A Security Privilege may be set in the Merit tab to restrict access to this feature.

Analog Input Properties: Scaling Tab

The Scaling Tab requires values that will be used to convert unscaled, raw data from the I/O device associated with this tag into scaled data that will be reported to operators. VTScada uses an algorithm to calculate this tag's value internally, based on the known unscaled and scaled values you supply.

The default value for each of these fields is stored in an application property. You are advised to change those properties to match the values used by your hardware by using the Edit Properties page of the Application Configuration dialog. You will need to use the Advanced mode of the properties page. The properties to copy and edit are:

- AnalogUnscaledMinDefault
- AnalogUnscaledMaxDefault
- AnalogScaledMinDefault
- AnalogScaledMaxDefault

Assuming a linear relationship between the values, by providing a known minimum scaled value for a given minimum unscaled value, and a known maximum scaled value for a given maximum unscaled value, VTScada can provide a scaled value for any given unscaled input value.

> Note: Analogs can have text values if the driver returns a text value. Set all four of the scaling parameters to "1" in order for the tag to read text values.

Analog Input Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag.

> Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data.
An owner tag is one which you must design and then create,
using the VTScada scripting language.
The owner tag may keep track of different aspects of each con-
tributor's data, from the presence of a user-defined manual
data value, to questionable data, according to the con-
figuration of the check boxes appearing beneath the Owner
field. These check boxes also determine the way that this con-
tributor tag's value should be used in the owner tag's cal-
culations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is
used to set the value of this contributor tag as the nth element
in the owner tag's array. You may choose to set this con-
tributor tag's value in more than one of the owner tag's array
elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable

data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Analog Input Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
The Quality Based On field is for use when this tag is to be a contributor to a custom-built container tag. Please see the topic Merit Tab and Quality Tab for details.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege. Applies only if the Enable Output option is checked in the I/O tab.

Analog Input Properties: Display Tab
When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.

Analog Input Properties: Alarm Tab

If you want this tag to trigger an alarm, use the Add button to open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Analog Input Properties: Script Tab

Optionally, link one or more Script tags to this tag. A Script tag provides a means of creating a procedure, using VTScada's programming language, that will run whenever this tag's value changes.

Analog Input Properties: Logger Tab
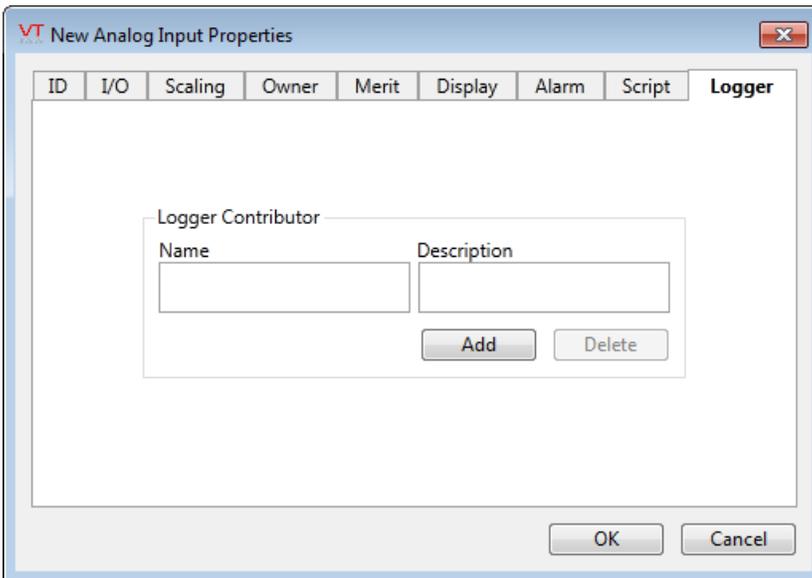
Analog Input tags do not have a built-in connection to a Historian and therefore must be linked to a Logger tag if data is to be stored. The Logger tag works with an attached Historian to record this tag's data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Analog Input Tag Widgets

The following widgets are available to display information about your application's Analog Input tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
Compass 1 Widget	Compass 2 Widget
Draw Text Widget	Droplist Control Widget
LCD Bar Widget	Image Change Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Glass Gauge Widget	Classic Gauge Widget
Retro Gauge Widget	Cockpit Gauge Widget
Aqua Gauge Widget	Army Gauge Widget
Backlit Gauge Widget	Nautical Gauge Widget
Quarter Arc Gauge Widget	Linear Gauge Widget
Antique Gauge Widget	LCD Ring Widget
LCD Arc Widget	LCD Meter Widget
Momentary Button Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Set Analog Value Widget
Set Value Button Widget	Set Value Hotbox Widget
Slider Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget

Rotary Control Widget (Knob)	Indicator Light Widget
Equipment / Status Color Indicator	LCD 7 Segment Widget
LCD 5x7 Matrix Widget	

Related Information:

Comparison of Analog Input Versus Analog Status Types

Analog Status Tags

Counted towards your tag license limit.

It is the role of Analog Status tags to read analog data such as well level or a motor speed from an I/O device.

> Note: **Values smaller than the Deadband setting will not be logged! Always check the settings in the Historian tab.**
> Analog Status tags will apply a default deadband of 1/4 of one percent of your display scale.
> Read about the deadband in the section for the Historian Tab. Do not risk either a value so low that it logs system noise or a value so high that important information is lost.

Analog Status Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.
There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.
Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Analog Status Properties: I/O Tab

The I/O Tab includes properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the polling driver responsible for determining the order and rate at which data polls will occur.

I/O Device

Select the communication driver tag from which data will be read.By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Address

Provide the address within the I/O device from which this tag is to read data. This value must match the configuration of your PLC or RTU hardware. Refer to the Addressing topic for your particular device driver for guidance.

For some drivers (SNMP and the OPC Client) an address browser is provided to assist you.

Scan Interval

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

History Address

If set, then the address field becomes optional. This field provides a means of reading values as recorded by a data logger.

If both the address and the history address are provided, then the address will be polled for data at the interval set by the Scan Rate, but the values read from the data logger will overwrite the values logged by this tag when it is updated.

The form for the history address will depend on the RTU. Please refer to the addressing reference chapter for your driver.

Note that it is not possible to enter a parameter expression directly in the History Address field. As an alternative, you may enter compound address in the Address field using a parameter expression. For example, if Addr, HistAddr and HistoryRate are all properties of a parent tag, the following expression in the Address field will create a compound address with all three components:
concat(.\Addr , "):(" , .\HistAddr , "):(" , .\HistoryRate)

The characters "):(" separate each component of the address. Do not add a leading or trailing parenthesis around the entire expression.

Further information on reading history information can be found in the topic SCADAPack History Read

History Scan Interval

Sets the time between polls of the history data logger. If a History address has been set, but this field is left blank, then it will default to the value set in the Scan Rate field. The recommended value for this field is 60 seconds. Lesser values may cause communications to slow down.

Note that it is not possible to enter a parameter expression directly in the History Scan Interval field.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a

new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Enable Output

If checked, this input tag may also be used to write data to the specified address of the communication driver. A Security Privilege may be set in the Quality tab to restrict access to this feature.

Analog Status Properties: Scaling Tab

The Scaling Tab requires values that will be used to convert unscaled, raw data from the I/O device associated with this tag into scaled data that will be reported to operators.

Note: **Values smaller than the Deadband setting will not be logged! Always check the settings in the Historian tab.**

Analog Status tags will apply a default deadband of 1/4 of one percent of your display scale.

Read about the deadband in the section for the Historian Tab. Do not risk either a value so low that it logs system noise or a value so high that important information is lost.

Note: Analogs can read text values if those are returned by the driver. Set the first four of the scaling parameters to "1" in order for the tag to interpret data as text. Alternatively, use the String I/O tag for reading text data.

Note: The first four fields on this tab affect the tag's value, scaling it from the input range to a desired output range.

The default value for each of these fields is stored in an application property. You are advised to change those properties to match the values used by your hardware by using the Edit Properties page of the Application Configuration dialog. You will need to use the Advanced mode of the properties page. The properties to copy and edit are:

- AnalogUnscaledMinDefault
- AnalogUnscaledMaxDefault
- AnalogScaledMinDefault
- AnalogScaledMaxDefault

Assuming a linear relationship between the values, by providing a known minimum scaled value for a given minimum unscaled value, and a known maximum scaled value for a given

maximum unscaled value, VTScada can provide a scaled value for any given unscaled input value.
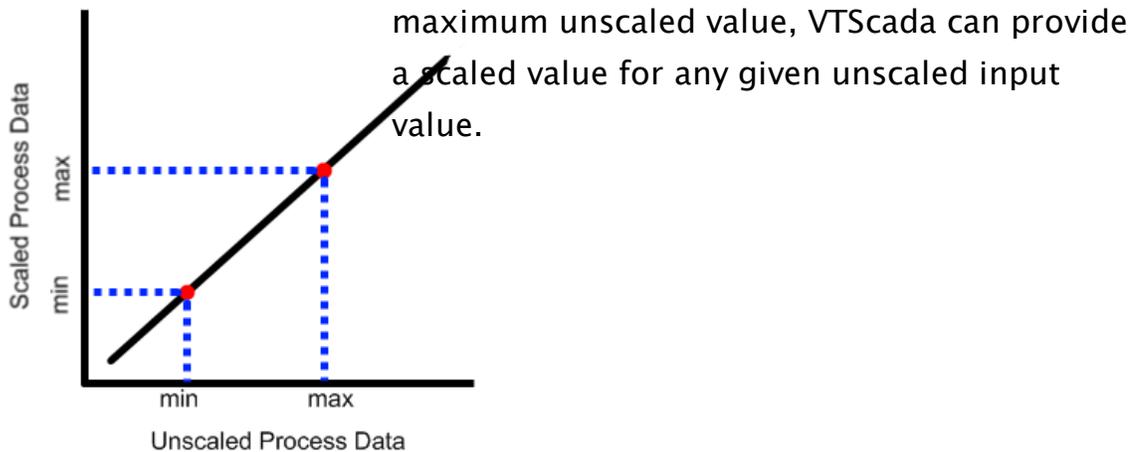
Low and High Scale Values

Note: These two fields have no effect on the value of the tag.

Set the Low and High scale values to define the default scale range for the HDV trend view and certain widgets, such as the various meters. Use these when values are expected to stay within a portion of the full scale range and it is therefore helpful to adjust the displayed range as needed.
If left blank, the Scaled Process Data Minimum and Scaled Process Data Maximum values will be used as the Low and High Scale values.

Engineering Units

Provide the units of measure that the calculated value represents. This text will be used by widgets that include engineering units as part of the display. Possible values for this field include "rpm", "degrees C", "%", etc.
You can add a degree symbol to the engineering units by holding down the Alt-key while typing 0176. Release the Alt-key before typing the C or F.

Analog Status Properties: Alarm Tab

The Alarm Tab contains two (optional) built-in alarms. One is for a high

set point and one for a low set point, based on the value of this tag. The initial priority for each is "None", preventing unconfigured alarms from being counted as disabled alarms.

Note: While the priority is set to "None," all other controls are disabled.
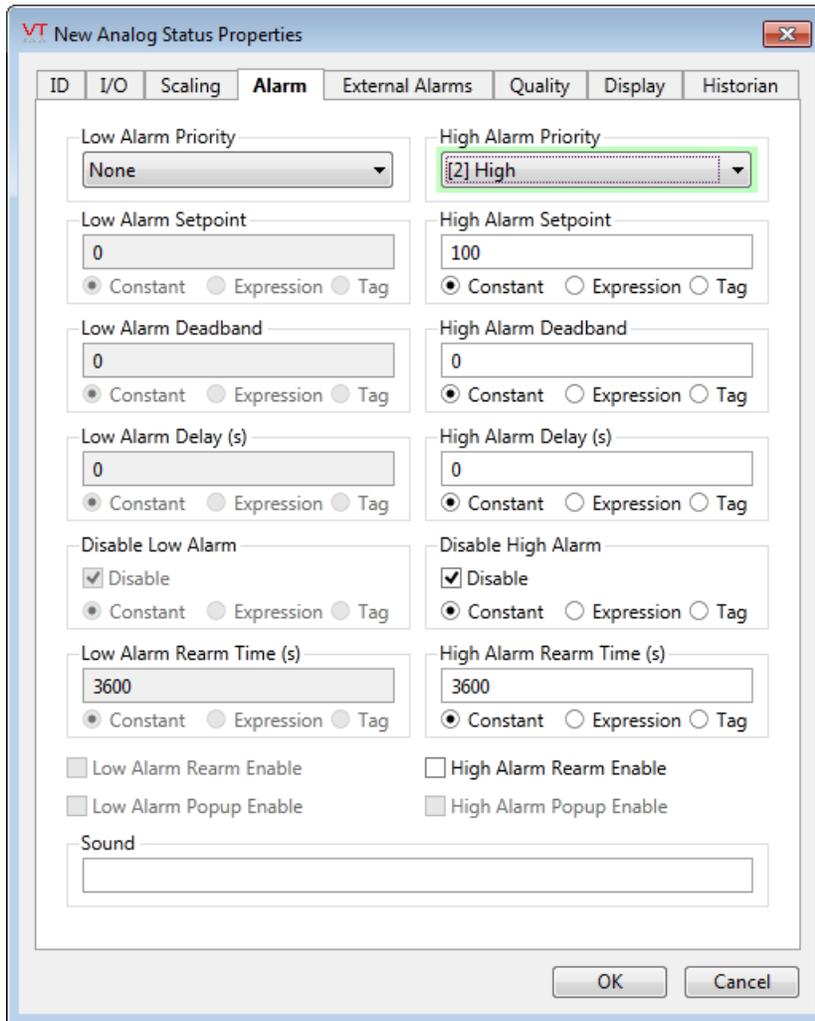
Low Alarm Priority

The Low Alarm Priority drop-down list enables you to select the priority of the low alarm that will be triggered for this tag if its value drops below the value defined for the Low Alarm Setpoint. The available priorities are:

- None
- 0 – Event
- 1 – Critical Alarm
- 2 – High Alarm

- 3 – Warning Alarm
- 4 – Notice Alarm

If you have defined your own Alarm Priority tags, those will also be available for selection.

Low Alarm Setpoint

The Low Alarm Setpoint field enables you to enter the lowest value that this tag's value will be permitted to attain, after which a low alarm will be triggered. The source for this field may be either a constant, the result of an expression, or the value of a tag.

Please see the topic: Constant, Expression or Tag for more information on selecting between these three choices.

Low Alarm Deadband

If noise in the system causes a value to fluctuate above and below the set point, the alarm will repeatedly switch between active and inactive status. By using the deadband, you can dampen out these fluctuations by setting an amount by which the tag's value must rise above the low alarm set point before it is considered to no longer be active. The deadband is used only for changing a current alarm's active status. It does not affect the set point.

Low Alarm Delay

Spikes in the value being monitored may cause transient alarms. By setting an alarm delay, you can specify that the trigger condition must remain true for a given length of time before the alarm is considered to have occurred. If required, refer to the topic: Constant, Expression or Tag for more information on selecting between these three choices.

Disable Low Alarm

The Disable Low Alarm field enables you to specify whether the low alarm for this tag is disabled or enabled. Disabling of alarms is typically used in situations where you wish to avoid false alarms. For example, in the event that routine maintenance is being performed on the equipment represented by this tag, or when you are aware that another interruption in communications will occur for a period of time; in such situations, the alarm can be disabled until the maintenance is complete and communications are reestablished. A value of 0 (or any tag or expression that returns 0) means that the alarm has not been disabled. Any value other than 0 in this field indicates that the alarm has been disabled.

The value accepted by the Disable Low Alarm field can be provided via any of a constant, an expression, or a tag. Please see the topic: Constant, Expression or Tag for help selecting which to use.

Alarm Rearm Time

Applies only if Alarm Rearm Enable is checked for this alarm. If the alarm has been acknowledged, but remains active for the time shown in this field, the alarm will return to the unacknowledged state. Audible and visible warnings configured for this alarm's priority level, will again be displayed. The Alarm Rearm Time is measured in seconds, and defaults to 3600 (1 hour). The value in this field must be greater than 0.

Alarm Rearm Enable

Controls whether this alarm will revert back to an unacknowledged state if it remains active for the length of time set in Alarm Rearm Time, after having been acknowledged by the operator.

High Alarm Setpoint, High Alarm Priority and Disable High Alarm

These work in exactly the same way as the low alarm set point, low alarm priority and disable low alarm fields, except that they are used to create an alarm that will be triggered when the tag's value goes above a given set point.

Low Alarm Pop-up Enable & High Alarm Pop-up Enable

If the configuration variable AlarmPopupsEnable is set to 1, then setting either the Low Alarm Pop-up Enable or the High Alarm Pop-up Enable, will result in a pop-up dialog being displayed whenever the respective alarm is triggered. It is strongly suggested that this feature be used sparingly.

Sound

The Sound field enables you to identify what sound will be played when this alarm is triggered. The Sound field can be set to blank, 0, 1, or to the name of a .WAV sound file to be played.

If the Sound field is set to 0, no sound will be played when this alarm is triggered.

If the Sound field is set to blank or 1, an alarm sound whose properties are configured on the associated Alarm Priority tag will be played.

If the Sound field identifies the name of a .WAV sound file, it will override any alarm sound configured for the associated Alarm Priority tag. When specifying a sound file, you must enter its name and extension (e.g.

MySound.wav). The specified sound file must be a .WAV file, and must be stored in the application directory. If the specified sound file is not found, the alarm will revert to using tones as specified in the associated Alarm Priority tag.

Analog Status Properties: External Alarms Tab
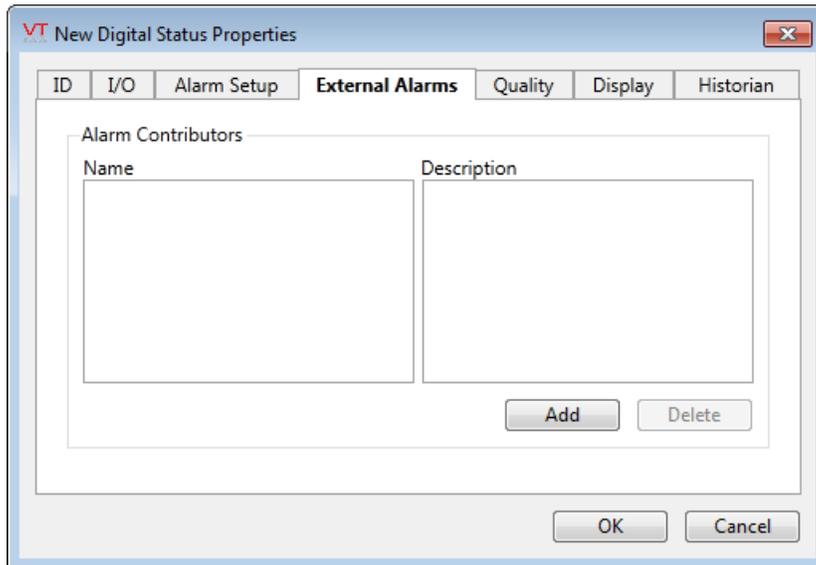
The External Alarms tab is used to display, add or remove separate Alarm tags that are triggered by this Analog Status tag. Use the Add button to create new external Alarm tags and the Delete button to remove them. The new alarm tags will be created as children of the current tag.

Analog Status Properties: Quality Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

The Quality Based On field can be used to indicate when there are communication or other problems. Quality is "good" when the linked tag has a value of zero and "bad" otherwise. If linked to the driver tag, which will have a non-zero error value when a problem is present, and if this tag is drawn using a widget that uses the System Style tag, then operators will see a visual notification of bad quality. Alternatively, you might use a Calculation tag to generate a "bad quality" warning based on an expression that you create.

The Quality Based On field can also be used when this tag is to be a contributor to a custom-built container tag. Please see the topic, Merit Tab and Quality Tab, for details.

This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Analog Status Properties: Display Tab

Style

> When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
>
> The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, which is the default style tag that is automatically part of every new VTScada application.

> *Related:*
> Style Settings Tags

Display Order

Used to specify the placement of this tag when viewed in an
associated station dialog. See Order Tab for further details.

Analog Status Properties: Historian Tab

Note: Read the label of the deadband setting on your tag! Values smaller than the deadband setting will not be written to the Historian.

Historian

If a Historian tag is selected, this tag's run-time values will be
saved for use in reports and the Historical Data Viewer. His-
torian configuration and advanced logging options are
described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag
after you have begun collecting data. If you switch to a new Historian
(perhaps for organizational or load sharing purposes), the data col-
lected for this tag by the previous Historian will become inaccessible.

Enable Logging

If values should be logged only while certain conditions are true, you can tie
the Enable Logging option to any tag or expression that will change from
zero (logging disabled) to non-zero (usually 1; logging enabled).

Deadband (Engineering Units)

The value of an Analog Status tag is automatically logged on change. The deadband range, specified in the defined engineering units, sets a minimum amount by which the tag's scaled data must change before a new value is written to the log file. This will dampen out minor changes of value, reducing the size of the log file by excluding system noise.
If not specified here, the deadband will be the difference between the high and low scale values (set on the Scaling tab), multiplied by the value of the application property, DefaultAnalogDeadbandFractionOfFullScale.
That property defaults to one quarter of one percent (0.25 on a range of 100).

> Note: No default will be applied to legacy applications. You may choose to set a value.

Analog Status Tag Widgets

The following widgets are available to display information about your application's Analog Status tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
Compass 1 Widget	Compass 2 Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Droplist Control Widget	Equipment / Status Color Indicator
Image Change Widget	Indicator Light Widget
LCD Bar Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget

Glass Gauge Widget	Classic Gauge Widget
Retro Gauge Widget	Cockpit Gauge Widget
Aqua Gauge Widget	Army Gauge Widget
Backlit Gauge Widget	Nautical Gauge Widget
Quarter Arc Gauge Widget	Linear Gauge Widget
Antique Gauge Widget	LCD Ring Widget
LCD Arc Widget	LCD Meter Widget
Momentary Button Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Rotary Control Widget (Knob)
Set Analog Value Widget	Set Value Button Widget
Set Value Hotbox Widget	Slider Widget
Text Change Widget	Top Bar Widget
Two Color Bar Widget	

Digital Input Tags

Counted towards your tag license limit.
Digital inputs represent incoming digital data values from equipment, and provide widgets that change color, appearance, or text to impart the value of the digital input and associated equipment.
A Digital Input tag can be a single bit digital input or a double bit digital input. For single bit digital inputs, there are two possible states. For digital inputs that use two address bits, there are four possible states. Digital inputs clamp raw data values to either zero or one on each input pin (yielding a result of 0, 1, 2, or 3 for pump status and Digital Status tags). Analog zero yields zero, while all other values yield 1. This is effective for analog values that have discrete states (e.g. line voltage can be read for on/off), but pure analog data (e.g. velocity) will only become false when the value drops below the sensitivity of the measuring device.

Digital Input Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Digital Input Properties: I/O Tab

The I/O Tab holds the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU). This is done by identifying the communication driver tag that communicates with the physical I/O device, the address at the physical I/O device from which this tag is to read its data, and the rate at which the I/O device should be scanned for data.

I/O Device

Select the communication driver tag from which data will be read.By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Bit 0 Address

The Bit 0 Address property is the address of the low order bit for this tag in the I/O device. If this is a single bit digital tag, this is the field that specifies the address of that bit.

Bit 1 Address

The Bit 1 Address property is the address of the high order bit for the digital input in the I/O device. If this is a single bit digital input, this field should be left blank.

Scan Interval

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Invert Input

The Invert Input check box enables you to specify whether or not the value of this tag should be reversed. If selected, the value for this tag will be inverted (i.e. 0 and 1 are swapped).

The typical use for input inversion is in projects where 1 indicates false or off, and 0 indicates true or on. By default, the Invert Input check box is not selected.

Enable Output

If checked, this input tag may also be used to write data to the specified Bit 0 address of the communication driver. Pulsed writes are not available. A Security Privilege may be set in the Merit tab to restrict access to this feature.

Single Bit and Double Bit Digital Inputs

A single bit digital status reads a value of 0 or a 1 from one address on an I/O device. An example of a single bit digital status would be a pump's status, where the pump is either off (0), or on (1).

A double bit digital status reads a value of 0 or 1 from two addresses on an I/O device. An example of an equipment process requiring a double bit digital status might be a valve with 4 states: open, closed, open or close action in progress, or error. The following table shows the possible values for a double bit digital status according to the values at each bit.

State	Bit 1	Bit 0	Description	Tag Value
State 0	0	0	A process is in action (either the valve is opening, or is closing)	0
State 1	0	1	The valve is closed.	1

State 2	1	0	The valve is open.	2
State 3	1	1	There is an error.	3
Invalid	Invalid	Invalid	No data.	Invalid

Digital Input Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag.

> Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create, using the VTScada scripting language.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual

data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is used to set the value of this contributor tag as the nth element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Digital Input Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

The Quality Based On field is for use when this tag is to be a contributor to a custom-built container tag. Please see the topic Merit Tab and Quality Tab, for details.

This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Digital Input Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
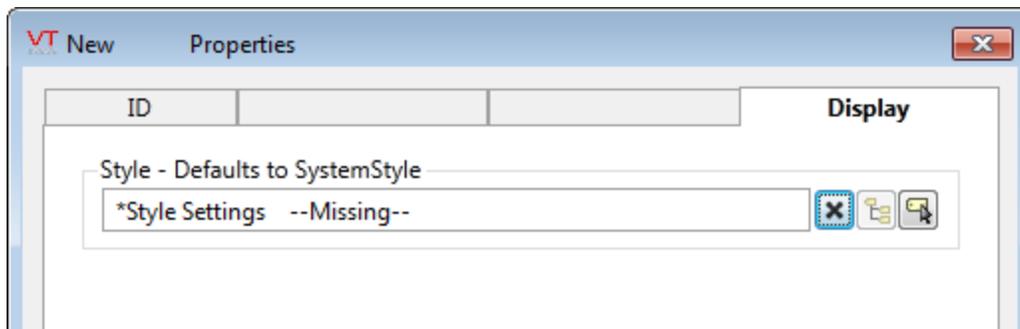
Related:

Style Settings Tags

Digital Input Properties: Alarm Tab
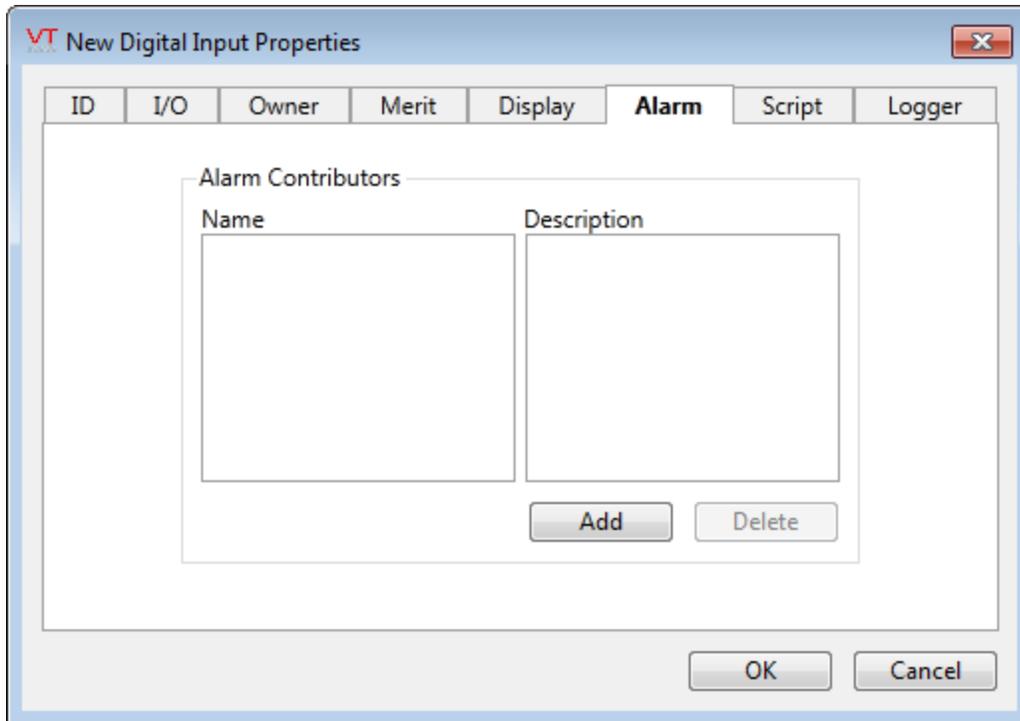
If you want this tag to trigger an alarm, use the Add button to open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Digital Input Properties: Script Tab

Optionally, link one or more Script tags to this tag. A Script tag provides a means of creating a procedure, using VTScada's programming language, that will run whenever this tag's value changes.

Digital Input Properties: Logger Tab

The logger tab enables you to associate a single Logger tag with this tag. The Logger tag and the attached Historian will record this tag's data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Note: Only one Logger tag can be directly associated with a single input or output tag. If you need to have multiple loggers with different logging rates, please refer to Using Function Tags to Create Multiple Data Logs of an I/O Tag

Digital Input Tag Widgets

The following widgets are available to display information about your application's Digital Input tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Momentary Button Widget

...Plot Data Widget

...Set Value Button Widget

...Set Value Hotbox Widget

...Text Change Widget

Digital Status Tags

Counted towards your tag license limit.
 Digital Status tags accept incoming digital data from an I/O device. A Digital Status tag might be used to read the status (e.g. off or on), or the mode (e.g. running or stopped) of equipment.

Note: VTScada also includes Pump Status tags, which are very similar to Digital Status tags. One difference to note is that there are reports that will work only with the data collected from a Pump Status tag, not a Digital Status tag.

A digital status can be a single bit digital input or a double bit digital input. For single bit digital inputs, there are two possible states (e.g. off or on, or running or stopped), whereas for digital inputs that use two

address bits, there are four possible states (e.g. opened, closed, process in action (either opening or closing), or error).

Digital inputs clamp raw data values to either zero or one on each input pin (yielding a result of 0, 1, 2, or 3 for pump status and Digital Status tags). Analog zero yields zero, while all other values yield 1. This is effective for analog values that have discrete states (e.g. line voltage can be read for on/off), but pure analog data (e.g. velocity) will only become false when the value drops below the sensitivity of the measuring device.

Digital Status Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might

also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Digital Status Properties: I/O Tab

The I/O Tab holds the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the polling driver responsible for determining the order and rate at which data polls will occur.

Note: Digital Status tags can have single or double bit addresses. Inform-
ation on single and double bit digital inputs can be found in Single Bit
and Double Bit Digital Inputs.

I/O Device

Select the communication driver tag from which data will be
read.By default, the tag will look for a parent tag that is a
device driver (*Driver). If none is found, the text "--Missing--"
will be displayed. The tag button to the right of the field opens
the tag browser, from which you can either select an existing
communication driver tag or add a new one. The X button will
clear the field. Right-clicking on a tag in the field will open a
dialog with which you can add or remove a Snapshot Expres-
sion, or open a selected driver's properties dialog.

Bit 0 Address

The Bit 0 Address property is the address of the low order bit for this tag in the I/O device. If this is a single bit digital tag, this is the field that specifies the address of that bit. The reported value of this digital tag is the data at this address plus two times the data at the Bit 1.

Note: If a data logger is being used on a tag, it can only be for a single bit address.

Bit 1 Address

The Bit 1 Address property is the address of the high order bit for the digital input in the I/O device. If this is a single bit digital input, this field should be left blank.

If the option, Read History is selected, this will become the History Address.

Scan Rate

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Read History

If checked, the field Bit 1 Address becomes History Address and the field History Scan Interval is enabled.

History Address

If set, then the Bit 0 address becomes optional. This field provides a means of reading values as recorded by a data logger.

If both the Bit 0 address and the history address are provided, then the bit 0 address will be polled for data at the interval set

by the Scan Rate, but the values read from the data logger will overwrite the values logged by this tag when it is updated.
The form for the history address will depend on the RTU. Please refer to the addressing reference chapter for your driver.
Note that it is not possible to enter a parameter expression directly in the History Address field. As an alternative, you may enter compound address in the Bit 0 field using a parameter expression. For example, if Bit0Addr, HistAddr and HistoryRate are all properties of a parent tag, the following expression in the Bit 0 field will create a compound address with all three components:
concat(.\Bit0Addr , "):(" , .\HistAddr , "):(" , .\HistoryRate)
The characters "):(" separate each component of the address. Do not add a leading or trailing parenthesis around the entire expression.

History Scan Interval

Sets the time between polls of the history data logger. If a History address has been set, but this field is left blank, then it will default to the value set in the Scan Rate field. The recommended value for this field is 60 seconds. Lesser values may cause communications to slow down.
Note that it is not possible to enter a parameter expression directly in the History Scan Interval field.

Manual Data

Optionally provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new Analog Input tag when application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark.

Off Text

Optionally configure text that will be displayed in some of the tag's widgets when the value of this tag is 0 or false/off.

On Text

Optionally configure text that will be displayed in some of the tag's widgets when the value of the tag is 1 or true/on.

Invert Input

If the Invert Input check box is selected, the value of the tag being used as the data source is reversed (i.e. a 1 becomes a 0 or a 0 becomes a 1) before it is written to the I/O device.

Enable Output

If checked, this input tag may also be used to write data to the specified Bit 0 address of the communication driver. Pulsed writes are not available. A Security Privilege may be set in the Quality tab to restrict access to this feature.

Digital Status Properties: Alarm Setup Tab

Optionally configure one alarm based on the status of this tag. Note that while the priority is set to "None" all other controls are disabled.
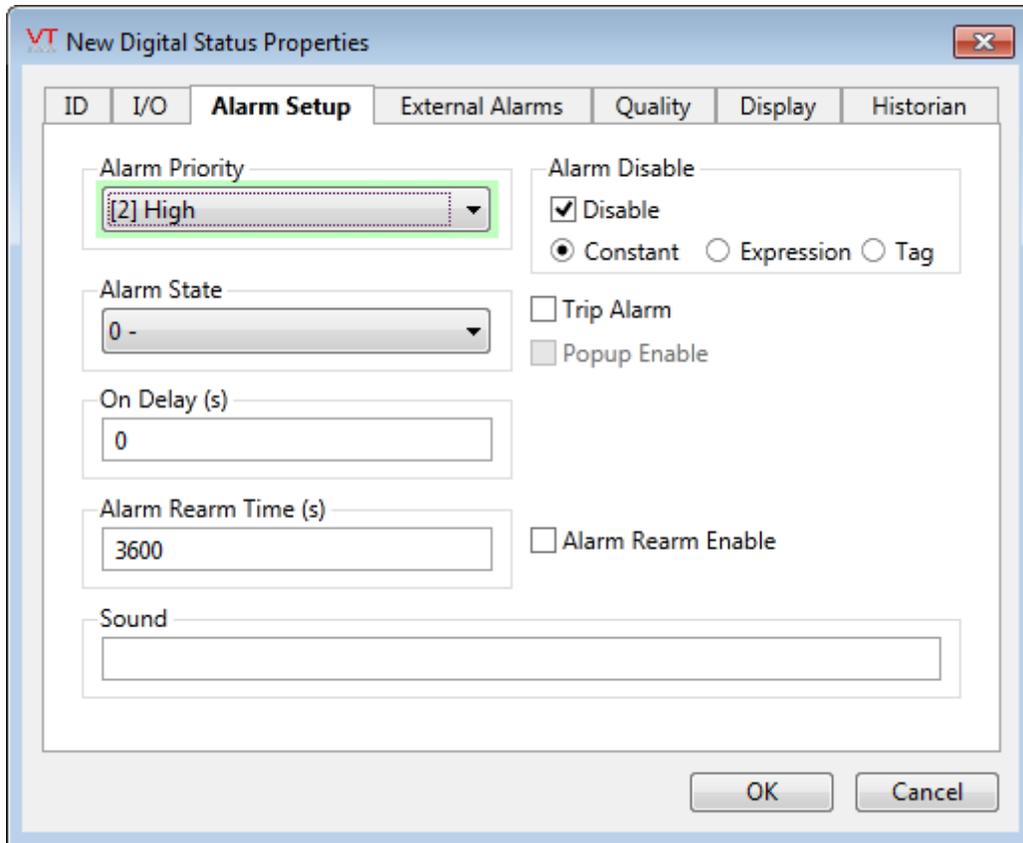
Alarm Priority

Select the priority of the alarm that will be triggered for this tag. The available priorities are:

- None (all other controls will be disabled)
- 0 – Event
- 1 – Critical Alarm
- 2 – High Alarm
- 3 – Warning Alarm
- 4 – Notice Alarm

If you have defined your own Alarm Priority tags, those will also be available for selection.

Alarm State

Select the trigger for the alarm. The alarm will be triggered

when the tag's state becomes one of the following:

0 The alarm is triggered when Bit 0 goes low. The Bit 1 address either has not been provided or is low.

1 The alarm is triggered when Bit 0 goes high. The Bit 1 address either has not been provided or is low.

2 Two addresses provided. Bit 0 goes low and Bit 1 goes high.

3 Two addresses provided. Both Bit 0 and Bit 1 go high.

Upon Change – any change of state triggers an alarm. Note that when the Upon Change option is selected, the trip alarm option will be set and cannot be deselected by the developer.

Alarm Delay

Transient changes in state can result in multiple alarms being triggered. By setting an alarm delay, you can specify that if the tag's state changes back one which does not constitute an alarm situation, it must remain in that state for a given length of time before the alarm is deemed to be inactive.

Alarm Rearm Time

Applies only if Alarm Rearm Enable is checked for this alarm. If the alarm has been acknowledged, but remains active for the time shown in this field, the alarm will return to the unac-knowledged state. Audible and visible warnings configured for this alarm's priority level, will again be displayed. The Alarm Rearm Time is measured in seconds, and defaults to 3600 (1 hour). The value in this field must be greater than 0.

Alarm Rearm Enable

Controls whether this alarm will revert back to an unac-knowledged state if it remains active for the length of time set in Alarm Rearm Time, after having been acknowledged by the

operator.

Disable Alarm

Specify whether the alarm for this tag is disabled or enabled. Disabling of alarms is typically used in situations where you wish to avoid false alarms. For example, in the event that routine maintenance is being performed on the equipment rep-resented by this tag, or when you are aware that another inter-ruption in communications will occur for a period of time. In such situations, the alarm can be disabled until the main-tenance is complete and communications are re-established. May be linked to a tag or expression for suppression by design.

Trip Alarm

Select if this alarm is to be a trip alarm.
Trip alarms do not have the attribute of being Active or Inact-ive based on the value of the Trigger tag. Once triggered, the alarm must be acknowledged by an operator, but the under-lying cause of the alarm need not be cleared.

Append State to Alarm Description

When selected, VTScada will append the text used for the state of the tag, as taken from the Off Text and On Text fields in the tag's I/O tab, to the alarm message.

Pop-up Enable

If the configuration variable AlarmPopupsEnable is set to 1, then setting either the Low Alarm Pop-up Enable or the High Alarm Pop-up Enable, will result in a pop-up dialog being dis-played whenever the respective alarm is triggered. It is

strongly suggested that this feature be used sparingly.

Digital Status Properties: External Alarms Tab

The External Alarms tab is used to display, add or remove separate Alarm tags that are triggered by this Digital Status tag. Use the Add button to create new external Alarm tags and the Delete button to remove them. The new alarm tags will be created as children of the current tag.

Digital Status Properties: Quality Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

The Quality Based On field can be used to indicate when there are communication or other problems. Quality is "good" when the linked tag has a value of zero and "bad" otherwise. If linked to the driver tag, which will have a non-zero error value when a problem is present, and if this tag is drawn using a widget that uses the System Style tag, then operators will

see a visual notification of bad quality. Alternatively, you might use a Cal-
culation tag to generate a "bad quality" warning based on an expression
that you create.

The Quality Based On field can also be used when this tag is to be a con-
tributor to a custom-built container tag. Please see the topic Merit Tab
and Quality Tab, for details.

Digital Status Properties: Display Tab

Style

When this tag is represented on screen by widgets that are
able to use a Style Settings tag, you can save development time
by choosing the Style Settings tag that holds the correct dis-
play configuration for this tag instance.
The default configuration will search for the first parent tag of
the Style Settings type. If there is none, (message reads, "--

Missing--") it will use System Style, which is the default style tag that is automatically part of every new VTScada application.

Related:
Style Settings Tags

Off Text and On Text

Certain widgets will display the text you configure here. This text will also be used in confirmation boxes for the control action (if configured).

Display Order

Used to specify the placement of this tag when viewed in an associated station dialog. See Order Tab for further details.

Digital Status Properties: Historian Tab

HistorianIf a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Enable Logging

If values should be logged only while certain conditions are true, you can tie the Enable Logging option to any tag or expression that will change from zero (logging disabled) to non-zero (usually 1; logging enabled).

Digital Status Tag Widgets

The following widgets are available to display information about your application's Digital Status tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Draw Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Momentary Button Widget

...Plot Data Widget

...Set Value Button Widget

...Set Value Hotbox Widget

...Text Change Widget

...Two Color Bar Widget

Pulse Input Tags

Counted towards your tag license limit.
Pulse input tags accept incoming analog data from a pulse input device (such as a rain gauge, transducer, or a gas, water, or electric meter).

Note: The Pulse Input Tag was designed to work with the Data Flow driver and will not permit the selection of any other driver type.

Typically, the value sent is the number of tips of a rain collection device since the last poll. Use the scaling parameters of this tag to assign the relation between each tip of the device and the amount of rain measured. After each poll, the driver will re-set to zero. Note: the driver provides no support for storing the last value sent. If a communication error prevents a value from being sent on any given poll interval, that value is lost.

Values from the pulse input can be displayed using the Totalizer widget, which includes a configuration option to reset the value back to zero every N hours. There is also a report – The Rainfall Report – designed for use with the Pulse Input tag.

It is not uncommon to want to trigger an alarm when the rainfall rate exceeds a certain amount. For that purpose, you should use a History Statistics tag's Totalizer option as the trigger for the alarm. The Data Period parameters of the History Statistics tag should be set short enough to give a frequent-enough update but remain in the Update Interval long enough so as not to overload the system. For every Update Interval, the history of the tag must to be accessed from disk. Suggested starting values are to use the latest 1 hour with an update rate of 5 minutes. An alarm can be attached to the History Statistics tag to check if the total rainfall per hour (assuming that the "Latest" parameter is set to 1 hour) exceeds a limit.

Pulse Input Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.
There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Pulse Input Properties: I/O Tab

The I/O Tab holds the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the polling driver responsible for determining the order and rate at which data polls will occur.

I/O Device

A Data Flow Driver required for this tag to function as a pulse input. If a different type of driver is chosen, this tag will approximate the behavior of an Analog Input.

Address

Provide the address within the I/O device from which this tag is to read data. This value must match the configuration of your PLC or RTU hardware. Refer to the Addressing topic for your particular device driver for guidance.For some drivers (SNMP and the OPC Client) an address browser is provided to assist you.

Scan Interval

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Engineering Units

Provide units of measure that the input data represents. Possible values for this field include "rpm" "degrees C", "%", etc

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Pulse Input Properties: Scaling Tab
The Scaling Tab requires a scaled value that will be counted per

incoming pulse.

Pulse Input Properties: Quality Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Quality Based On field, for use when this tag is to be a contributor to a custom-built container tag. Please see the topic Merit Tab and Quality Tab, for details.

Pulse Input Properties: Order Tab

The Order Tab for a Pump Status tag enables you to specify the placement of this tag when viewed in an associated station dialog.

Pulse Input Properties: Historian Tab

HistorianIf a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Pulse Input Tag Widgets

The following widgets are available to display information about your application's Pulse Input tags:

...Alarm Priority Box

...Alarm Priority Icon

...Draw Widget

...Totalizer Widget

Pump Status Tags

Counted towards your tag license limit.

Pump status tags accept incoming digital data from pumps. These are similar to Digital Status tags, but are designed specifically for use with pumps. predefined reports, including Pump Activity and Pump Discrepancy are designed to work with data from Pump Status tags. Digital inputs, including Pump Status tags, clamp raw data values to either zero or one on each input pin (yielding a result of 0, 1, 2, or 3 for pump status and Digital Status tags). Analog zero yields zero, while all other values yield 1. This is effective for analog values that have discrete states (e.g. line voltage can be read for on/off), but pure analog data (e.g. velocity) will only become false when the value drops below the sensitivity of the measuring device.

Pump Status Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Pump Status Properties: I/O Tab

The I/O Tab holds the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the polling driver responsible for determining the order and rate at which data polls will occur.
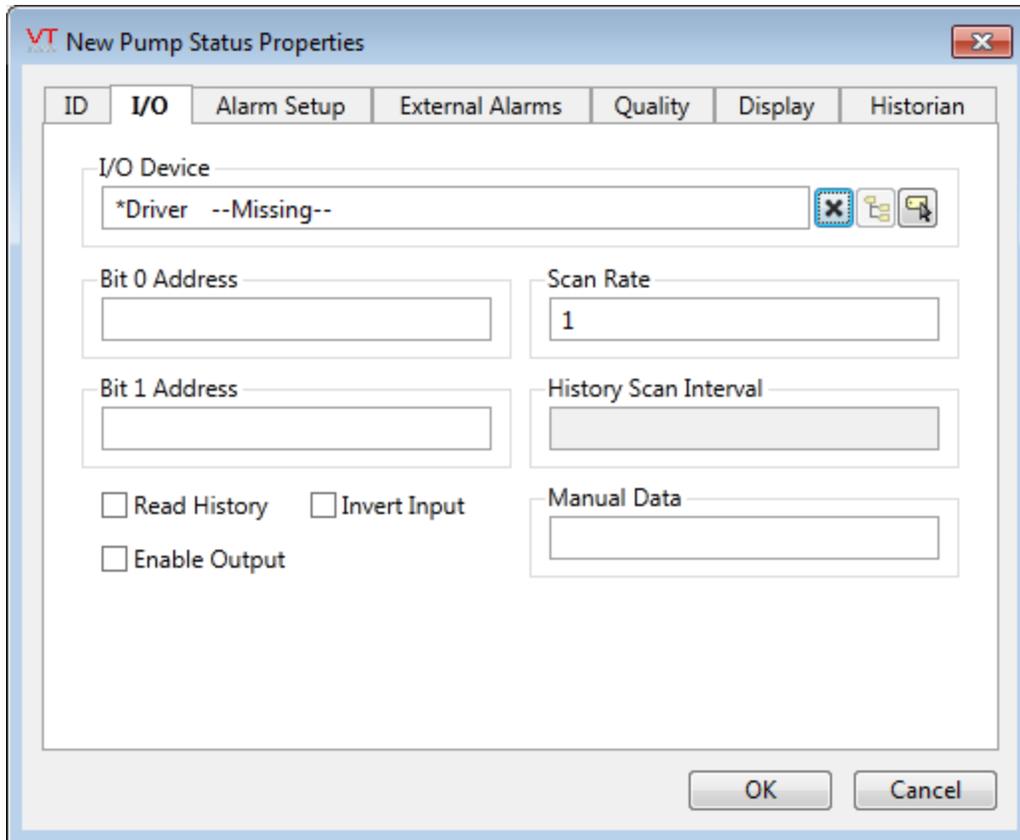
Pump status tags can have single or double bit addresses.

I/O Device

Select the communication driver tag from which data will be read.By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Bit 0 Address

The Bit 0 Address property is the address of the low order bit

for this tag in the I/O device. If this is a single bit digital tag, this is the field that specifies the address of that bit. The reported value of this digital tag is the data at this address plus two times the data at the Bit 1.
Note: If a data logger is being used on a tag, it can only be for a single bit address.

Bit 1 Address

The Bit 1 Address property is the address of the high order bit for the digital input in the I/O device. If this is a single bit digital input, this field should be left blank.

Scan Rate

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Read History

If checked, the field Bit 0 Address becomes History Address and the field History Scan Interval is enabled.

History Address

If set, then the Bit 0 address becomes optional. This field provides a means of reading values as recorded by a data logger.
If both the Bit 0 address and the history address are provided, then the Bit 0 address will be polled for data at the interval set by the Scan Rate, but the values read from the data logger will overwrite the values logged by this tag when it is updated.

The form for the history address will depend on the RTU. Please refer to the addressing reference chapter for your driver.

History Scan Interval

Sets the time between polls of the history data logger. If a History address has been set, but this field is left blank, then it will default to the value set in the Scan Rate field. The recommended value for this field is 60 seconds. Lesser values may cause communications to slow down.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Invert Input

If the Invert Input check box is selected, the value of the tag being used as the data source is reversed (i.e. a 1 becomes a 0 or a 0 becomes a 1) before it is written to the I/O device.

Enable Output

If checked, this input tag may also be used to write data to the specified Bit 0 address of the communication driver. A Security Privilege may be set in the Quality tab to restrict access to this feature.

Pump Status Properties: Alarm Setup Tab
Optionally configure one alarm based on the status of this tag. Note that,

while the priority is set to "None," all other controls are disabled.

Alarm Priority

The Alarm Priority drop-down list enables you to select the priority of the alarm that will be triggered for this tag. The available priorities are:

- None
- 0 – Event
- 1 – Critical Alarm
- 2 – High Alarm
- 3 – Warning Alarm
- 4 – Notice Alarm

If you have defined your own Alarm Priority tags, those will also be available for selection.

Alarm State

The Alarm State spin box enables you to select the trigger for the alarm. The alarm will be triggered when the tag's state becomes one of the following:

0 The alarm is triggered when Bit 0 goes low. The Bit 1 address either has not been provided or is low.

1 The alarm is triggered when Bit 0 goes high. The Bit 1 address either has not been provided or is low.

2 Two addresses provided. Bit 0 goes low and Bit 1 goes high.

3 Two addresses provided. Both Bit 0 and Bit 1 go high.

Upon Change – any change of state triggers an alarm. Note that when the Upon Change option is selected, the trip alarm option will be set and cannot be deselected by the developer.

Alarm Delay

Transient changes in state can result in multiple alarms being triggered. By setting an alarm delay, you can specify that if the tag's state changes back one which does not constitute an alarm situation, it must remain in that state for a given length of time before the alarm is deemed to be inactive.

Alarm Rearm Time

Applies only if Alarm Rearm Enable is checked for this alarm. If the alarm has been acknowledged, but remains active for the time shown in this field, the alarm will return to the unacknowledged state. Audible and visible warnings configured for this alarm's priority level, will again be displayed. The Alarm Rearm Time is measured in seconds, and defaults to 3600 (1 hour). The value in this field must be greater than 0.

Alarm Rearm Enable

Controls whether this alarm will revert back to an

unacknowledged state if it remains active for the length of time set in Alarm Rearm Time, after having been acknowledged by the operator.

Disable Alarm

Specify whether the low alarm for this tag is disabled or enabled. Disabling of alarms is typically used in situations where you wish to avoid false alarms. For example, in the event that routine maintenance is being performed on the equipment represented by this tag, or when you are aware that another interruption in communications will occur for a period of time; in such situations, the alarm can be disabled until the maintenance is complete and communications are reestablished.
May be linked to a tag or expression for suppression by design.

Trip Alarm

Select if this alarm is to be a trip alarm.
Trip alarms differ from level alarms in that they are not active or inactive based on the value of the Trigger tag. Once triggered, they must be acknowledged, but there is no requirement that the cause of the alarm condition be cleared.

Append State to Alarm Description

When selected, VTScada will append the text used for the state of the tag, as taken from the Off Text and On Text fields in the tag's I/O tab, to the alarm message.

Pop-up Enable

If the configuration variable AlarmPopupsEnable is set to 1,

then setting either the Low Alarm Pop-up Enable or the High Alarm Pop-up Enable, will result in a pop-up dialog being displayed whenever the respective alarm is triggered. It is strongly suggested that this feature be used sparingly.

Pump Status Properties: External Alarms Tab

The External Alarms tab is used to display, add or remove separate Alarm tags that are triggered by this Pump Status tag. Use the Add button to create new external Alarm tags and the Delete button to remove them. The new alarm tags will be created as children of the current tag.

Pump Status Properties: Quality Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

The Quality Based On field is for use when this tag is to be a contributor to a custom-built container tag. Please see the topic Merit Tab and Quality Tab, for details.

This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Pump Status Properties: Display Tab

Style

> When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
>
> The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--

Missing--") it will use System Style, which is the default style tag that is automatically part of every new VTScada application.

Related:
Style Settings Tags

Off Text and On Text

Certain widgets will display the text you configure here. This text will also be used in confirmation boxes for the control action (if configured).

Display Order

Used to specify the placement of this tag when viewed in an associated station dialog. See Order Tab for further details.

Pump Status Properties: Historian Tab

HistorianIf a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Enable LoggingIf values should be logged only while certain conditions are true, you can tie the Enable Logging option to any tag or expression that will change from zero (logging disabled) to non-zero (usually 1; logging enabled).

Pump Status Tag Widgets

The following widgets are available to display information about your application's Pump Status tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Color Blink Widget
Color Box Widget	Color Line Widget
Draw Widget	Indicator Light Widget
Image Change Widget	Momentary Button Widget
Plot Data Widget	Set Value Button Widget
Set Value Hotbox Widget	Text Change Widget
Two Color Bar Widget	Equipment / Status Color Indicator

String I/O Tags

String I/O tags are used to read or write text values. They are designed to be simple to configure and straight-forward to use.

Retained or Persisted values

Analog Control and Analog Output tags will both retain their last written value across application restarts. Use these tags when there is a need for a value to persist.

String I/O tags with output enabled, and with no driver ("No tag selected" rather than the default, "[*Driver] None found") will retain their last written value across application restarts.

The Digital Control and Digital Output tags do not have a similar feature, but an analog tag can be used in their place, writing a zero or a one. If the intent is to write a clear zero or one with an analog tag, both the scaled and unscaled range should be adjusted for a minimum value of

zero and a maximum value of one, so that scaling does not adjust the values.

String I/O Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags

(note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added.

To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

String I/O Properties: I/O Tab

Use this tab to specify the device driver and the address where values are to be read or written.

If output strings are to have a minimum length, you can choose to add spaces, further specifying whether those spaces should be added at the beginning, the end, or equally to either side of the string.
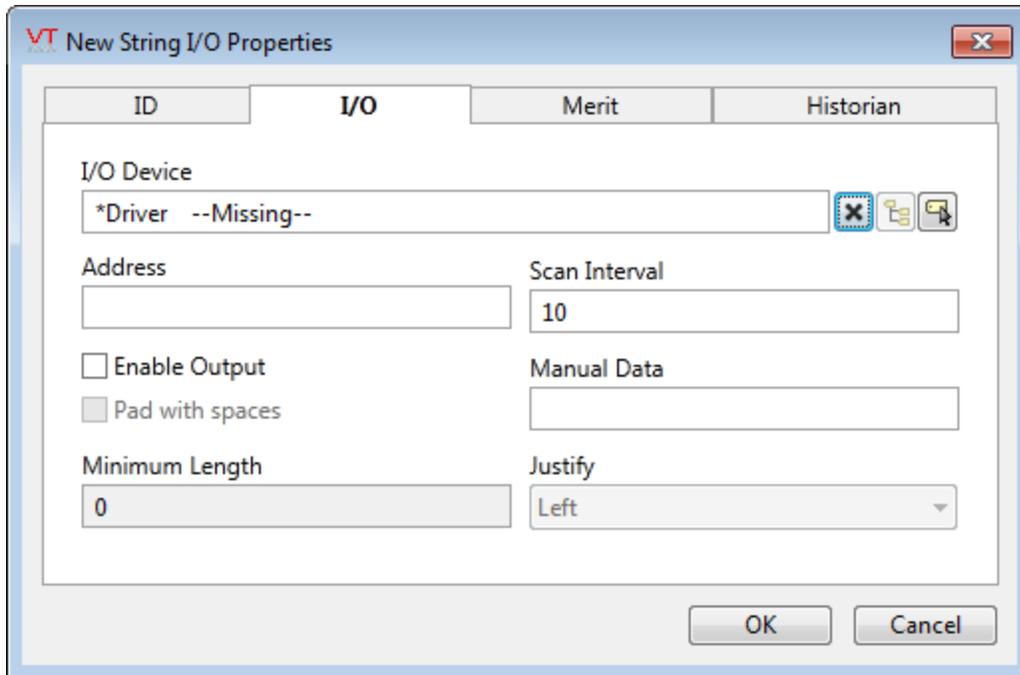
I/O Device

Select the communication driver tag from which data will be read.By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Scan Interval

Provide the frequency, measured in seconds, that the I/O device should be scanned for new data. If the I/O Device is a Polling driver, which provides its own scan interval, then this field will not accept data.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Enable Output

If checked, this input tag may also be used to write data to the specified address of the communication driver. A Security Privilege may be set in the Quality tab to restrict access to this feature.
Values that are to be written may be padded with blank spaces to achieve a minimum length

Pad With Spaces

Enabled only when output is enabled. Selecting this box is the first step in specifying a minimum number of characters to be sent in each write operation.

Minimum Length

Enabled only when Pad With Spaces has been selected. Use this to specify the minimum number of characters to be sent in each write operation.

Justify

Enabled only when Pad With Spaces has been selected. Extra spaces, added to meet the minimum string length, may be added at the beginning of the string (Right Justify), ath the end of the string (Left Justify) or equally at either end of the string (Center).

String I/O Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

String I/O Properties: Historian Tab

Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Enable Logging

If values should be logged only while certain conditions are true, you can tie the Enable Logging option to any tag or expression that will change from zero (logging disabled) to non-zero (usually 1; logging enabled).

String I/O Tag Widgets

The following widgets are available to display information about, or allow data entry for, your application's String I/O tags:

...Draw Text Widget

...Set Analog Value Widget

...String Droplist Widget

Output Tags

Output tags are used to write data to an I/O device, such as a programmable logic controller (PLC), remote terminal unit (RTU), or an I/O board. These tags permit the handling of either digital (discrete), or analog (continuous floating point) values.

> Note: Output tags will not write INVALID data. (INVALID is a specific VTScada data type, roughly meaning, "no valid value available"). Also, with the exception of the MultiWrite tag, no output type will write an unchanged value twice unless directed by the driver performing a "rewrite outputs" operation.

Record Operator-Entered Values

While the output tags are designed to write to a device, they can also be used to record purely operator-entered values. For example, an operator may be asked to read a value from a device that is not connected to the automation system and record that value in a place where it can be included in the SCADA system's reports.

To record operator-entered values:

1. Create an Analog Control tag.
   The name and description should match the information that is to be recorded.
   Do not configure an I/O device or address.
   Ensure that the scaled minimum and maximum values allow the full range of expected numeric values. (The Operator Notes page exists to record text.)

2. Add a Logger tag to the Analog Control.
   Check the option, Log on Change as well as Time.
   Set the Log Range Interval to a very large value (for example, 3153600000 == 100 years). This ensures that a value will be written to the Historian only upon operator entry.

3. Draw the Analog Control tag on a page as a Numeric Entry widget, having a relevant label. Operators should note that it is necessary to press enter, or tab away from the data entry field before the value will be saved.

Each time the operator enters a new value, it will be written to the Historian, and may be used in trend graphs and reports. Note that a click on any widget linked to an output tag does not open the trend window.
The following output tags are included with VTS:

Analog Control Tags	Analog Output Tags
Deadband Control Tags	Digital Control Tags
Digital Output Tags	Multi-Write Tags
Selector Switch Tags	Trigger Tags
String I/O Tags	

Related Information:

...Communication Driver Tags – Driver tag reference

...Working With Tags – The Tag Browser – Using the Tag Browser to create and edit tags.

Confirmation Prompts for Output Tags

Many of the widgets for output tags include an option to require the operator to confirm that they intended to write a value. For analog tags, the confirmation text will include the new value. For example, "Are you sure you want to set the value to 1500?".
For some digital tags, the message displayed will be more or less meaningful depending on how you configure the tag. For example, the Digital Control tag includes the fields "On Text" and "Off Text". Selector Switches provide a label for each position. If you provide a descriptive word for the control action matching each output state ("On", "Open", "Start"), then that word will appear in the confirmation prompt. For tags that do not have this option, and for those with the option that you leave unconfigured, a generic message will be displayed.

| Generic confirmation prompt | The prompt from the same control with "On Text" and "Off Text" configured. |
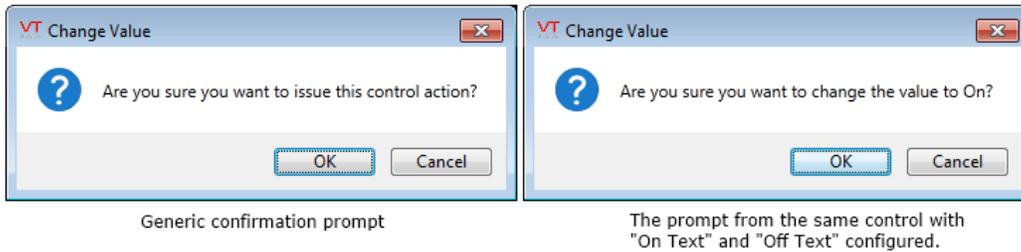
Analog Control Tags

Counted towards your tag license limit.

It is the role of Analog Control tags to transmit analog data entered by the user to an I/O device. An Analog Control tag might be used to change the speed of a motor, or to specify a feed time for chemicals to be added to a well or tank.

In most cases, an operator will use the control tag to write information out to the system. You also have the option of configuring an automatic data source such as a Calculation tag to provide the information that is to be written.

Retained or Persisted values

Analog Control and Analog Output tags will both retain their last written value across application restarts. Use these tags when there is a need for a value to persist.

String I/O tags with output enabled, and with no driver ("No tag selected" rather than the default, "[*Driver] None found") will retain their last written value across application restarts.

The Digital Control and Digital Output tags do not have a similar feature, but an analog tag can be used in their place, writing a zero or a one. If the intent is to write a clear zero or one with an analog tag, both the scaled and unscaled range should be adjusted for a minimum value of zero and a maximum value of one, so that scaling does not adjust the values.

Record Operator-Entered Values:

While the output tags are designed to write to a device, they can also be used to record purely operator-entered values. For example, an operator

may be asked to read a value from a device that is not connected to the automation system and record that value in a place where it can be included in the SCADA system's reports.

To record operator-entered values:

1. Create an Analog Control tag.
   The name and description should match the information that is to be recorded.
   Do not configure an I/O device or address.
   Ensure that the scaled minimum and maximum values allow the full range of expected numeric values. (The Operator Notes page exists to record text.)

2. Add a Logger tag to the Analog Control.
   Check the option, Log on Change as well as Time.
   Set the Log Range Interval to a very large value (for example, 3153600000 == 100 years). This ensures that a value will be written to the Historian only upon operator entry.

3. Draw the Analog Control tag on a page as a Numeric Entry widget, having a relevant label. Operators should note that it is necessary to press enter, or tab away from the data entry field before the value will be saved.

Each time the operator enters a new value, it will be written to the Historian, and may be used in trend graphs and reports. Note that a click on any widget linked to an output tag does not open the trend window.

Analog Control Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.
There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.
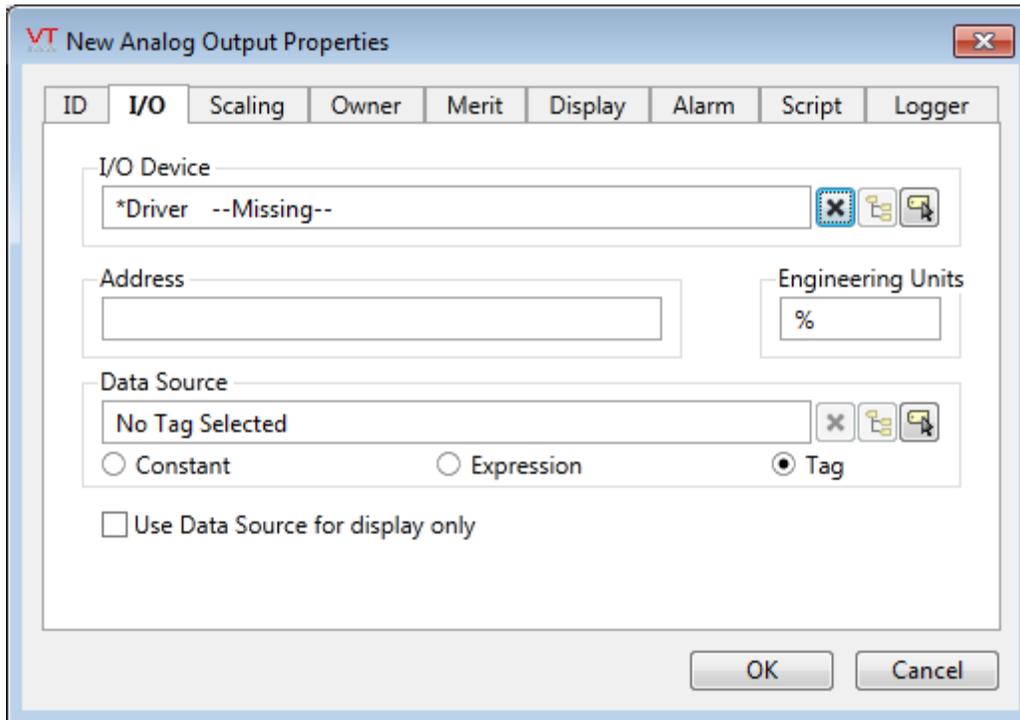
Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Analog Control Properties: I/O Tab

In the I/O Tab, you will find the properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the polling driver responsible for determining the order and rate at which data polls will occur.

I/O Device

Select the communication driver tag from which data will be read. By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing

communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Address

Provide the address within the I/O device to which this tag is to write data. This value must match the configuration of your PLC or RTU hardware. Refer to the Addressing topic for your particular device driver for guidance.For some drivers (SNMP and the OPC Client) an address browser is provided to assist you.

Engineering Units

Provide units of measure that the input data represents. Possible values for this field include "rpm" "degrees C", "%", etc

Data Source

[Optional] While Analog Control tags usually take their value from an operator-controlled drawing object, the Data Source field provides the option of configuring this tag to accept a value from another tag or expression instead.

The value from the Data Source can be written to the I/O device, or it can be used to simply update the widget display.

If you have selected a data source that is configured to write output, and have also drawn the tag as an object that operators can use to set values, then both can write to the I/O device. The value written will be whichever was most recently changed.

Write output when Data Source changes

When selected, the value supplied by the data source tag or expression will be written to the I/O device and will also be used to set this tag's value. Selecting this option automatically de-selects the "Use Data Source for display only" option.

Use Data Source for display only

When selected, the value supplied by the data source tag or expression will not be written to the I/O device, but will be used to set this tag's value. Selecting this option automatically de-selects the "Write output when Data Source changes" option.

Analog Control Properties: Scaling Tab
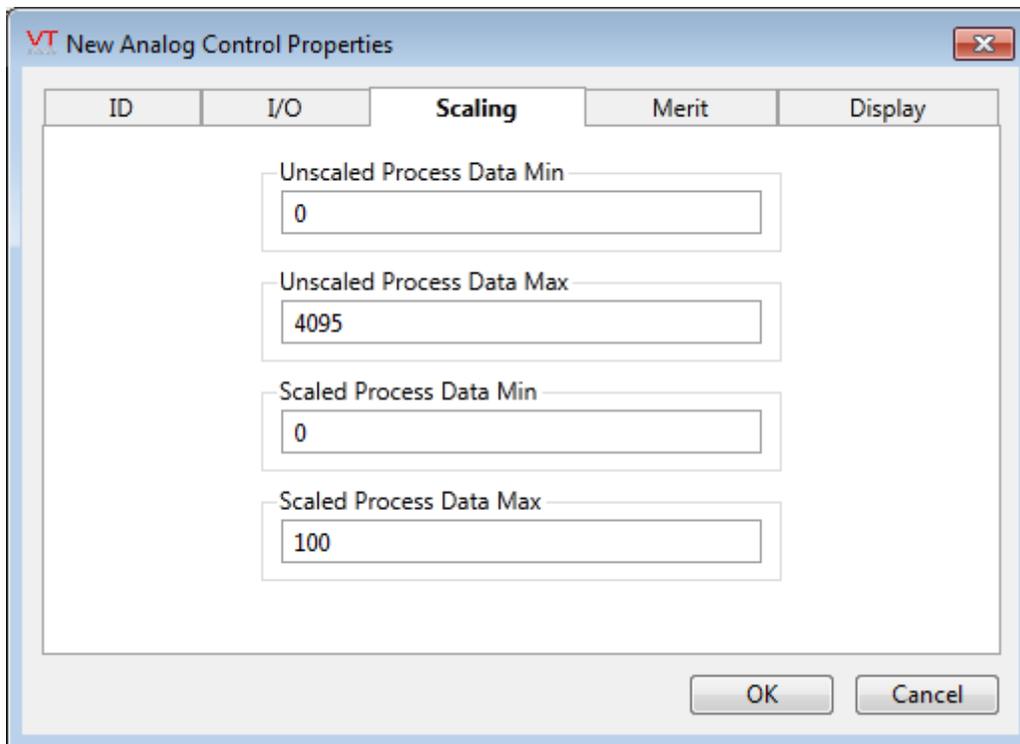
The Scaling Tab is used to convert scaled data (intelligible to operators) into unscaled, raw data for the I/O device. VTScada uses an algorithm to calculate this tag's value internally, based on the known unscaled and scaled values you supply.

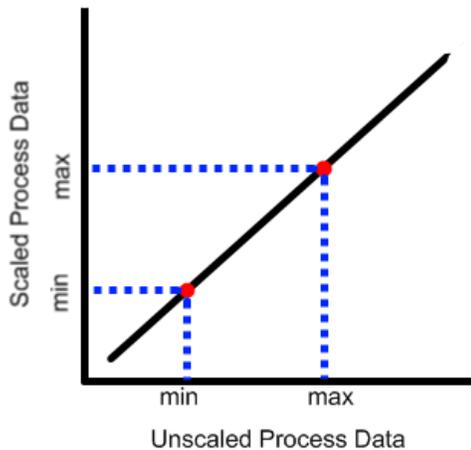The default value for each of these fields is stored in an application property. You are advised to change those properties to match the values used by your hardware by using the Edit Properties page of the Application Configuration dialog.

You will need to use the Advanced mode of the properties page. The properties to copy and edit are:

- AnalogUnscaledMinDefault
- AnalogUnscaledMaxDefault
- AnalogScaledMinDefault
- AnalogScaledMaxDefault

Assuming a linear relationship between the values, by providing a known minimum scaled value for a given minimum unscaled value, and a known maximum scaled value for a given

maximum unscaled value, VTScada can provide a scaled value for any given unscaled input value.

Analog Control Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Analog Control Properties: Display Tab

Style

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "-- Missing--") it will use System Style, which is the default style tag that is automatically part of every new VTScada application.

Related:
Style Settings Tags

Display Order

Used to specify the placement of this tag when viewed in an associated station dialog. See Order Tab for further details.

Analog Control Tag Widgets

The following widgets are available to Analog Control tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Comm Line Widget
Compass 1 Widget	Compass 2 Widget
Draw Text Widget	Droplist Control Widget
	Image Change Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Momentary Button Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Set Analog Value Widget
Set Value Button Widget	Set Value Hotbox Widget
Slider Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget
Rotary Control Widget (Knob)	Equipment / Status Color Indicator
Indicator Light Widget	LCD 5x7 Matrix Widget
LCD 7 Segment Widget	

Related Information:

Confirmation Prompts for Output Tags – Control options for confirmation prompt text.

Analog Output Tags

Counted towards your tag license limit.
It is the role of analog outputs to take analog data from the user or from another tag, and write it to an I/O device. Analog data can be transmitted by an operator via a slider or a numeric entry field.

Retained or Persisted values

Analog Control and Analog Output tags will both retain their last written value across application restarts. Use these tags when there is a need for a value to persist.

String I/O tags with output enabled, and with no driver ("No tag selected" rather than the default, "[*Driver] None found") will retain their last written value across application restarts.

The Digital Control and Digital Output tags do not have a similar feature, but an analog tag can be used in their place, writing a zero or a one. If the intent is to write a clear zero or one with an analog tag, both the scaled and unscaled range should be adjusted for a minimum value of zero and a maximum value of one, so that scaling does not adjust the values.

Analog Output Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you

can change any tag's area as required. In the case of a child tag, the field
background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area
for each I/O driver and its related I/O tags in order to group all the tags rep-
resenting the equipment processes installed at each I/O device. You might
also consider naming the Area property for the physical location of the tag
(i.e. a station or name of a landmark near the location of the I/O device). For
serial port or Roster tags, you might configure the Area property according
to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as
many areas as you wish and you may leave the area blank for some tags
(note that for Modem tags that are to be used with the Alarm Notification Sys-
tem, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added.
To use an existing area, use the drop-down list feature. Re-typing an exist-
ing area name is not recommended since a typo or misspelling will result in a
second area being created.
There is no tool to remove an area name from VTScada since such a tool is
unnecessary. An area definition will exist as long as any tag uses it and will
stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each
tag a human-friendly note describing its purpose. While not mandatory, the
description is highly recommended.
Tag descriptions are displayed in the tag browser, in the list of tags to be
selected for a report and also on-screen when the operator holds the pointer
over the tag's widget. For installations that use the Alarm Notification Sys-
tem, the description will be spoken when identifying the tag that caused the
alarm.
The description field will store up to 65,500 characters, but this will exceed
the practical limits of what can be displayed on-screen.

Analog Output Properties: I/O Tab

The I/O Tab holds the properties used to identify and establish a con-
nection to the communication driver tag being used to exchange data

with your physical I/O device (e.g. PLC or RTU). This is done by identifying the communication driver tag that communicates with the physical I/O device, the address at the physical I/O device from which this tag is to read its data, and the rate at which the I/O device should be scanned for data.

I/O Device

Select the communication driver tag to which data will be written.By default, the tag will look for a parent tag that is a device driver (..*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Address

Provide the address within the I/O device to which this tag is to write data. This value must match the configuration of your PLC or RTU hardware. Refer to the Addressing topic for your particular device driver for guidance.For some drivers (SNMP and the OPC Client) an address browser is provided to assist you.

Engineering Units

Provide units of measure that the input data represents. Possible values for this field include "rpm" "degrees C", "%", etc

Data Source

The value output by an analog control can be supplied by an operator via a widget on the screen, it can be supplied by a data source, or both can be used.
Normally, the data source would be another tag or an expression. If both a data source and an operator control are provided, then the value of the control tag will be whichever value was most recently set.

Use Data Source for display only

The data source can be used either as the value of the tag, updating the tag's value whenever the data source changes, or it can be used for display purposes only, leaving the value output by the tag unchanged. Use this check box to select one mode or the other.

Analog Output Properties: Scaling Tab
The Scaling Tab requires values that will be used to convert unscaled, raw data from the I/O device associated with this tag into scaled data that will be reported to operators. VTScada uses an algorithm to calculate this tag's value internally, based on the known unscaled and

scaled values you supply.

The default value for each of these fields is stored in an application property. You are advised to change those properties to match the values used by your hardware by using the Edit Properties page of the Application Configuration dialog.

You will need to use the Advanced mode of the properties page. The properties to copy and edit are:

- AnalogUnscaledMinDefault
- AnalogUnscaledMaxDefault
- AnalogScaledMinDefault
- AnalogScaledMaxDefault

Assuming a linear relationship between the values, by providing a known minimum scaled value for a given minimum unscaled value, and a known maximum scaled value for a given maximum unscaled value, VTScada can provide a scaled value for any given unscaled input value.

Unscaled Process Data

Analog Output Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag.

Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create, using the VTScada scripting language.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is used to set the value of this contributor tag as the nth element

in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Analog Output Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Analog Output Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
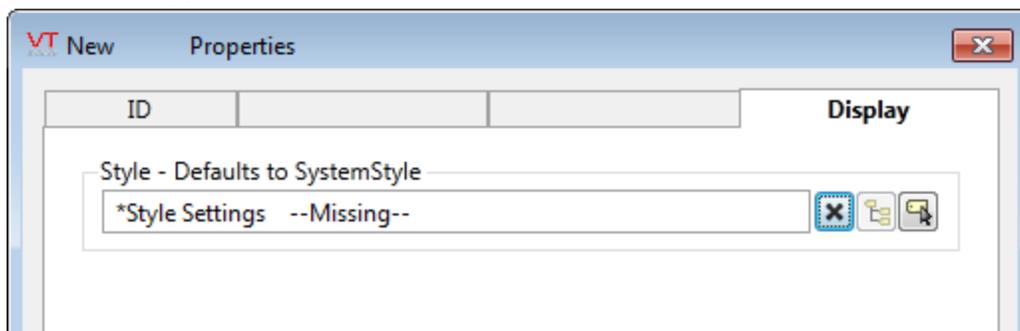
Related:

Style Settings Tags

Analog Output Properties: Alarm Tab

Use the Alarm tab to create new Alarm tags that will be triggered by this tag. The Add button will open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Analog Output Properties: Script Tab

Optionally, link one or more Script tags to this tag. A Script tag provides a means of creating a procedure, using VTScada's programming language, that will run whenever this tag's value changes.

Analog Output Properties: Logger Tab

Use the logger tab to associate a single Logger tag with this tag. The Logger tag will work with the attached Historian to record this tag's data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Note: It is recommended that only one Logger tag be directly associated with a single input or output tag. If you require multiple loggers with different logging rates, please refer to Using Function Tags to Create Multiple Data Logs of an I/O Tag

Analog Output Tag Widgets

The following widgets are available to Analog Output tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Comm Line Widget
Compass 1 Widget	Compass 2 Widget
Draw Text Widget	Droplist Control Widget
	Image Change Widget
Left Bar Widget	Meter 1 Widget

Meter 2 Widget	Meter 3 Widget
Momentary Button Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Set Analog Value Widget
Set Value Button Widget	Set Value Hotbox Widget
Slider Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget
Rotary Control Widget (Knob)	Equipment / Status Color Indicator
Indicator Light Widget	LCD 5x7 Matrix Widget
LCD 7 Segment Widget	

Related Information:

Confirmation Prompts for Output Tags – Control options for confirmation prompt text.

Deadband Control Tags

Counted towards your tag license limit.

Deadband control tags monitor the value of any tag with a numeric value and compare the monitored tag's value to a low set point and a high set point. If the value of the monitored tag falls below the low set point, the value of the Deadband Control tag is set to 1. If the value of the monitored tag rises above the high set point, the value of the Deadband Control tag is set to 0.

A check box enables this action to be reversed.

(Characteristics available in the Table of Type Characteristics.)

Deadband Control Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Deadband Control Properties: Options Tab

The Options tab of a Deadband Control tag's properties folder enables you to select the tag whose value will be monitored, and to configure details such as whether the control is enabled and whether it will send a start signal when the monitored value goes above or goes below the set point.

Monitored Value

This should be either a tag, or an expression based on one or more tag values. While Constant is available as an option due to the nature of this field selector, it will not be useful as a monitored value.

The deadband control compares the Monitored Value with its low and high set points to determine its own value. If the Monitored Value is less than the value of the Low Setpoint, then the value of this Deadband Control tag will be 1. If the Monitored Value is greater than the value of the High Setpoint, the value of this Deadband Control tag will be 0.

Note: The behavior of this Deadband Control tag can be reversed such that if the Monitored Value is greater than the value of the High Setpoint, the value of this Deadband Control tag will be 1 using the Operation radio buttons (see following notes).

Inhibit

The effect of setting the inhibit flag is to force the tag's value to be 0, regardless of other settings.

A valid value for the Inhibit field can be provided via any of a constant, an expression, or a tag. Please see Constant, Expression or Tag for help selecting which to use.

Operation

The Operation section contains two radio buttons that enable you to set the behavior of this tag.

Start On High: If the Start On High radio button is selected, the value of this tag will be reversed from its default. When the Monitored Value is greater than the High Setpoint (see Deadband Control Properties: Setpoints Tab), the value of this

Deadband Control tag will be 1 (rather than 0, which is the default).

Start On Low: If the Start On Low radio button is selected, if the Monitored Value is less than the Low Setpoint (see Deadband Control Properties: Setpoints Tab), the value of this Deadband Control tag will be 1.

Manual Data

The Manual Data property enables you to set a user-defined value for this tag. If the value of the Manual Data field is left Invalid, or is not a valid number, then the value for this tag is read.

Note: If the value entered into the Manual Data field is a 0, the actual value of this tag is also overridden, as 0 is considered a valid value. Therefore, you must clear the Manual Data field entirely if you wish the actual value of this tag to be read, rather than a manually-entered value.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

Deadband Control Properties: Setpoints Tab

In this tab, you can configure the low and high set points that will be compared to the monitored value.
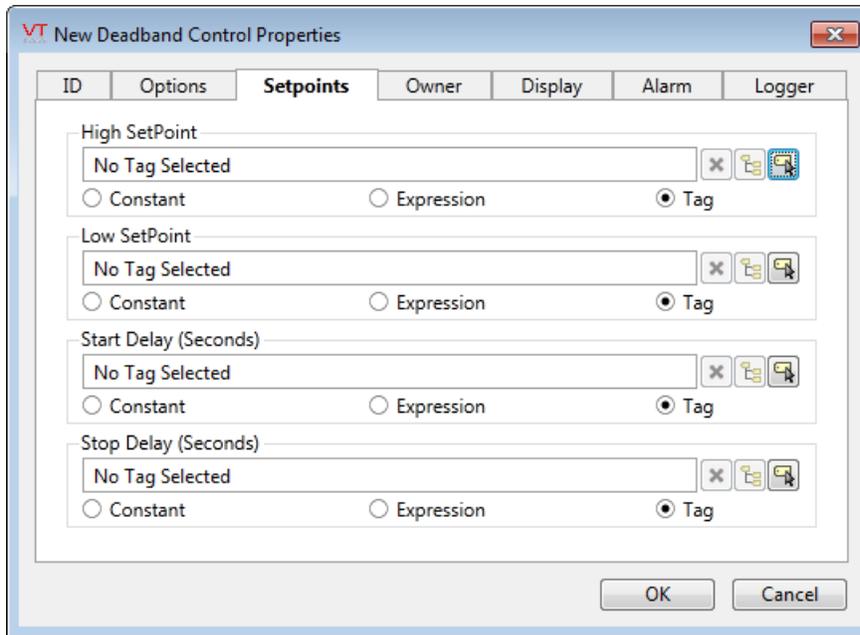
High Setpoint and Low Setpoint

The Deadband Control tag will compare the values in the Setpoint fields with the Monitored Value to determine its own value.

If the Monitored Value exceeds the value of the High Setpoint, the value of this Deadband Control tag will be 0. The result of this comparison is referred to as the stop condition.

If the Monitored Value is less than the value of the Low Setpoint, the value of this Deadband Control tag will be 1. The result of this comparison is referred to as the start condition.

Note: The behavior of this Deadband Control tag can be reversed such that if the Monitored Value is greater than the value of the High Setpoint, the value of this Deadband Control tag will be 1.

A valid value for the High Setpoint field and the Low Setpoint field can be provided via any of a constant, an expression, or a tag. Please see Constant, Expression or Tag for help selecting which to use.

Start Delay and Stop Delay

You can enter a value (measured in seconds) in the Start Delay field to set a length of time that to wait after the start condition before setting the value of this Deadband Control tag to 1.

The Stop Delay field works in the same way: Enter the number of seconds that the tag should wait after the stop condition before setting the value of this Deadband Control tag to 0.

Deadband Control Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag.

Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create,

using the VTScada scripting language.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is used to set the value of this contributor tag as the nth element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing

manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Deadband Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
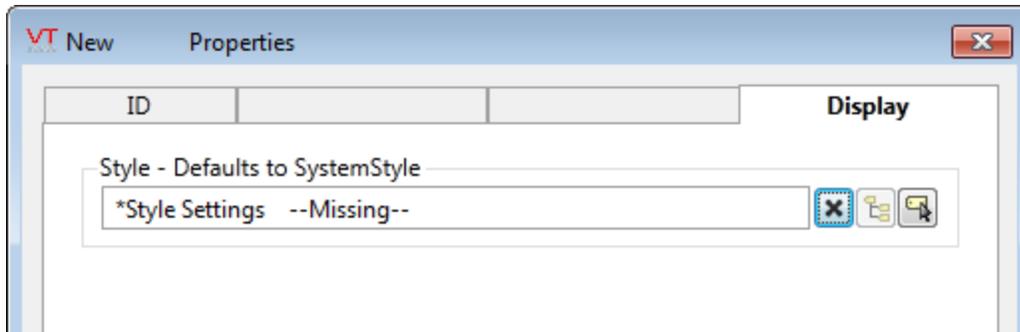
Related:

Style Settings Tags

Deadband Control Properties: Alarm Tab

If you want this tag to trigger an alarm, use the Add button to open a con-figuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Deadband Control Properties: Logger Tab

The logger tab enables you to associate a single Logger tag with this tag. The Logger tag will work with the attached Historian to record this tag's data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Deadband Control Tag Widgets

Deadband control tags monitor the value of any tag with a numeric value and compare the monitored tag's value to a low set point and a high set point. If the value of the monitored tag falls below the low set point, the value of the Deadband Control tag is set to 1. If the value of the monitored tag rises above the high set point, the value of the Deadband Control tag is set to 0. The widgets for Deadband Control tags enable the operator to set the value if it falls between the low and high set points. The following widgets are available to Deadband Control tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	LCD 5x7 Matrix Widget
Draw Text Widget	
Equipment / Status Color Indicator	Image Change Widget
Indicator Light Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Multi-Color Widget	Multi-Text Widget
Network Link Widget	Plot Data Widget
Right Bar Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget

Related Information:

Confirmation Prompts for Output Tags – Control options for confirmation prompt text.

Digital Control Tags

Counted towards your tag license limit.
Digital Control tags transmit digital data that is entered by the user or from another tag and write it to an I/O device.
A Digital Control tag might be used to turn a pump off or on, open a valve, or change the mode of a piece of equipment from manual to auto.

Retained or Persisted values

Analog Control and Analog Output tags will both retain their last written value across application restarts. Use these tags when there is a need for a value to persist.
String I/O tags with output enabled, and with no driver ("No tag selected" rather than the default, "[*Driver] None found") will retain their last written value across application restarts.
The Digital Control and Digital Output tags do not have a similar feature, but an analog tag can be used in their place, writing a zero or a one. If the intent is to write a clear zero or one with an analog tag, both the scaled and unscaled range should be adjusted for a minimum value of zero and a maximum value of one, so that scaling does not adjust the values.
(Characteristics available in the Table of Type Characteristics.)

Digital Control Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).
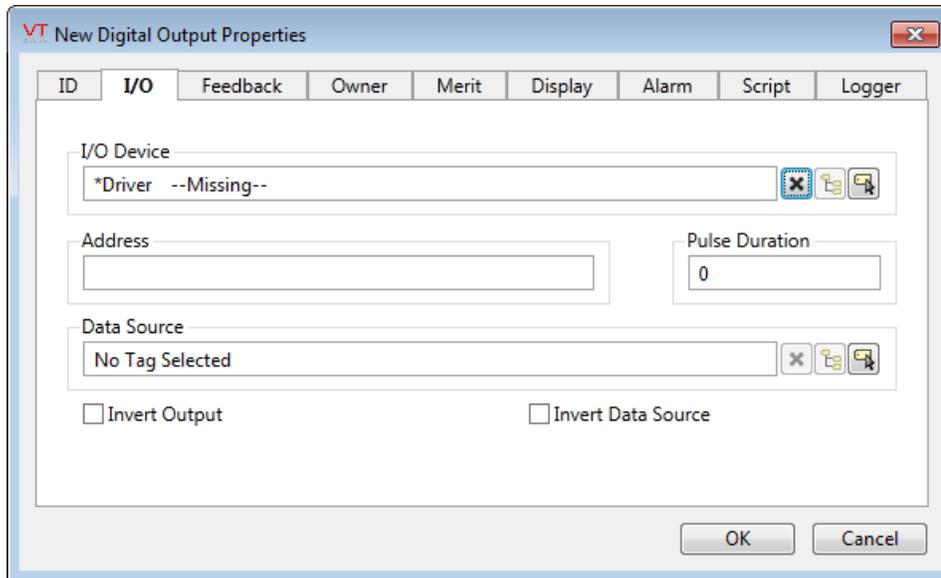
Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Digital Control Properties: I/O Tab

The I/O Tab contains properties used to identify and establish a connection to the communication driver tag being used to exchange data with your physical I/O device (e.g. PLC or RTU), or to the Polling drive responsible for determining the order and rate at which data polls will occur.
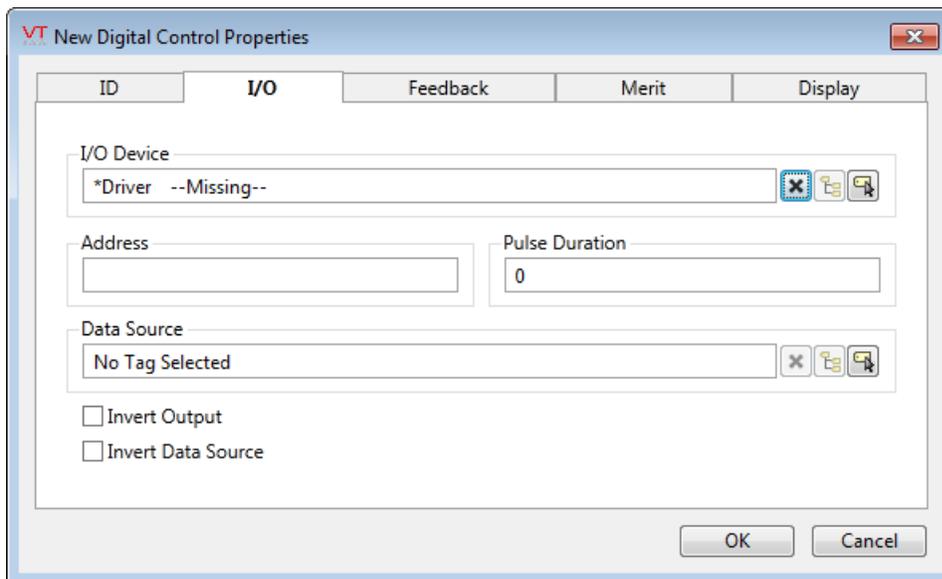
I/O Device

Select the communication driver tag to which data will be written.

By default, the tag will look for a parent tag that is a device driver (..*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dia-

log with which you can add or remove a Parameter Expression, or open a
selected driver's properties dialog.

Address

Provide the address within the I/O device to which this tag is to write data.
This value must match the configuration of your PLC or RTU hardware. Refer
to the Addressing topic for your particular device driver for guidance.
For some drivers (SNMP and the OPC Client) an address browser is provided
to assist you.

Pulse Duration

If you are using a pulsed signal, enter the duration of the pulse here in
seconds or decimal parts of a second.
If the value of Pulse Duration is 0 or blank (Invalid), then a constant value will
be written to the I/O device.
A pulsed output is used to send a 1; seldom if ever, a zero. Use care with the
configuration of the linked widget or multi-write tag.

Data Source

The value output by the control can be supplied by a data source. For a
digital control, the values supplied by the data source should be limited to 1
or 0 (true or false).

Off Text and On Text

Some widgets used by Station tags are able to display the text you provide
here. Note that the value of Digital Control itself is used to indicate "currently
writing" or "not currently writing" rather than the value being written. See
also: Confirmation Prompts for Output Tags.

Invert Output

Reverse the value of the output before writing.

Invert Data Source

Reverse the value coming from the data source before using.

Digital Control Properties: Feedback Tab

The Feedback tab can be used to select an alarm, alarm priority, digital
status, or Digital Control tag that will provide feedback when this tag's
control action has successfully been completed.

Note: Remember, the existence of an output tag is not enough to out-
put a value to the I/O device; you must draw the tag using a control
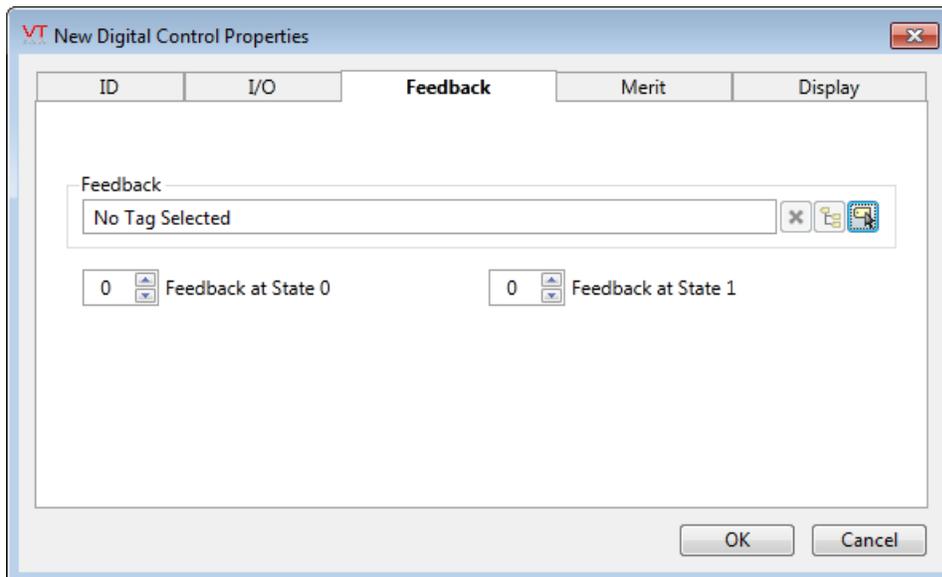
Feedback

The Feedback field enables you to specify an existing alarm,
alarm priority, digital status, or Digital Control tag whose
value can be used to determine whether the control action for
this tag was successful. When a value is output by this tag, the
feedback tag reads its value from the I/O device at the same
address, and compares this value with the expected value(s)
configured in the Feedback At State 0 and Feedback At State 1
spin boxes (see following notes).
If the Pulse Duration property for this tag (as set on the I/O
tab) is set to a valid value greater than 0 (i.e. a pulsed signal),
then the Feedback At State 0 property is ignored, as the logic
assumes that the control attempt is always to drive the value of
the feedback tag to the Feedback At State 1 value.

Note: The 0 state may be inverted to a 1 before writing to the I/O
device if this tag's Invert Output property (I/O tab) is set to 1 (true).

Feedback At State 0

If the Pulse Duration property for this tag, as set on the I/O tab, is set to any of: 0 (the default), invalid, or blank (i.e. a non-pulsed signal) then...

The Feedback At State 0 spin box should be set to the value you expect for the selected Feedback tag when this tag has successfully gone to the 0 state. For example, if you've configured this tag to stop a motor by issuing a value of 0 to the I/O device, you would expect the feedback tag (e.g. a Digital Status tag) to read a 0 back from the I/O device, indicating that your output was received.

Feedback At State 1

If the Pulse Duration property for this tag, as set on the I/O tab, is set to any of: 0 (the default), invalid, or blank (i.e. a non-pulsed signal), then...

The Feedback At State 1 spin box can be used to set the value you expect for the feedback tag when this tag has successfully gone to the 1 state. For example, if you've configured this tag to stop a motor by issuing a value of 1 to the I/O device, you would expect that the feedback tag (e.g. a Digital Status tag) would read 1 back from the I/O device, indicating that the value was received.

This value can be used as the source for an Alarm tag with a time delay to alert the operator that the requested output change has not had the desired effect within a reasonable period of time.

Digital Control Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and

you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application
security privilege from this drop down if you wish to limit the operation
of this control to only those operators who have been granted the match-
ing security privilege.

Digital Control Properties: Display Tab

Style

When this tag is represented on screen by widgets that are
able to use a Style Settings tag, you can save development time
by choosing the Style Settings tag that holds the correct dis-
play configuration for this tag instance.
The default configuration will search for the first parent tag of
the Style Settings type. If there is none, (message reads, "--
Missing--") it will use System Style, which is the default style
tag that is automatically part of every new VTScada application.

Related:
Style Settings Tags

Off Text and On Text

Certain widgets will display the text you configure here. This

text will also be used in confirmation boxes for the control action (if configured).

Display Order

Used to specify the placement of this tag when viewed in an associated station dialog. See Order Tab for further details.

Digital Control Tag Widgets

Digital Control tags enable users to write digital data to a PLC or RTU, which transmits the entered value to the equipment process associated with the tag. The widgets for Digital Control tags enable users to set values to be written to the equipment process associated with the tag, and display these set values.

The following widgets are available to Digital Control tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Momentary Button Widget

...Set Value Button Widget

...Set Value Hotbox Widget

...Text Change Widget

...Two Color Bar Widget

Note: Of the widgets displayed above, only the Momentary Button, Set Value Button, and Set Value Hotbox widgets allow users to actually

output data to the equipment process associated with the tag being drawn. All other widgets simply display the value that has been output.

Related Information:
Confirmation Prompts for Output Tags – Control options for confirmation prompt text.

Digital Output Tags

Counted towards your tag license limit.
Digital Output tags take digital information from the operator (via a widget) or from another tag and write it to the I/O device.
The value of a digital output can be:

- 1 when there is an attempt to set the output that has not yet been confirmed by a change in the feedback value;

- 0 if neither case is met. i.e. when not performing a write, or after confirmation is received that a write has occurred.

- Of interest only to programmers: A transient value of 2 will occur when the feedback value changes without any prompting from the digital output. This value is extremely transitory and will not be caught by any alarm.

Retained or Persisted values

Analog Control and Analog Output tags will both retain their last written value across application restarts. Use these tags when there is a need for a value to persist.
String I/O tags with output enabled, and with no driver ("No tag selected" rather than the default, "[*Driver] None found") will retain their last written value across application restarts.
The Digital Control and Digital Output tags do not have a similar feature, but an analog tag can be used in their place, writing a zero or a one. If the intent is to write a clear zero or one with an analog tag, both the scaled and unscaled range should be adjusted for a minimum value of zero and a maximum value of one, so that scaling does not adjust the values.

Digital Output Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an

existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Digital Output Properties: I/O Tab

The I/O Tab contains the properties used to identify and establish a connection to the communication driver tags being used to exchange data with your physical I/O device (e.g. PLC or RTU). This is done by identifying the communication driver tag that communicates with the physical I/O device, and the address at the physical I/O device to which this tag is to write its data.

I/O Device

Select the communication driver tag to which data will be written.
By default, the tag will look for a parent tag that is a device driver
(..*Driver). If none is found, the text "--Missing--" will be displayed. The
tag button to the right of the field opens the tag browser, from which
you can either select an existing communication driver tag or add a new
one. The X button will clear the field. Right-clicking on a tag in the field
will open a dialog with which you can add or remove a Snapshot Expres-
sion, or open a selected driver's properties dialog.

Address

Provide the address within the I/O device to which this tag is to write
data. This value must match the configuration of your PLC or RTU hard-
ware. Refer to the Addressing topic for your particular device driver for
guidance.
For some drivers (SNMP and the OPC Client) an address browser is
provided to assist you.

Pulse Duration

If you are using a pulsed signal, enter the duration of the pulse here in
seconds or decimal parts of a second.
If the value of Pulse Duration is 0 or blank (Invalid), then a constant value will
be written to the I/O device.

A pulsed output is used to send a 1; seldom if ever, a zero. Use care with the configuration of the linked widget or multi-write tag.

Data Source

The value output by the control can be supplied by a data source. For a digital control, the values supplied by the data source should be limited to 1 or 0 (true or false).

Invert Output

Reverse the value of the output before writing.

Invert Data Source

Reverse the value coming from the data source before using.

Digital Output Properties: Feedback Tab

The Feedback tab can be used to select an existing tag, or create a new tag belonging to the Digitals Group, that will provide feedback indicating the current status of this tag's control action.

When the feedback tag's value changes to what was expected as a result of the digital output, then the Digital Output tag's value will go to 0, indicating that no output is in progress (i.e. the write has finished)

Until the feedback tag changes to the value expected as a result of the digital output (perhaps because a communication error is preventing the output from occurring) then the Digital Output tag's value will remain at 1, indicating that the output is still in progress.

If the feedback tag's value changes independently of a write occurring from the digital output, the tag's value will undergo a transitory change to a 2. This signal is designed for the use of VTScada programmers – it does not last long enough to be displayed and cannot be captured by an alarm.
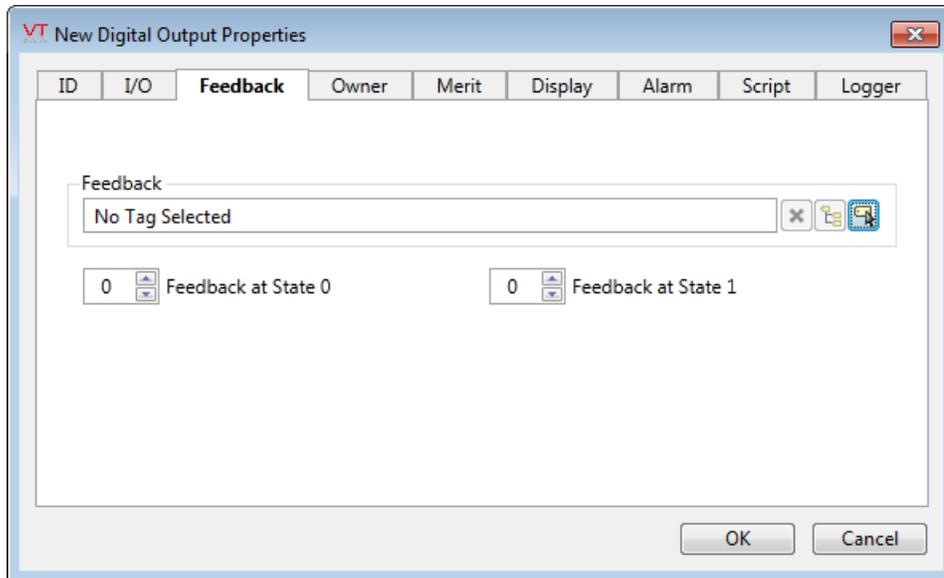
Feedback

> Select or create a tag from the digitals group that will provide the source of the feedback.

Feedback At State 0

> The Feedback At State 0 spin box can be used to set the value you expect back from the feedback tag when the digital output has successfully written a 0 (State 0).

Feedback At State 1

> The Feedback At State 1 spin box can be used to set the value you expect back from the feedback tag when the digital output has successfully written a 1, 2 or 3 (State 1).

Digital Output Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag.

Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create, using the VTScada scripting language.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is used to set the value of this contributor tag as the nth element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable

data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Digital Output Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Digital Output Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
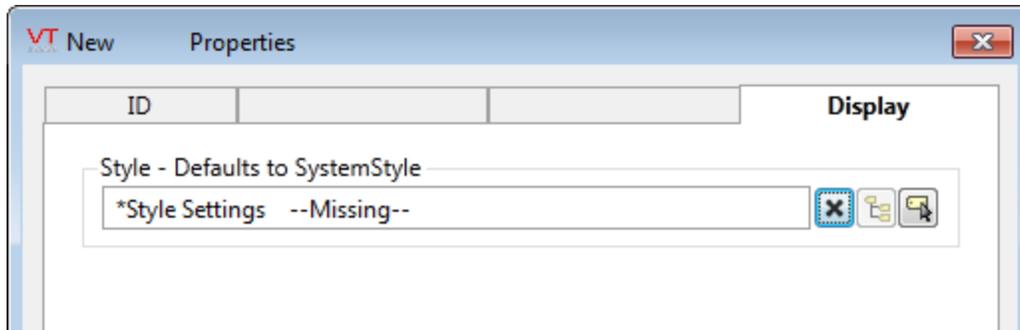
Related:

Style Settings Tags

Digital Output Properties: Alarm Tab

If you want this tag to trigger an alarm, use the Add button to open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Digital Output Properties: Script Tab

Optionally, link one or more Script tags to this tag. A Script tag provides a means of creating a procedure, using VTScada's programming language, that will run whenever this tag's value changes.

Digital Output Properties: Logger Tab

The logger tab enables you to associate a single Logger tag with this tag. The Logger tag will work with the attached Historian to record this tag's

data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Note: Only one Logger tag can be directly associated with a single input or output tag. If you need to have multiple loggers with different logging rates, please refer to Using Function Tags to Create Multiple Data Logs of an I/O Tag

Digital Output Tag Widgets

Digital Output tags write digital equipment process data to a PLC or RTU. The widgets for Digital Output tags enable users to write this data and view the data being written.

The following widgets are available to Digital Output tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Momentary Button Widget

...Plot Data Widget

...Set Value Button Widget

...Set Value Hotbox Widget

...Text Change Widget

Note: Of these widgets, only the Set Value Button, Set Value Hotbox, and Momentary Button methods enable users to output values to the associated tag; the remaining widgets display the value of the tag.

Related Information:

Confirmation Prompts for Output Tags – Control options for confirmation prompt text.

Multi-Write Tags

Counted towards your tag license limit.
The MultiWrite tag will write predefined values to a list of selected tags upon triggering. This can be used to place a plant into a state of operation or to quickly shut a plant down in one step.

Reference Notes:

Up to 100 output tags can be controlled by a single MultiWrite tag, with a defined value set to be written to each. The trigger for the write may be any of a manual button press, a tag that changes state from false to true, or an expression that evaluates to true.
Each value to be written will be checked to ensure that it is valid before it is written. Invalid writes will be ignored. All valid values will be written, regardless of whether or not they have changed since the last write.
Should 100 tags not be enough, it is possible for one MultiWrite tag to trigger another upon finishing its write sequence.
An event will be recorded in the alarm history whenever a MultiWrite is triggered.
If triggering a Digital Control or Digital Output with a pulse duration, use care that you do not attempt to send a pulsed zero.
(Characteristics available in the Table of Type Characteristics.)

MultiWrite Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

MultiWrite Properties: Write List Tab

The Write List displays the output tags that will be written to, and controls what value will be written to each. The outputs will be written in the order in which they are displayed.
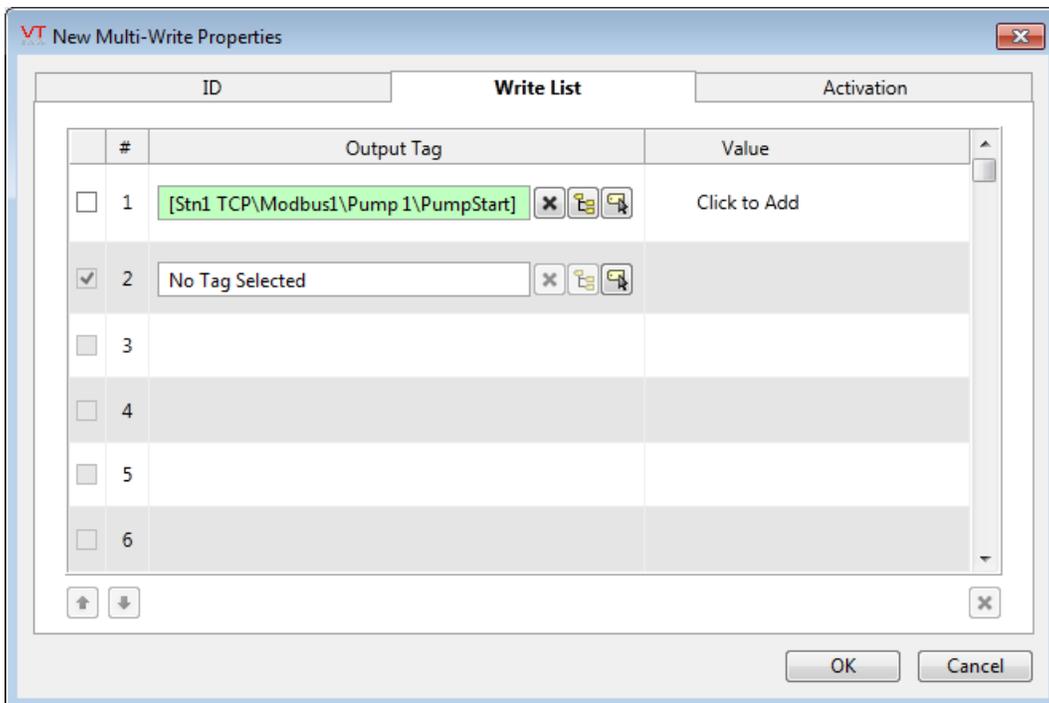
You can add or edit any tag in the list by clicking on a row. You must select a tag before configuring the value to be written to it.

The Tag and Value input fields will be activated when a row is selected, except that rows must be filled in sequence from #1. You cannot add a tag to row three before row two has been completed.

The grid displays the name and description of each tag selected to be written to. The Value column shows what will be written to each tag when the MultiWrite is triggered. The value may be any of a constant, an expression or another tag's value. In the case where another tag's value is to be written, that tag's name will be display in the Value column. Note that, the display may show a tag's Unique ID rather than its name when the tag is not selected. The description will always be shown for every tag, to help avoid confusion.

To add a tag to the list, select the next available row, then click on the Tag Browser button. Select or create an output tag. After you have specified the tag to write to, you can provide a value to write using the Value field.

To remove a tag from the list, select its row, then click on the X button in the lower-right corner of the window.

The arrow buttons below the grid may be used to move a selected row up or down in the list, thus controlling the order in which output tags are written to.

MultiWrite Properties: Activation Tab

The Activation tab provides a way for MultiWrite tags to be controlled automatically. Any tag or expression that will change from a false to a true value can be used to trigger the write.

If you intend to provide a button for operators to manually trigger the write, then you can also define an application Security Privilege that will restrict access to that button to authorized operators.
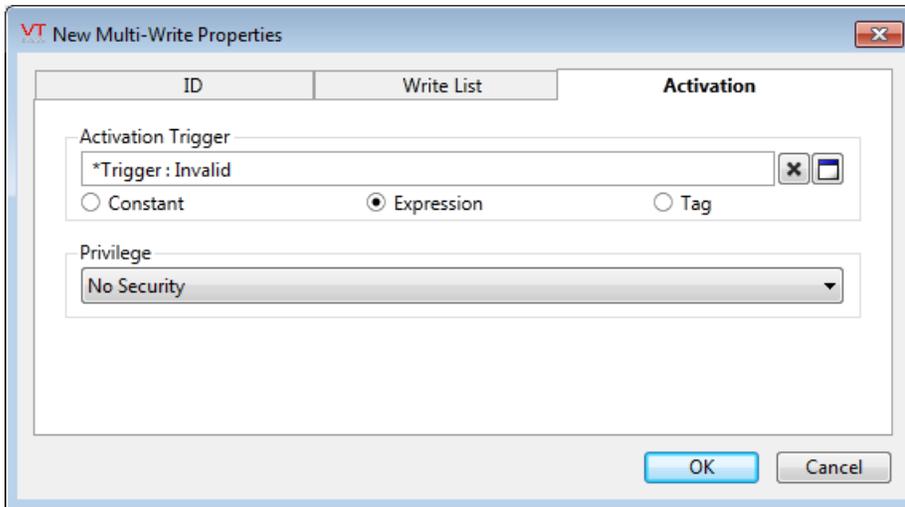
MultiWrite Tag Widgets

MultiWrite tags write defined values to a group of PLC or RTU addresses, as defined by the associated tags. The widgets for these tags enable users to trigger the writing of this data.

...Multi-Write Button Widget

...Multi-Write Hotbox Widget

Selector Switch Tags

Counted towards your tag license limit.
Selector Switch tags provide the means to create multi-position switches that can be used to create on-off toggles, mode selectors, or Hand-Off-Auto (H-O-A) switches.
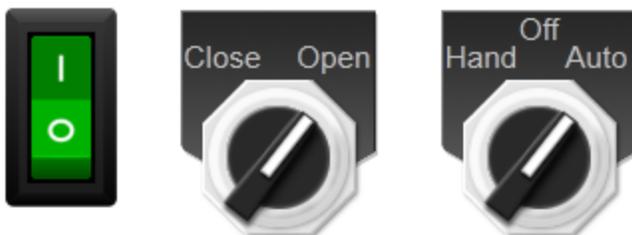
Reference Notes:

For each switch position, a value will be written to an address. The address can be the same for each position, or it can be different. If it is different, then it is customary to configure the switch to write a zero to the address for each position not selected. Also, you configure what

value will be written for each switch position.

Selector Switches can be used as digital controls, but they also have many differences from Digital Control tags. Use care when choosing which type to use for any situation.

A Selector Switch can accept two types of feedback:

- Equipment feedback indicates the current status or state of the equipment. When configuring the Selector, you can specify what the expected value will be for each switch position.

- Switch Position feedback updates the selected position of the Selector Switch. This value must match the numeric switch position (0, 1, or 2). A new value arriving via the position feedback does not cause the Selector Switch tag to do a fresh write.

In both cases, feedback must be provided by a tag or an expression.

Selector Switch Mismatch Conditions

The selector switch is designed to alert the operator to a mismatch condition by providing visual feedback through its native widgets.

There are two possible causes for a mismatch condition:

- **Position mismatch** The switch position as requested by the operator does not match the actual position as reported by the switch position feedback value. This can occur when there is a lag in response time from the equipment.

- **Value mismatch** The equipment feedback does not match the expected value for the requested or current position. This type of mismatch can also be used to trigger an alarm. The alarm must be enabled and the mismatch must exist for a set length of time before an alarm will become active.

The native widgets for the selector switch will use the following conventions to notify the operator of a mismatch or an alarm condition:

- Alarm indicators take precedence. While an alarm indicator is active, no other mismatch indicator will be displayed.

- Alarms are normally indicated with red, and (if possible) will be shown on the requested or current position of the switch.

- Alarm indicators are displayed until the alarm condition is cleared, regardless of whether the alarm has been acknowledged.

- When a position mismatch occurs and no alarm state is active, then (if possible) the requested position will be indicated by the widget – often by blinking.
- When a value mismatch occurs and the alarm is either disabled or has not yet triggered, an indication similar to a position mismatch will be displayed, if possible.

Selector Switch Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For

serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Selector Switch Properties: I/O Tab

The I/O Tab includes properties used to identify and establish a connection to the communication driver tags being used to exchange data with your physical I/O device (e.g. PLC or RTU). This is done by identifying the communication driver tag that is linked with the physical I/O device.

In addition, the I/O tab also provides a means of selecting tags or expressions that are to provide feedback for the state of the equipment or external control over the switch position.

Feedback versus Switch Position Feedback

Feedback is a value that you read from the controller by configuring a tag to monitor an addresses on the controller. By monitoring a feedback value, you can ensure that the correct value was written after changing the switch position, and that the controller and the SCADA system agree on the current switch position.

Referring to the labels from the configuration panel (see following image):

Feedback

The Feedback field is used to ensure that the correct value was written to the remote device. This is similar to the feedback field of a Digital Control tag. When configuring the switch positions and the value to write for each position, you will also configure the value that will be expected via the feedback for each position of the switch.

If the value of the selected feedback tag or expression does not match the expected value for the current position, then this will trigger an alarm, assuming that the alarm option of the selector has been enabled.

Switch Position Feedback

Switch Position Feedback is used to ensure that the selector in the SCADA system remains synchronized with the controller. The tag or expression used for this field must be able to have values matching the possible positions of the selector. (0 and 1 for a two-position selector, 0, 1 and 2 for a three-position selector).

The VTScada Selector Switch position will follow the switch position feedback value. If the tag or expression providing position feedback changes to a new value, this Selector Switch will update to match. While this will update the switch's selected position, it does not cause the tag to do a fresh write to the equipment.

If an operator changes the position of this Selector Switch within the application, but the switch position feedback indicates that the physical position has not changed, the widget will blink to show a mismatch.
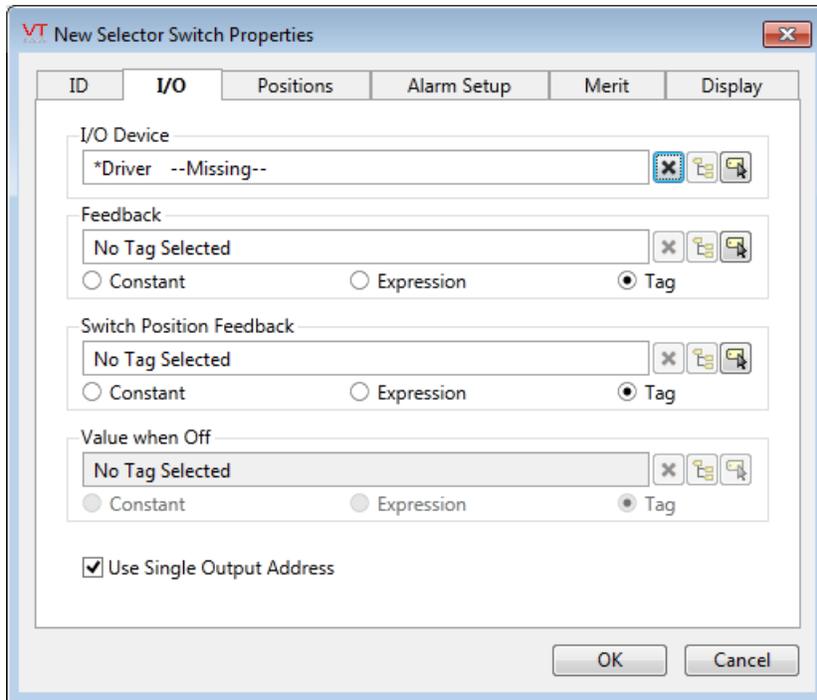
I/O Device

Select the communication driver tag to which data will be written.By default, the tag will look for a parent tag that is a device driver (..*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Feedback (equipment feedback)

[Optional] Select the tag or expression that has been configured to provide feedback indicating whether the control action of this switch has been successful. "Success" means that the state of the equipment matches what is expected (as configured on the Positions tab) for each possible switch position.

A mismatch in values here will trigger an alarm, if one is configured on the Alarm Setup tab.

Switch Position Feedback

[Optional] Select the tag or expression that has been configured to provide an indication of the current position of the equipment. (0, 1 or 2 according to switch positions) The selector switch will change positions to follow this feedback value. If the selector switch is changed, but the feedback value does not follow, the widget will blink, but this will not by itself cause an alarm. See Feedback (equipment feedback).

Value When Off

Available only when the Use Single Output Address option has been deselected. The value provided here will be written to the addresses of the unselected switch positions after the current position's value has been written.

Use Single Output Address

This check box provides an option to use a single output address for all switch positions, or to allow a separate address for each position.
In the case where each position of the switch writes to a different output address, you can deselect the "Use Single Output Address" box, and provide a value that will be written to each address not matching the current switch position. Note: Addresses are specified on the next tab, Positions.
Values will be written in a "make before break" order. In other words, the value for a new switch position will always be written before the value-when-off is written to the addresses matching the unselected switch positions

Selector Switch Properties: Positions Tab

In the Positions tab you can configure the expected feedback values, the values to write and the address(es) corresponding to each of the possible switch positions. (Whether a single or multiple output addresses are used, depends on the check box in the I/O panel.)
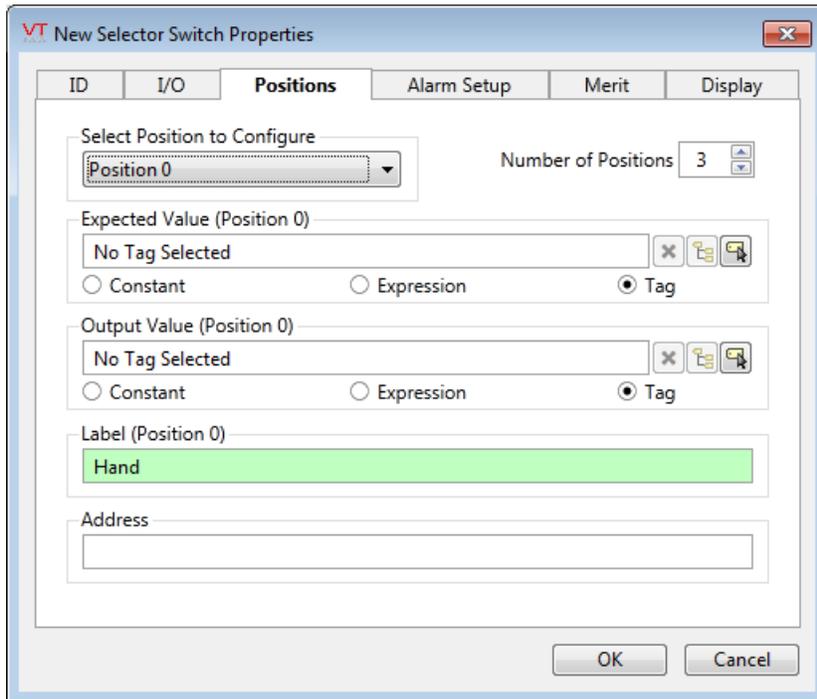
Select Position to Configure

Select each of the available switch positions to configure the output and expected values for that position. The label over each parameter will change to show which position is being configured. The list is controlled by the parameter, Number of Positions.

Number of Positions

May be either "2" or "3". Limits the number of configurable positions in the switch. Always select "2" for a toggle switch. (This differs from earlier versions where it was common practice to create toggle switches by configuring positions "0" and "2,"

leaving position "1" blank.)

Expected Value (Position x)

Specify the value that is expected to be returned in the feed-back field of the I/O panel, to indicate that the tag's switching operation caused a successful change in the system. This value can be a constant, an expression, or it may come from a tag.

Output Value (Position x)

This field is where you to specify the constant, expression or tag value that will be written to the output address when the corresponding position is selected. If a tag or expression is used here, then changing output values will trigger a write to the device as long as the position has not changed.

Label (Position x)

If the switch requires operator confirmation (configured when drawing the switch as a toggle) then this label will appear as part of the confirmation prompt for the matching switch position.

Address

The I/O address to write the output value to when position x is selected.
If the Use Single Address option was checked in the I/O panel, then this will field will be labeled simply "Address" since the same address is used for all positions. Otherwise, the label will be "Address (Position x)" to indicate which switch position you are providing the address for.

Selector Switch Properties: Alarm Setup Tab
Specify the parameters of the alarm that will be triggered when the

equipment feedback for the tag does not match expected value.
Since physical equipment can sometimes take a few moments to
respond, a delay can be specified to prevent the alarm from being
triggered while waiting for the expected feedback value.
By design, this alarm cannot be configured as a trip alarm.

> Note: Invalid data, whether on the I/O Device or a feedback tag, will
> neither cause nor clear an alarm condition.
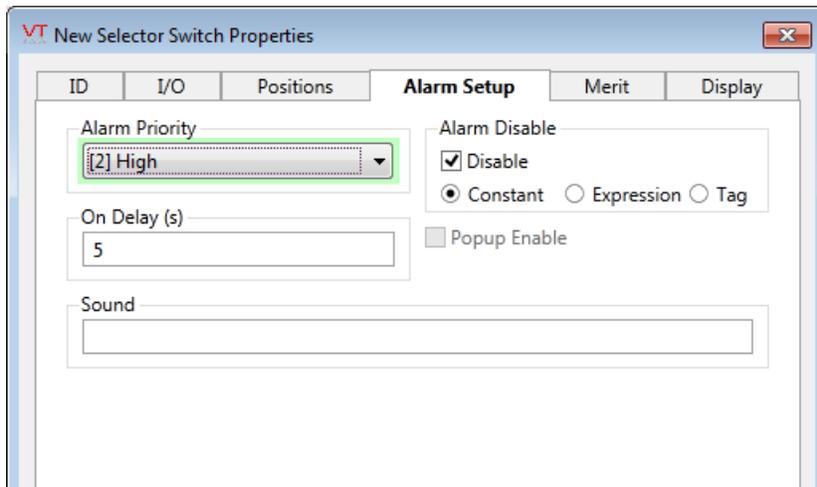
Alarm Priority

Select the priority level of this alarm. All other controls are dis-
abled while the priority level is set to "None".

On Delay

Specify a delay in seconds to wait when the feedback value
from the PLC does not match the expected value for the reques-
ted switch position before triggering an alarm.

Sound

[optional] Enter the name of a .wav file in your application dir-
ectory in order to use that as the alarm sound.

Alarm Disable

Deselect in order to enable alarms on this switch. May be linked to a tag or expression for suppression by design.

Pop-up Enable

Available only when alarm pop-ups are enabled in the application properties (see: AlarmPopupsEnable)

Selector Switch Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

Selector Switch Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.
The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use

System Style, the default style tag that is automatically part of every new VTScada application.
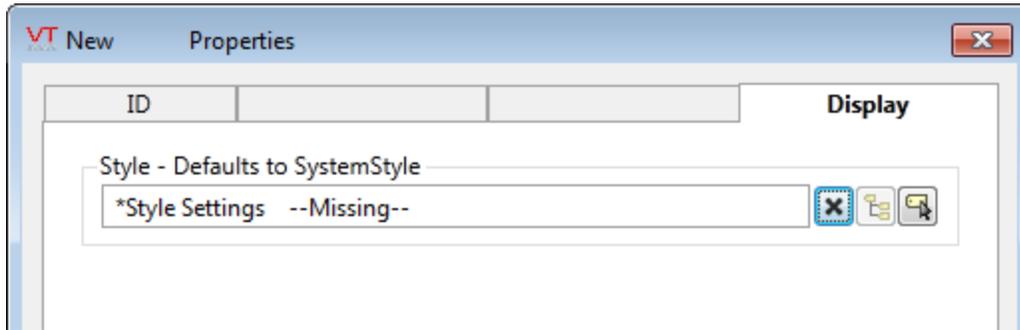
Selector Switch Tag Widgets

Selector switch tags allow you to add multi-position widgets such as Toggle buttons, mode selectors and Hand-Off-Auto switches to your applications.

The following widgets are available to Selector Switch tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Checkbox Switch Widget	Color Blink Widget
Color Box Widget	Color Fill Widget
Color Line Widget	Compass 1 Widget
Compass 2 Widget	LCD 5x7 Matrix Widget
Draw Text Widget	Droplist Control Widget
Equipment / Status Color Indicator	Horizontal Button Widget
Image Change Widget	Indicator Light Widget
LCD Bar Widget	
LCD 7 Segment Widget	Left Bar Widget
Meter 1 Widget	Meter 2 Widget
Meter 3 Widget	Momentary Button Widget

Multi-Color Widget	Multi-Text Widget
Numeric Entry Widget	Numeric Value Widget
Plot Data Widget	Right Bar Widget
Selector Switch Widget	Set Analog Value Widget
Set Value Button Widget	Set Value Hotbox Widget
Slider Widget	Text Change Widget
Top Bar Widget	Toggle Switch Widget
Two Color Bar Widget	Vertical Button Widget

Trigger Tags

Counted towards your tag license limit.

Trigger tags can be used to activate a process at a set time and date.
They may be configured to repeat at regular intervals, or they can be configured to trigger based on an expression or another tag's value. By drawing the Trigger tag on a page, you can also provide a way for operators to manually trigger an event.

As well as defining a schedule or event that will set the trigger to 1 (On), you can should also define a schedule for when it will be reset to 0 in preparation for the next trigger event. This may be configured as a delay measured in seconds after the On trigger event, it may be set to a regular schedule or it may be tied to an expression or tag value.

Note: tThe trigger depends on the under-lying Windows operating system and your computer system clock. It should not be relied upon for precise to-the-second starts or stops.

An "Enable" parameter is provide to allow you to control whether the Trigger tag is activated or not. If the tag is drawn on a page, this can be toggled by right-clicking on the widget. Disabling a Trigger tag will also force its value to be 0 (OFF).

Trigger Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Trigger Tag Properties: On Conditions Tab

This tab is used to set the condition that will switch the tag's value to 1. The tag may be switched to an ON state based on a schedule or based on an expression (which may include a tag).

Both options may be configured, resulting in a tag that will be triggered both on a regular schedule and also in response to an event.

Note the field labeled "Enable". When not checked, this means that no schedule is in effect. All other time-related fields will be disabled. Note also the Starting Time. This serves as both an initial trigger time and the base date for repeating trigger events. For example, to trigger a report to be created at midnight on the first day of each month, you would set the following values:

- "Enable" field checked.
- "Starting Time" set to 12:00 a.m. on the first day of this month.

(Setting the first day of next month would achieve the same result, but putting the base date of repeating events in the past ensures that you do not miss the first iteration.)

- "Every" value set to 1.
- "Months" option selected.
- "Same Day as Starting Day" selected. ("Same Day" refers to date in the month, not day of the week)

You must also configure the Off Conditions tab to set the schedule or condition that will reset the tag's value to 0. Once the trigger has occurred and the tag's value set to 1, it remains at a value of 1. If you need a 0 to non-zero transition to trigger an event, you must reset the value to Off (0) before the next desired On-time.
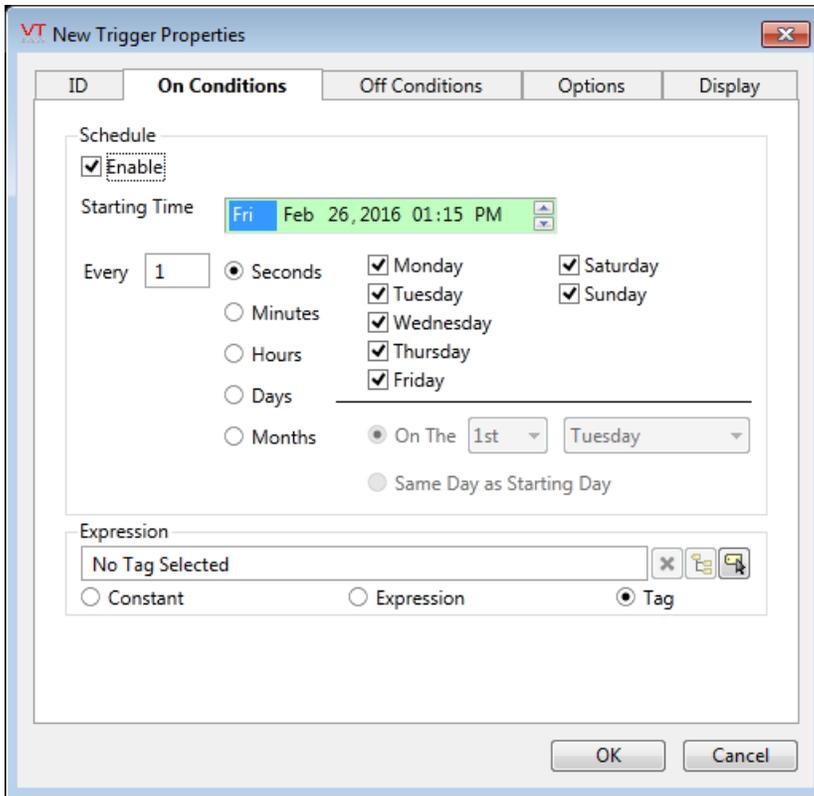
Enable

Must be checked in order for the trigger to activate based on a time and date schedule.

Starting Time

Sets the base date and time for the schedule configuration options. Subsequent configuration options will be counted from this start time. If set to the future, no trigger will occur until that date.

Every [N] Seconds / Minutes / Hours / Days

The field beside "Every" controls the frequency of the following units.

For example, if the Every value is set to 1 and the Hours check box is selected, then beginning with the time configured in the Start Time, the trigger will be set to true every hour.

If the Every value is set to 7 and the Days check box is selected, then beginning with the configured Start Time date, the trigger will be set to true every week at the time given in Start Time.

Day check boxes

Allow you to control which days the Trigger tag will operate on. For example, you may choose to disable the trigger's action on weekends by deselecting Saturday and Sunday.

These check boxes will not be enabled when the Months option is selected.

Months

Selecting the Months option is similar to configuring to Trigger tag to switch on every N Hours. If this option is chosen, then further refinements include "On The" or "Same Day as Starting Day".

For example, you may wish to configure the trigger to run "on the 1st Wednesday" or "on the 2nd Monday" of the month. Alternatively, of you have selected "Same Day as Starting Day" then the trigger will run on the given numeric day of the month, at the selected time.

When the Months option is selected, the Day check boxes will not be enabled.

Expression

Select a tag, or enter an expression that will be used to activate the Trigger tag. The On condition will be set when the selected tag or expression's value becomes non-zero.

Trigger Tag Properties: Off Conditions Tab

This tab is used to set the condition that will switch the tag's value to 0, resetting the Trigger tag for the next scheduled event. The tag may be switched to an OFF state based on a delay in seconds after switching ON, a schedule, or based on an expression (which may include a tag).

A combination of options may be configured, resulting in a trigger that will reset either based on time, or in response to an event.

Note the field labeled "Every". When set to 0 this means that no schedule is in effect. All other time-related fields will be disabled.
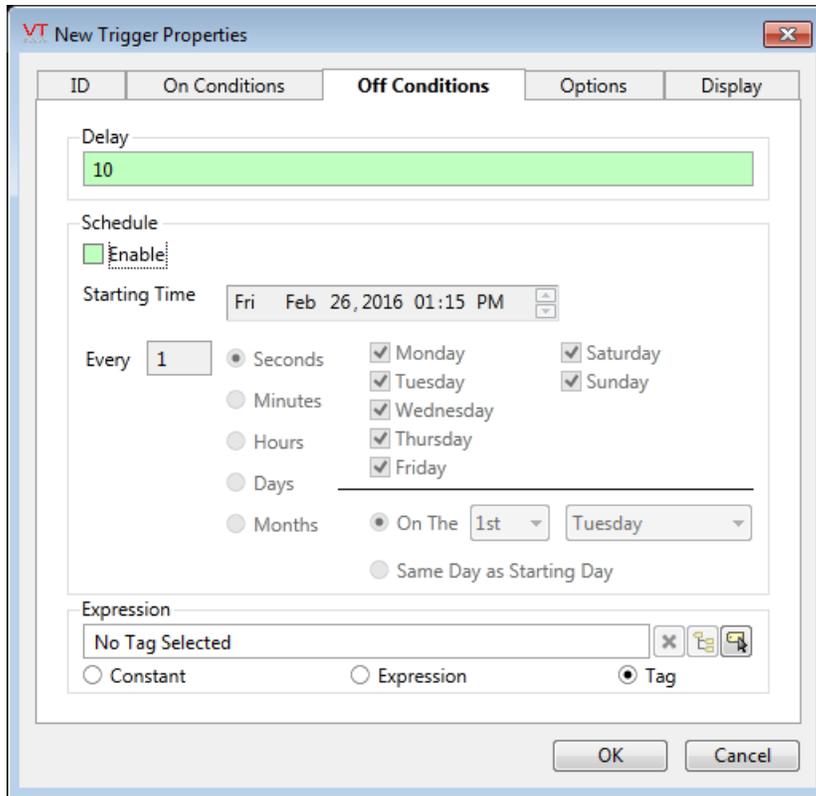
Delay

When set to a value greater than 0, the Trigger tag's value will be set to 0 (OFF) this many seconds after being triggered by the On Condition.

Enable

Must be checked in order for the trigger to de-activate based on a time and date schedule.

Starting Time

Used to set the base date and time for the reset (Off-condition) schedule. The other configuration options on this tab will be counted from the given Starting Time.

Every [N] Seconds / Minutes / Hours / Days

The field beside "Every" controls the frequency of the following units.

For example, if the Every value is set to 1 and the Hours check box is selected, then beginning with the time configured in the Start Time, the trigger will be reset to 0 every hour.

If the Every value is set to 7 and the Days check box is selected, then beginning on the configured Start Time date, the tag will be reset to 0 every week.

Day check boxes

Allow you to control which days the reset schedule will operate on. For example, you may choose to not reset the tag to 0 on weekends by deselecting Saturday and Sunday.

These check boxes will not be enabled when the Months option is selected.

Months

Configures the tag to reset to 0 every month. If this option is chosen, then further refinements include "On The" or "Same Day as Starting Day".

For example, you may wish to configure the trigger to reset "on the 1st Wednesday" or "on the 2nd Monday" of the month. If you have selected "Same Day as Starting Day" then the trigger will be reset on the given numeric day of the month, at the selected time.

When the Months option is selected, the Day check boxes will not be enabled.

Expression

Select a tag, or enter an expression that will be used to reset the Trigger tag's value to 0. The Trigger tag's value will be set to 0 when the value of the selected tag or expression becomes non-zero.

Trigger Tag Properties: Options Tab

A trigger tag may be drawn as a button on a page, thus allowing operators to trigger the associated action at unscheduled times, if required.

By adding an application privilege, you can restrict access, allowing only authorized users to use the button.

Check the Disable option to deactivate the Trigger tag. Disabling a Trigger tag forces its value to 0.
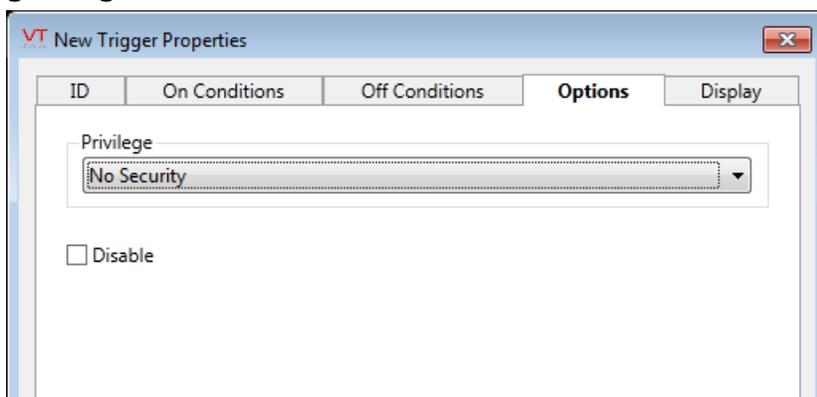
Related Information:

...Refer to the VTScada Admin Guide for "Protect Pages and Output Tags"
– Create your own privileges for protection.

Trigger Tag Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
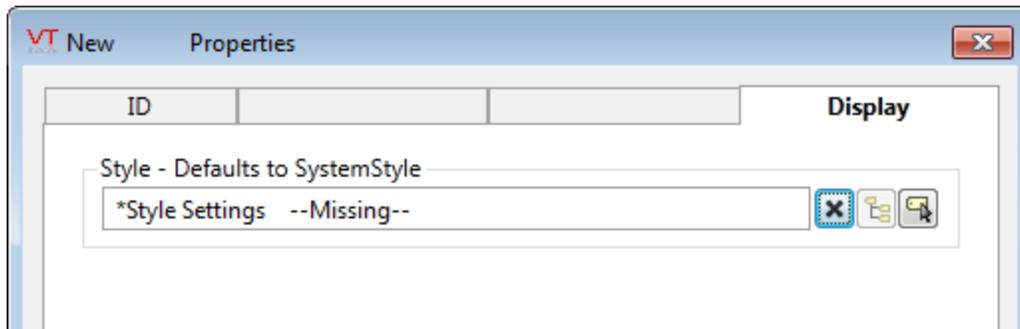
Related:

Style Settings Tags

Trigger Tag Widgets

The following widgets are available to Trigger tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Momentary Button Widget

...Plot Data Widget

...Set Value Button Widget

...Set Value Hotbox Widget

...Text Change Widget

> Note: Of these widgets, only the Set Value Button, Set Value Hotbox Widget, and Momentary Button Widget methods enable users to output values to the associated tag; the remaining widgets display the value of the tag.

Related Information:
Confirmation Prompts for Output Tags - Control options for confirmation prompt text.

Alarm System Tags

This group of tags includes all tags related to alarms that can be used in every VTScada installation.
Tags which are used by the optional Alarm Notification System can be found in the following group, Alarm Notification System Tags.

Related Information:

...Alarm Tags. Create and configure an alarm, to be triggered by another tag.

...Alarm Priority Tags. Control the notification parameters for each priority of alarm. Create customized notifications.

...Alarm Status Tags. Monitor for any alarms matching a configured profile.

Alarm Tags

Not counted towards your tag license limit.
Alarm tags are used to establish rules for when an alarm should be triggered, and what behavior should occur. An Alarm tag monitors the

value of another tag and triggers an alarm when that tag's value reaches or passes a given set point.

The Setpoint property can be a user-defined number or it can be supplied by another tag's value. The two values are compared using an operation such as greater than, equal to, or less than. If the result of this comparison is determined to be true, then the Alarm tag will be triggered.

When the Alarm tag is triggered and while the trigger condition remains true, the value of the Alarm tag itself becomes 1; otherwise, the value of the Alarm tag is 0.

The Alarm tag itself can also serve as a set point object since its value is 1 when the alarm condition exists or 0 when an alarm condition does not exist.

The urgency of the alarm (from Event to Critical) is defined by the Alarm Priority tag selected in the Alarm tag's properties.

Alarms can be configured such that, if they are acknowledged by an operator, but remain active for a set length of time (that is, the triggering condition for the alarm remains in effect), the alarm will go back to an unacknowledged condition, thereby re-activating any audible and visible warning indicators that are attached to that alarm.

Note that shelving or unshelving an alarm is an operational action, not a configuration change.

Note: Alarms should be matched to alarm databases when the application is created. If an alarm is switched from one database to another, you will lose access to that portion of the alarm's history that was recorded using the first database.

The selection of which alarm database will be used is done based on each tag's position in the tag hierarchy. An alarm will be matched to the closest AlarmDatabase tag ancestor in the hierarchy, defaulting to the built-in System Alarm DB if no other database ancestor is found.

Trip Alarms Versus Level Alarms

Alarm tags can be configured as either "Trip" or "Level".

A level alarm will show up in the Active List as long as the underlying trig-ger condition remains true. The underlying condition must be cleared before the alarm is removed from the list of Active alarms.

Trip alarms do not have an Active or Inactive status that is based on the value of the Trigger tag. Once triggered, the trip alarm is simply in an alarm condition, waiting for operator acknowledgment. There is no need to also clear the alarm state by adjusting the Trigger tag's value.

A trip alarm will not be displayed in the list of Active alarms. They are primarily used in situations where the momentary existence of an alarm condition should "trip" the alarm, after which the state of the Trigger tag is irrelevant. As an example, consider an intrusion detection system: When an unauthorized person opens a door, the alarm is triggered. What later happens to the door is irrelevant to the alarm.

Both trip and level alarms must be acknowledged by an operator before their audible and visual warnings will stop.

Alarm Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Alarm Properties: Trigger Tab

The Trigger tab for the Alarm tag properties folder consists of the attributes used to identify the conditions that the Alarm tag will evaluate to determine whether an alarm should be triggered.

The Triggered By field is used to select the tag that is being monitored by this Alarm tag. The Setpoint property displays a value (or a tag) that will be compared to the value of the tag being monitored (as identified in the Triggered By tab). The Function property selects the operation that is to be performed to compare the monitored tag's value with the Setpoint value.

Triggered By

Select the tag whose value will be monitored and compared to the value of this Alarm tag's Setpoint value to determine if this alarm should be triggered.

The Triggered By field can be used to associate this Alarm tag with a new or existing numeric tag using the tag browser but-

ton to its right. The Triggered By field can be cleared of its current value using the X button to its right.

Right-clicking the name of the I/O tag that has been selected in the Triggered By field opens the tag properties folder for the selected communication driver tag.

Function

The Function drop-down list enables you to select one of 11 operators to use to compare the value of the Triggered By tag with the Setpoint value configured for the Setpoint field to determine if there is an alarm state.

Setpoint

You must provide a setpoint in order to configure an alarm. The alarm will trip or activate when the triggering value matches the setpoint, according to the rule set by the selected function.

The valid value for the Setpoint field can be provided via any of a constant, an expression, or a tag. Please see Constant, Expression or Tag for help selecting which to use.

Deadband

The Deadband field specifies how far the triggering value must return into the safe zone before the alarm is no longer considered active. Deadband values are used in systems where analog values fluctuate often, sometimes providing a false data reading.

On Delay

Use the On Delay field to specify a length of time (in seconds or in fractions of a second) that the system will wait before

triggering this alarm. This enables you to disregard transient spikes in value as the tag must remain in an alarm state for the amount of time specified in the Delay field before an alarm will be indicated.

Off Delay

Similar to the On Delay, the Off Delay field can be used to specify a length of time (in seconds or in fractions of a second) that the system will wait before the alarm will return to the normal state. (The triggering condition is no longer true). This enables you to disregard transient spikes in value that may occur after an alarm has been activated. The Off Delay does not apply to trip alarms, which do not have an active state.

Alarm Properties: Actions Tab

The Actions tab for the Alarm tag properties folder enables you to select an Alarm Priority tag whose of properties will be used to indicate the category of alarm to be triggered by this Alarm tag (see Alarm Priorities). Other properties available on the Actions tab allow you to disable the alarm in the event that alarm conditions could be triggered due to routine equipment maintenance or another interruption in communications of which you are aware. The sound to be used for the alarm siren for this alarm can also be indicated here.
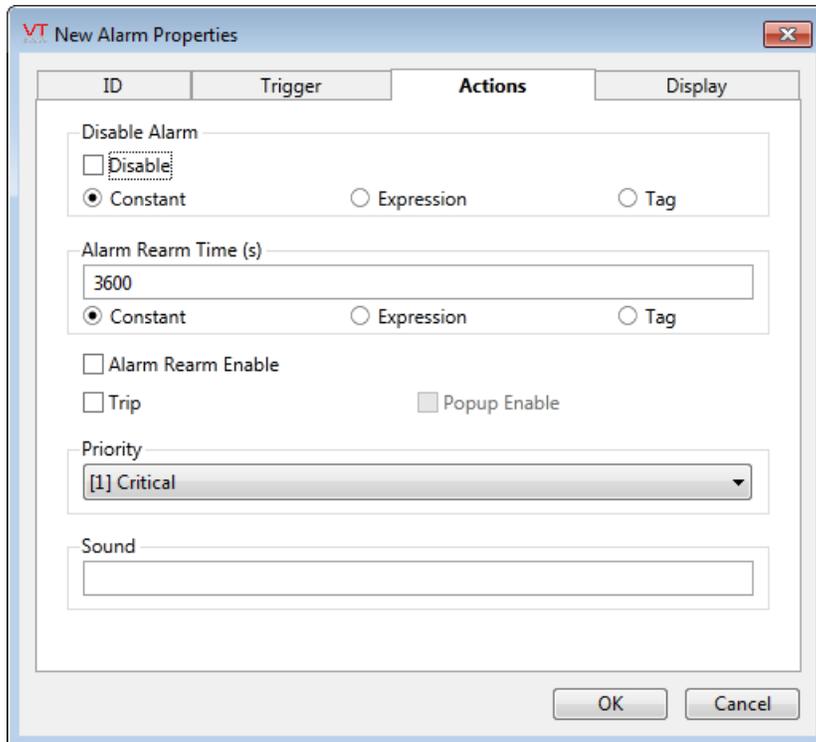
Disable Alarm

The Disable Alarm field enables you to disable or enable this Alarm tag, using a 0 (not disabled) or 1 (disabled), an expression or, the value of a second tag. This feature is typically used in situations such as when routine maintenance is being performed. In such a situation, an alarm can be disabled until the maintenance is complete and communications are reestablished, thus avoiding false alarms.

Alarm Rearm Time

Applies only if Alarm Rearm Enable is checked for this alarm. If the alarm has been acknowledged, but remains active for the time shown in this field, the alarm will return to the unacknowledged state. Audible and visible warnings configured for this alarm's priority level, will again be displayed. The Alarm Rearm Time is measured in seconds, and defaults to 3600 (1

hour). The value in this field must be greater than 0.

Alarm Rearm Enable

Controls whether this alarm will revert back to an unac-
knowledged state if it remains active for the length of time set
in Alarm Rearm Time, after having been acknowledged by the
operator.

Trip

Choose whether this alarm is a trip alarm or a level alarm, as
described in the introduction to this tag.

Priority

Set the priority of this alarm by selecting an Alarm Priority tag.
The built-in Alarm Priority tags have numbers from 0 to 4 with
the following meanings:

- None.
- 0 (event) – the alarm will not occur, but its value may still be used
  as a set point by other tags;
- 1 (critical) – the highest priority alarm;
- 2 (high) – a regular priority alarm;
- 3 (warning) – warning
- 4 (notice) – a notice or informational event. Logged in the event
  history, but otherwise silent.

You may also create your own Alarm Priority tags.

Sound

Identify what audible warning will be played when this alarm is
triggered. The Sound field can be set to a 0, a 1, or to the
name of a .WAV sound file to be played.

If the Sound field is set to 0, no sound will be played when this alarm is triggered.

If the Sound field is set to 1, the sound configured in the associated Alarm Priority tag will be played.

If the Sound field identifies the name of a .WAV sound file, it will override any alarm sound configured for the associated Alarm Priority tag. When specifying a sound file, you must enter its name and extension (e.g. MySound.wav). The specified sound file must be a .WAV file, and must be stored in the application directory. If the specified sound file is not found, the alarm will revert to using tones as specified in the associated Alarm Priority tag.

Pop-up Enable

If the configuration variable AlarmPopupsEnable is set to 1, then Pop-up Enable will result in a pop-up dialog being displayed whenever the alarm is triggered. It is strongly suggested that this feature be used sparingly.

The pop-up will display the names of all active alarms for which Pop-up Enable has been set. As alarms are acknowledged, they will be removed from the pop-up. Should all the alarms be acknowledged or go inactive while the pop-up is displayed, it will close. You may close the pop-up at any time by selecting the Close button. The pop-up will not be displayed again until a fresh alarm event occurs.

The pop-up does not provide a way to acknowledge the alarm. Its purpose is solely to provide extra notification to the operator of current, unacknowledged alarms.

Alarm Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag

instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
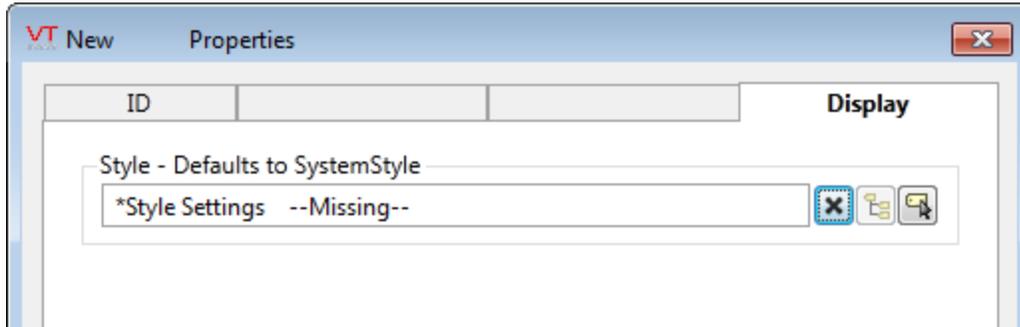
Related:

Style Settings Tags

Alarm Tag Widgets

An alarm tag has a value of zero or one, depending on whether its trigger is in an active state (one for active). This is true even of trip alarms. The alarm tag may therefore be drawn as if it were a digital input. The following widgets are available to Alarm tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Blink Widget

...Color Box Widget

...Color Line Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Text Change Widget

Related Information:

...Alarm Priority Tags

Alarm Database Tags

Not counted towards your tag license limit.

The Alarm Database tag type is used to link alarms and events to Historian tags. They are also used to preselect the columns that will be shown when viewing the alarms from a particular database in the Alarm Page.

Every VTScada installation comes with two Alarm Database tags: System Alarm DB and System Event DB. The term "System Event" refers to the VTScada event-level actions that will be recorded using the System Event DB tag. This includes operator actions such as logging in and changing output tag settings. It also includes events such as report generation using a Report tag.

Alarms are associated with their closest Alarm Database tag ancestor in your tag hierarchy. Any alarm that does not have a specific Alarm Database tag as an ancestor will be associated with the System Alarm DB. This includes user-defined alarms that are given priority 0 – Event. Only VTScada events are logged with the System Event DB tag.

If you decide to create your own Alarm Database tags, and have ensured that all alarms are child tags of those, then you might choose to disable the System Alarm DB tag. The System Event DB tag should not be disabled under any circumstance.

Developers might choose to create more Alarm Database tags for any of the following reasons:

- Security across realms.
  Realm area filtering can be applied so that an operator can be restricted to see only the alarms from a single alarm database (or set of databases). All other databases could be hidden from this operator.
  Note that it is not a requirement that you create multiple alarm databases in order to use realm area filtering.
- Management of alarms.
  If operators in different roles or locations are required to look after a certain subset of alarms, those alarms could be kept in a separate database so that

no filtering is required in the alarm list beyond the selection of which database's alarms to show.

- Efficiency.
  It can be slow to filter alarm history if there are millions of alarm records in a database. if the alarms are distributed between multiple databases then each will be smaller, thus making it faster to filter alarm history.
- Management of large, widely distributed systems.
  If two or more plants are part of the same application, and those plants are separated by a slow network link, then you can improve local performance by setting up a Historian-AlarmDatabase pair for each plant, where each plant will be use local server(s) for its primary server.

Switching Databases

There is no restriction on moving an alarm from one database to another, but there are consequences:

> Warning: If the old and new Alarm Database tags use different historians, *do not proceed*. You will lose alarm history.

If the alarm is moved from one db to another while the application isn't running, the alarm will be temporarily orphaned on the old database once the application is started. This could result in duplicate alarms on the alarm page and may result in an alarm activation not being logged in the new db. The orphaned alarms will be purged automatically within a few minutes of application start, as controlled by the property, AlarmDatabasePurgeDelay. See: Orphaned Alarms

To move an alarm to a new database, move the tag to become a child of the new Alarm Database tag. A decommission event will be registered for the old database, and a commission event will be registered for the new. Note that every Alarm Database tag will have its own child Notebook tag, which will be created automatically. All notes created for an alarm are stored in the notebook associated with the matching alarm database.

Related Information:

...Realm Area Filtering

...Historian and Logger Configuration

Alarm Database Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

    Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

    The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

    When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

    To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).
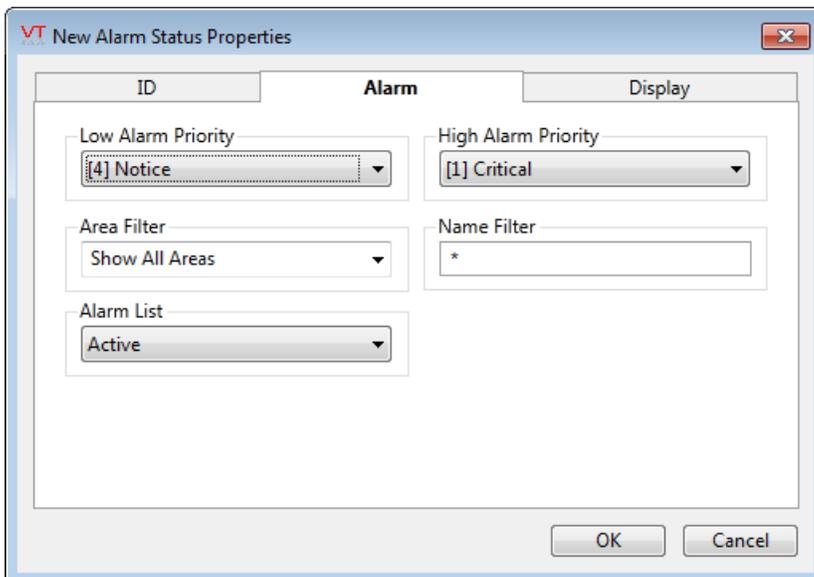
Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Alarm Database Properties: Settings Tab

Use the Column Format selector to choose how alarms in this database should be displayed on the Alarm Page. Your options are as follows:

Column Format	Displayed columns
Alarms	Time – Priority – State – Event – Area – Name – Description – Value – Setpoint – Units – Operator – Notes
Events	Time – Area – Name – Description – Workstation – Device – Operator – Notes
Legacy 1	Event – Priority – Time – Area – Name – Description – Operator
Legacy 2	Time – Name – Description
Legacy 3	Time – Ack – State – Event – Priority – Area – Name – Description – Operator
Legacy 4	Time – State – Event – Priority – Area – Name – Description – Operator
Legacy 5	Ack – State – Event – Priority – Area – Name – Description –

	Operator
Legacy 6	Time – Event – Priority – Area – Name – Description – Operator
Legacy 7	Time – Name – Description – Ack
Popups	Priority – Ack – Name – Description

Alarm Database Properties: Historian Tab

Historian

Note: Do not use the same Historian for both alarm history and tag data.

An Historian tag must be selected in order for this alarm database to save alarm history. Historian configuration and advanced logging options are described in the discussion of the Historian Tags. By default, the System Alarm Historian is selected.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Alarm Priority Tags

Not counted towards your tag license limit.
The Alarm Priority tag type is used to specify different alarm indicators such as the frequency of the alarm tones, the tone cycle, the priority number, and the symbol and highlight color used to represent alarms of this priority. All alarms are associated with an Alarm Priority tag so that when the Alarm tag is triggered, the sound and color indicators configured for the Alarm Priority tag with which the alarm is associated will be activated.
Every standard VTScada application includes five Alarm Priority tags , as described in the following table:

(Tone Cycles: 0 means no tone, positive values set the number of repetitions. All priorities can have two tones, each of which can be configured for frequency (pitch) and duration.

All alarm priority tags are part of area, "System". The default duration for all tones is 0.5 seconds.

Name	Descriptor	Priority #	Color	Symbol	Tone 1 Freq	Tone 2 Freq	Tone Cycles
AlarmPriority0	Event	0	White	none	0 Hz	0 Hz	0
AlarmPriority1	Critical	1	Red		2000 Hz	1000 Hz	unlimited
AlarmPriority2	High	2	Orange		1800 Hz	1000 Hz	unlimited
AlarmPriority3	Warning	3	Yellow		1600 Hz	1000 Hz	unlimited
AlarmPriority4	Notice	4	Cyan		1400 Hz	1000 Hz	0

Alarm Priority Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Alarm Priority Properties: Display Tab

The Display tab for the Alarm Priority tag properties folder sets the priority number of the alarm and holds the properties used to help operators visually identify the priority of alarms associated with this tag.
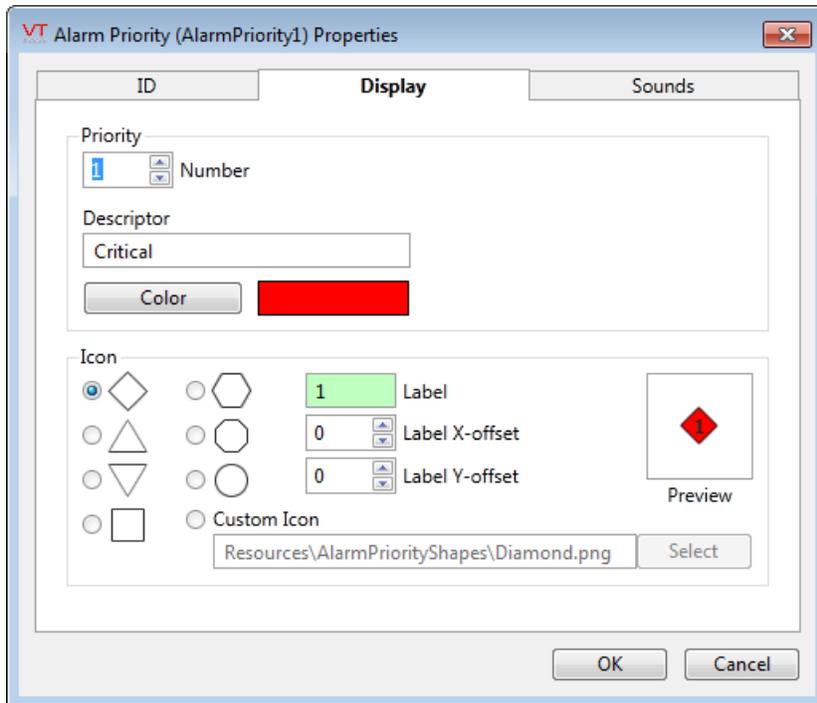
Priority

The Priority field sets the priority of this alarm as a number from 0 to 4, where:

- 0 indicates an event rather than an alarm. Events are recorded in the history, but require no operator response.
- 1 indicates the highest priority alarm, usually described as "critical".
- 2 indicates a regular priority alarm, usually described as "high".
  Only priority 1 and priority 2 alarms will be sent to the alarm notification system for dial-out, email or SMS text notifications.
- 3 indicates a warning.
- 4 indicates a notice or informational event

You may set the Priority property to values higher than 4 for user-defined alarm priorities.

Priority Descriptor

Use this field to enter a short text description that will be used to identify the priority of those alarms associated with this Alarm Priority tag. This description will appear when filtering drop-down lists, and in the alarm list, wherever alarm priority is referenced.

Color

When clicked, the Priority Color button opens the Select Color dialog from which you can select the color you wish to be used to highlight alarms of this priority in the alarm list (thereby helping operators to identify the priority of the alarm visually.

Icon

Select any of the preset icons, or choose a custom icon. This will be used in alarm lists, and by the Alarm Priority Icon and Alarm Priority Box widgets to help operators identify the priority of alarms.
If you decide to use a custom icon, the Select button will open the VTScada graphics library. Select any existing image, or import a new graphic for use.

Icon Label

Sets the text to be displayed within the icon. By default, this will match the priority number.

Icon Label Offsets - X and Y

Use these to adjust the location of the label within (or outside)

the icon. An offset may be required for custom icon images in order to ensure that the label is clearly visible. Offsets are measured in pixels.

Alarm Priority Properties: Sounds Tab

Use this tab to control the sound that will signal alarms with this priority. You can define a tonal sound, enable speech, use any .WAV format file from disk, or set a combination. These settings do not apply to the voice calls made by the VTScada Alarm Notification system.

Sound Cycles

Specifies the number of times that the tonal or .WAV file alarm notifications will repeat. The default is -1, signifying unlimited repetitions until the alarm is silenced or acknowledged.

Speech - Enable Alarm Speech

> Note: Before enabling speech for this priority tag, you must open the Application Configuration dialog, select the Alarms tab of the Edit Properties page and there select, "Speak alarms instead of playing sounds".

A check box to enable or disable spoken alarms. Please refer to Spoken Alarms on the Server or Workstation for the instructions to make this feature work on your system.

Speech cycles

Sets the number of times that the message will be repeated.

Tone Frequency in Hz and Tone Period

The VTScada alarm sound consists of two high-pitched tones, alternating within a specific time period.

For all Alarm Priority tags, the default frequency for the *second* tone is 1000Hz. The frequency of the first tone varies from one Alarm Priority tag to the next so that each priority will have a distinct sound. Adjust tone and frequency as desired to create a distinctive sound.

.WAV File Name

The .WAV File Name field enables you to specify the name and extension of a .WAV sound file to be used in place of the typical alarm tones to signal to operators that an alarm has been triggered. A variety of sound files are included with VTScada. If a .WAV file is specified, the Sound Cycles field controls the number of times the .WAV file is to be played once an alarm has been triggered.

When specifying a sound file, you must enter its full name and extension (e.g. MyAlarmSound.WAV) . VTScada searches for the specified .WAV file in your application directory if no path is provided. If the .WAV file is not found, the alarm will revert back to using the tones as specified in the Tone 1 Frequency in Hz and Tone 2 Frequency in Hz fields.

Related Information:

...Alarm Tags

...Spoken Alarms on the Server or Workstation

Alarm Status Tags

Not counted towards your tag license limit.
This tag looks for active or unacknowledged alarms matching a given set of characteristics. It can be used to build a trigger for an output or another alarm. For example, you might use it to trigger a warning system when any alarm with a given priority in a given area is active.
The value of the Alarm Status tag will be either 1 or 0 depending on whether any matching alarms have been triggered.

Alarm Status Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Alarm Status Properties: Alarm Tab

The alarm tab is used to define which alarms will be counted an instance of the Alarm Status tag. Use the fields to select the characteristics that must be found in an alarm in order to be included in the count.

Low Alarm Priority

Select the lower end of the range of alarm priorities to be filtered for. Any alarm with a priority between the Low Alarm Priority and the High Alarm Priority, and matching the remaining conditions will be included in the filter. The minimum priority cannot be set above the maximum priority.

To filter for alarms of only a given priority, set both the Low Alarm Priority and the High Alarm Priority to that value.

High Alarm Priority

Select the upper end of the range of alarm priorities to be filtered for. Any alarm with a priority between the Low Alarm Priority and the High Alarm Priority, and matching the remaining conditions will be included in the filter. The maximum priority cannot be set below the minimum priority.

To filter for alarms of only a given priority, set both the Low Alarm Priority and the High Alarm Priority to that value.

Area Filter

Select the area that the alarm must be configured within in order to be included in the filter. Areas shown in the list include all system alarm areas as well as developer-created areas.
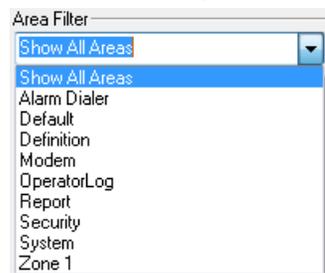
Name Filter

Use this field to restrict the list of filtered alarms to only those matching a given name. The * wildcard may be used to permit selection on a set of similar names. For example, "East*" will match "East Pump Alarm", "East Tank Alarm", etc.

Alarm List

Options include "Active", "Unacknowledged" and "Current". Only one option may be chosen. Restricts the filter to only those matching a particular list type.

Alarm Status Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a

Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
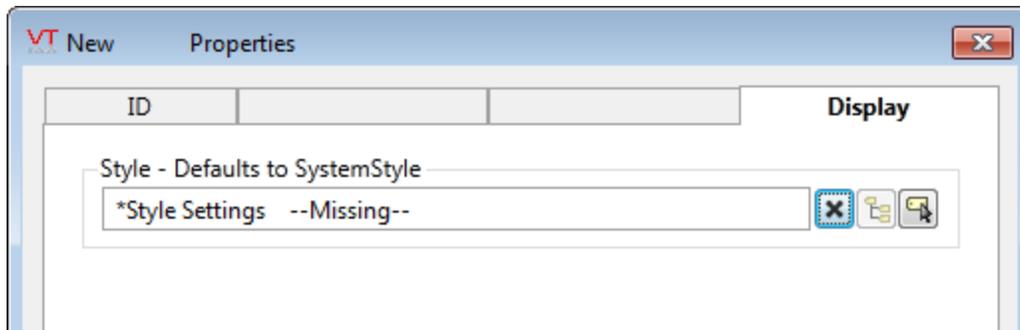
Related:

Style Settings Tags

Alarm Status Tag Widget

The following widget is available to display information about your application's Alarm Status tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
Compass 1 Widget	Compass 2 Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Droplist Control Widget	Equipment / Status Color Indicator
Image Change Widget	Indicator Light Widget
LCD Bar Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget

Glass Gauge Widget	Classic Gauge Widget
Retro Gauge Widget	Cockpit Gauge Widget
Aqua Gauge Widget	Army Gauge Widget
Backlit Gauge Widget	Nautical Gauge Widget
Quarter Arc Gauge Widget	Linear Gauge Widget
Antique Gauge Widget	LCD Ring Widget
LCD Arc Widget	LCD Meter Widget
Momentary Button Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Rotary Control Widget (Knob)
Set Analog Value Widget	Set Value Button Widget
Set Value Hotbox Widget	Slider Widget
Text Change Widget	Top Bar Widget
Two Color Bar Widget	

Alarm Notification System Tags

The VTScada Alarm Notification System contacts designated operators by phone, pager, or email when alarms have gone unacknowledged for a user-defined period of time. This is an optional component of your VTS license.
Tags used by the Alarm Notification System include:

Related Information:

...Modem Tags. Used to configure the modem over which voice calls will be made.

...Roster Tags. Used to designate who will be contacted for alarms in a given area, and the method of contact for each user.

...SMS Appliance Tags. Used to configure the SMS device over which text messages will be sent and received.

Modem Tags

Not counted towards your tag license limit.

Modem tags enable outgoing and incoming calls by providing a link between your VTScada application and a modem configured on your system. This is not the same as configuring a Serial Port that uses a modem for communication.

If configuring a modem for use with the Alarm Notification System, you must use a voice modem. See the links at the end of this topic for further information.

Your choice of Area property for a Modem tag will affect how it works. When used with the Alarm Notification System, the area property of a Modem tag must be left blank.

If your application uses one modem for I/O communication and another for the Alarm Notification System, then you should set the area property of the data modem to match the area property of the port. You must also add the application property, "UseSerialAreaInModemCall", and set it to 1. This will force all data calls to use the modem having an area that matches that of the attached serial port.

You can monitor the call status by using a Modem Indicator widget to display the modem tag's value. This will be a numeric value, with meanings according to the following table.

Value	Meaning
0	Modem idle
1	Modem calling
2	Modem answering
3	General failure (other than listed here).
4	Server is unavailable
5	Workstation that owns the modem is unavailable.
6	Cannot use serial port due to an external condition. The modem will poll occasionally for revised status. The polling time is set by MMUnAvailRetry, and defaults to 60 seconds.
7	Manually disabled – will not clear automatically.

Modem Properties: ID Tab

Note: The Area property for a voice Modem tag must be left blank.

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags

(note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added.

To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Modem Properties: Settings Tab

Warning: All fields are in this tab are mandatory.

The Settings tab of the Modem tag properties folder contains properties that allow you to configure the network and security settings for the Modem tag. A message is provided to the right of the workstation name, provides extra information about the station's availability, as follows:

If the Workstation field is left blank, the modem tag is for a so-called, local modem, and the text "Local" will be displayed in the workstation message field. This configuration is not recommended – it is much better to specify the workstation name even if VTScada is running only on a single workstation where the modem also resides. (The term "Local Modem" does not mean simply that the modem is on the same machine.)

The "Index" parameter (the modem name) for the modem tag is ignored

if this modem tag is for a local modem. Local modems are specified by a configuration setting in the <workstation>.Startup file for each workstation as follows:

```
<Modem tag name>Device = <Modem friendly name>
```

For example:

```
Modem1Device = MultiTech Systems MT9234ZPX-UPCI [Trihedral]
```

If this setting is not provided, the modem manager will select a modem from the list of modems not already associated with a modem tag. (Results may be unexpected.)

Local modems are typically used only if PPP is being used by VTScada, or in a system with custom software designed to use local modems. Under typical operating conditions, Trihedral recommends that local modems NOT be used.

The text, "Available" or "Unavailable" in the workstation message field refers to the workstation's availability. It is displayed even if no modem has been selected.
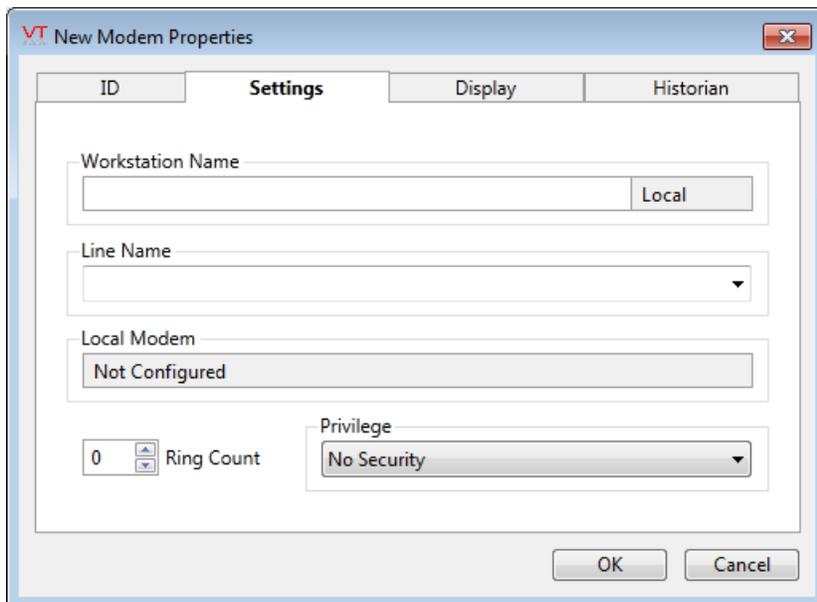
Workstation Name

The Workstation Name field enables you to specify the name of the computer to which the modem is connected. If you do not specify a workstation in the Workstation Name field, VTScada will use a local modem on a machine that

is running that tag within the application, selecting the first modem that is available.

> Note: Once the workstation name has been specified, the Line Name and Name Or Index On Workstation drop-down lists will automatically update to display the names of the Serial Port tags and modems available on this workstation.

Line Name

The Line Name drop-down list enables you to select the name of an existing phone line to which this modem is attached, or enter a new and unique line name to which this modem is attached. This name must be unique, and cannot be identified anywhere else in the system.

The same line name must be used by modems sharing the same physical phone line, even if they are on different workstations.

Name Or Index On Workstation

The Name Or Index On Workstation drop-down list enables you to select a modem on the workstation specified in the Workstation Name field, provided that the workstation is available. If the specified workstation is not available, you may enter the name of the modem you wish to use, or can enter a number or text value as follows:

If this value is numeric, it is the index into the list of modems shown in the Modems Properties dialog box in the Windows Control Panel, starting from 1. If the value of Index is a text string, it is the name of the modem exactly as it appears in the Modems Properties dialog that is accessed using the Windows Control Panel.

> Note: Because the index of available modems can change over time, it is recommended that you refer to the modem by name rather than by its index in the Modem Properties dialog box.

Ring Count

The Ring Count spin box enables you to select the number of rings you wish to occur before the selected modem answers an incoming call.

If the value of Ring Count is 0, the modem will not answer any calls.

> Note: Regardless of this setting, the modem will not be answered if the application property, AnswerCalls is set to 0. Information on

Security Privilege

> Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.
>
> In this case, "operation" refers to the ability to disable or enable the tag via the context menu. It does not affect whether or not the modem will operate while a user without the privilege is logged on.
>
> Application privileges are added using the Administrative Settings security dialog. Information on creating application privileges can be found in "Add Application Privileges Using the Administrative Settings Dialog".

Modem Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
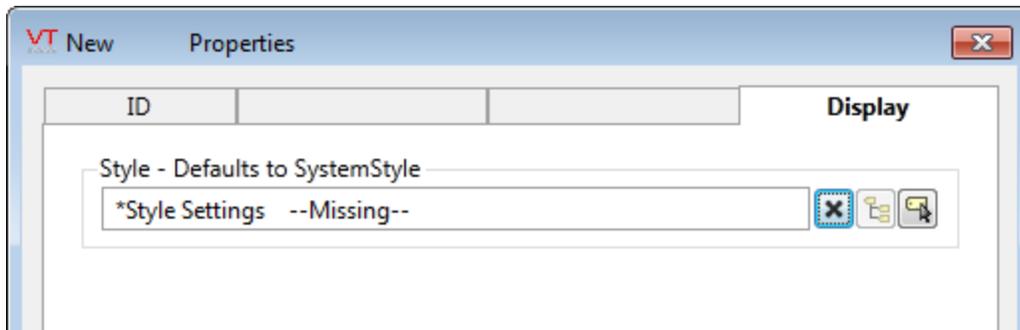
Related:

Style Settings Tags

Modem Properties: Historian Tab
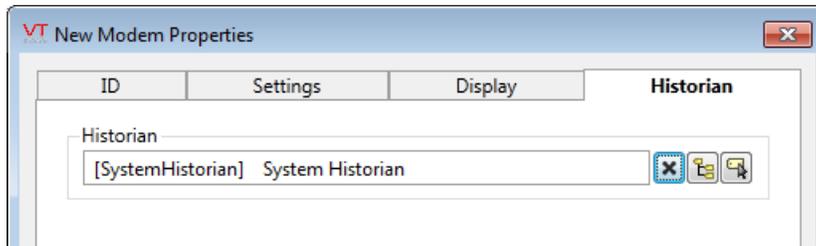
Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Modem Tag Widgets

The following widgets are available to display information about your application's Modem tags:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...LCD 5x7 Matrix Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...LCD 7 Segment Widget

...Modem Indicator Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Text Change Widget

> Note: VTScada also includes a Modem Tools folder within the Widgets palette, which contains modem diagnostics tools that may be added to a page.

Roster Tags

Not counted towards your tag license limit.

> Note: Contacts cannot be added to Roster tags before you enable at least one notification method. A voice modem or an email server or an SMS appliance must be correctly configured before you attempt to configure a roster.
> Further to this, no contact method can be selected unless VTScada is able to determine that it is available for use.

> Note: On a workstation with a run-time license, you must possess the Manual Data security privilege in order to add, change or remove roster entries.

Roster tags are designed to work with the Alarm Notification call-out system. Using a Roster tag, you may configure contact information for up to 30 operators. When a Roster tag has been enabled, the call-out system will contact the operators in order in the event that an alarm has been triggered and has gone unacknowledged for a user-defined period of time. The call-out system will continue to contact the operators on the active Roster tag until the alarm has been acknowledged.
The Area field, selected during configuration, is especially important for Roster tags. The Area field is used by VTScada to tie the Roster to those alarms that have this same area. If more than one Roster shares the same area, only one of them can be active at a time. The active Roster Tag in an area is always the one most recently activated – all other Roster tags with the same area property are automatically deactivated when one is activated.

One Roster Tag, the Default Call-Out Roster, is configured by default in every VTScada system. No names should be added to this roster. Its purpose is to provide a means of telling VTScada to not call anyone. Since only one roster in an area can be active at a time, if the roster that is active has no names attached, no one will be called.

Roster tags work with Modem tags, and the Alarm Notification System Options and Speech Lexicon tools to form the VTScada Alarm Notification System.

Note: Note: Whether or not existing alarms are sent to a roster as it becomes active depends on the configuration of the rosters.

Alarms are not transferred: Given two rosters, the first having no contacts (since operators are assumed to be on-site) and the second having a list of contacts for when operators are not on-site. As the second roster becomes active, existing alarms are not sent to it.

Alarms are transferred: Given two rosters, each of which has a list of contacts. As the second roster becomes active, existing alarms that were being called out to the now-inactive roster will switch to the currently active roster.

Note: If your system includes more than one modem, and those modems are attached to servers located in regions with different area codes, then you must use the Canonical Address Format for each phone number and you must configure the server's location within the Windows™ Phone and Modem Settings dialog.

In the event that you must configure a Roster tag to include the prefix "9" from some workstations, but not others, you should use the Modem Settings dialog in the Windows Control Panel.

On a computer that requires the 9:

1. Open the Control Panel.
2. Open the Phone and Modem Options dialog.
3. Select the Dialing Rules tab.

4. Click, Edit.

5. Ensure that the area code is set.

6. In the appropriate boxes, enter the number "9".

7. Click Apply and close.

8. In the Roster configuration, ensure that you use the canonical form of the phone number.

Note: If the communication method you want to select is not enabled: Communication methods are enabled only when VTScada is able to use those methods. For example, if you have not attached an SMS device and configured an SMS Appliance tag, then you cannot choose to send an alarm message by SMS text.

Roster Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Roster Properties: Contacts

The Contacts tab for a Roster tag is used to create the list of people who will be notified, and to define how to reach each of them. The available methods include voice (a phone call) email, pager, SMS text, or SMS text as an email via your cellular provider.

Note that the choice of communication methods controls the label and purpose of the next few fields, as shown. (Voice and SMS Text will have identical fields, as will Email and Compact Email.)

Contact List (Contact Order | Username | Mode | Enabled)

The list displays the operators who will be contacted in this roster, how they
will be contacted and whether that entry is enabled. Up to 30 names may be
added. The list is not interactive . You cannot change the contact mode or
enabled status by clicking on in the list. Instructions are provided later in this
topic.

Blank lines are permitted, but not recommended since they will make the list
more difficult to work with.

The list may be reorganized using the Move Selected buttons.
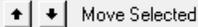
Communication Method

The Communication Method radio buttons enable you to select the method
of contact for this operator record. Options include phone/voice, pager, or
email.

Continue Roster Sequence

This option does not apply to voice contacts.

VTScada will assume that a message sent by pager, email or SMS-text will be
delivered (the equivalent of a phone call being picked up), and will therefore
not continue to the next name in the list.

Use the Continue Roster Sequence check box to command the system to con-
tinue to contact operators in the order they appear on the roster even though
the email or other message was sent.

The roster will continue contacting entries flagged as "continue roster
sequence" until the alarm is acknowledged. The time between each attemp-
ted contact is controlled by the application property, RosterDelay. If the
roster fails to send a notification to a contact, no delay is used before attempt-
ing the next (if any).

After calling a Voice contact, the roster will automatically call the next contact if the last was not successfully contacted and the alarm has not yet been acknowledged. Once the alarm is acknowledged (by either the person on the phone or by an operator at the workstation) the roster sequence will terminate until the next alarm goes active.

Do Not Contact

The Do Not Contact check box enables you to temporarily disable an operator's record in the roster. In the event that an alarm is triggered and goes unacknowledged, this operator will be skipped by the roster/dial-out system until the operator's contact information has been enabled.

Contact Description

Used to identify each contact in the list. This is an optional field. If left blank, the Phone Number or email address of the contact will be shown.

Contact User Name

Used if alarm acknowledgment by email or SMS text is configured and if the Communication Method is set to email or SMS text for this contact. If left blank, a notification will be sent, but that message will not include an acknowledgment code.

If set, the field must match an existing security account name.

Note: Realm Area Filtering will affect how you configure the roster so that users can acknowledge alarms via email or SMS-text message. Each contact user name in the roster (other than the admin account) must include the full group qualifier.

Phone Number / Email addresses

Voice lines: Enter the contact's phone number, using only the digits. If your phone systems requires a 9 to access an outside line, you will need to configure your modem options using the Windows™ control panel.

Note: If your system includes more than one modem, and those modems are attached to servers located in regions with different area codes, then you must use the Canonical Address Format for each phone number and you must configure the server's location within the Windows™ Phone and Modem Settings dialog.

Pagers: The Phone Number field requires the paging company's T.A.P. Modem Number (TAP = Telocator Alphanumeric Protocol). This number enables access to the alphanumeric paging network, enabling VTScada to send a message to the pager phone/PIN. The correct number for the Phone Number field can be obtained from your paging company's technical support team.

Email addresses: If the selected contact mode is email, this field will be used for email addresses rather than phone numbers. You may enter several addresses, using a semi-colon between each.

While there is no limit on the number of email addresses you may have, the combined total number of characters may not exceed 255.

Pager PIN

This field is used only with the pager communication method. A PIN is a required part of the pager contact information.

Time Zone

If one or more of the contacts are located in a time zone other than that of the alarm server, they may find it inconvenient or confusing to translate the time stated in the alarm notification to their local time. If the contact's time zone is provided, using this drop-down list, then they will receive a message that has been adjusted to their location for them.

The list uses the names built into the Microsoft Windows' Multilingual User Interface Pack.

To add contact to the list:

1. Click on a blank row in the list.
   To keep the list tidy, you should select the first blank row from the top. The row will be displayed with blue highlighting.

2. Select the communication method.
   Voice, Pager or Email

3. Enter the name of the operator in the Name field.
   As soon as you press enter or exit from the field, the name you typed will be added to the list.

4. Enter the phone number or email address (as per the communication method) for the contact.
   Email addresses may be in the basic form, Name@Company.Com" or

you may use either of the following longer forms. Multiple addresses may be entered, separated by semi-colons or commas. Do not add spaces within the angle brackets.

"Full Name" <Name@company.com>

Full Name <Name@company.com>

5. [Only if the contact mode is "Pager"] Enter the PIN number for the pager.

To update a contact in the list:

1. Click on the contact entry that you want to modify in the list.

2. Change any of the fields that need to be updated.
   Changes to the row take effect immediately on pressing <enter> in each field, or upon leaving the field.

To delete a contact from the list

1. Click on the contact entry that you want to delete from the list.

2. Clear the Name field

3. Clear the Phone Number/Email address field.

When both the name and the phone number fields have been made blank, the entry will be removed from the list.

Roster Tag Message Dialogs

If an operator's contact mode has been set to email and a badly formatted email address has been entered, then the following warning dialog will be displayed:
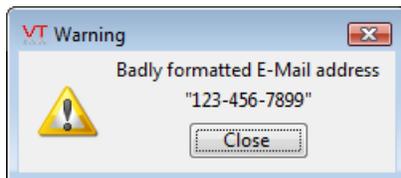
Click on the Close button and correct the email address before proceeding. The Roster tag will not permit you to save a badly formatted email address.

Roster Properties: Activation Tab

Optionally, define an external trigger that will activate a roster. A roster can also be activated manually by adding a roster button to the CallOut

List page of your application. (See: Roster Tag Widgets.)

Note that, given several Roster tags that share a common area, only one can be active at a time. Activating one Roster tag will automatically cause all other rosters that share the same area to be de-activated. There is no such thing as telling a roster tag to de-activate. This happens automatically when another roster tag in the same area activates. A roster tag remains active until another in the same area activates.

If you accidentally use the same trigger for two Roster tags that share an area, that trigger will be ignored since it would violate the "only one active roster at a time in an area" rule.
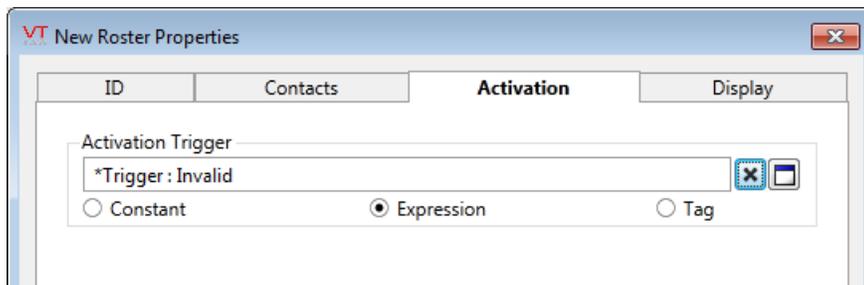
Manual activation of the roster will over-ride its current status.

Roster Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
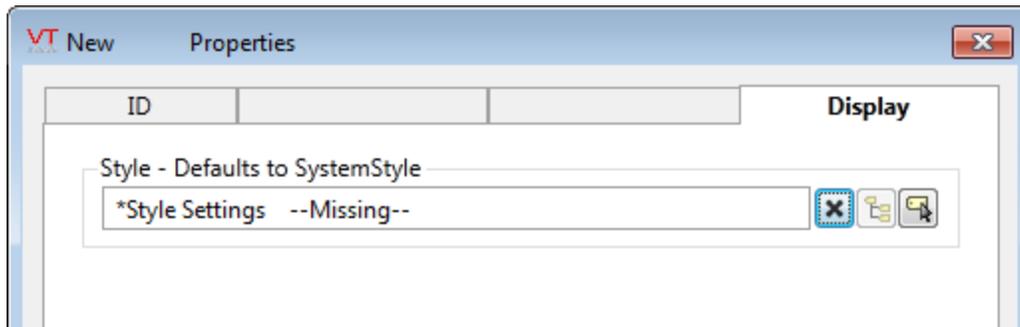
Roster Tag Widgets

The following widgets are available to display information about your application's Roster tags:

...Color Box Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Make Active Widget

...Roster Alarm Test Widget

...Text Change Widget

Related Information:

Alarm Notification System – Alerts by Phone, Email or SMS-Text –
Describes the overall Alarm Notification System.
Roster Selection Hierarchy – Which roster is used for which alarms.
SMS Appliance Tags – Required for the SMS text notification option.
See: "Alarm Notification System Properties" in the VTScada Admin Guide

SMS Appliance Tags

Not counted towards your tag license limit.
The SMS Appliance tag works in conjunction with the Alarm Notification System option to allow alarm notifications to be sent as text messages to a mobile cellular device, and to allow alarm acknowledgment via SMS. SMS messages are sent and received using an "SMS appliance," which is a cell phone (or "cell modem") tethered to a VTScada workstation via RS-232, Bluetooth, or USB. Any such "cell modem" should conform to ETSI TS 100 585, which is a part of the GSM standard applying to the AT command set for sending and receiving text messages. Devices that use the CDMA standard are not supported.

The SMS Appliance tag handles communications through the SMS appliance. The tag registers with the SMS Manager service as an SMS agent. Since SMS appliances are usually installed as "modems" under MS Windows™, the configuration of the SMS Appliance tag enables the specification of this modem and the workstation to which the modem is attached. You may also set a security privilege, and if necessary, can mark the modem as disabled.

The SMS Manager takes care of forwarding message requests to any registered SMS agents in a round-robin sequence. It also provides a publisher for received SMS text messages. Code that subscribes to received messages will get notification of all received text messages. Subscribers are responsible for any filtering of received text messages.

The SMS Appliance tag will re-initialize the appliance once per minute to verify that it is still attached and functioning. Any error will result in a complete reset, which is repeated at a regular interval until the tag is disabled or a functioning device is available.

The SMS Appliance tag will queue messages to send, so long as it is operating normally. Received messages are published immediately. If these should be queued, it is the responsibility of the subscriber to do so.

The value of the SMS Appliance tag can be used to monitor the device status, according to the following table. (These values are reflected in the SMS Indicator widget.)

Value	Meaning
1	Connected to cell modem
2	Sending an SMS Message
3	Receiving an SMS message
4	Checking the connection to the cell modem
5	Disabled
6	Workstation with modem is not available
7	Modem unavailable

Related Information:

Roster Tags

SMS Appliance Properties: ID tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
> The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

SMS Appliance Properties: Settings tab

The Settings tab is used to identify the SMS appliance (SMS "modem") that will be used to send and receive messages.
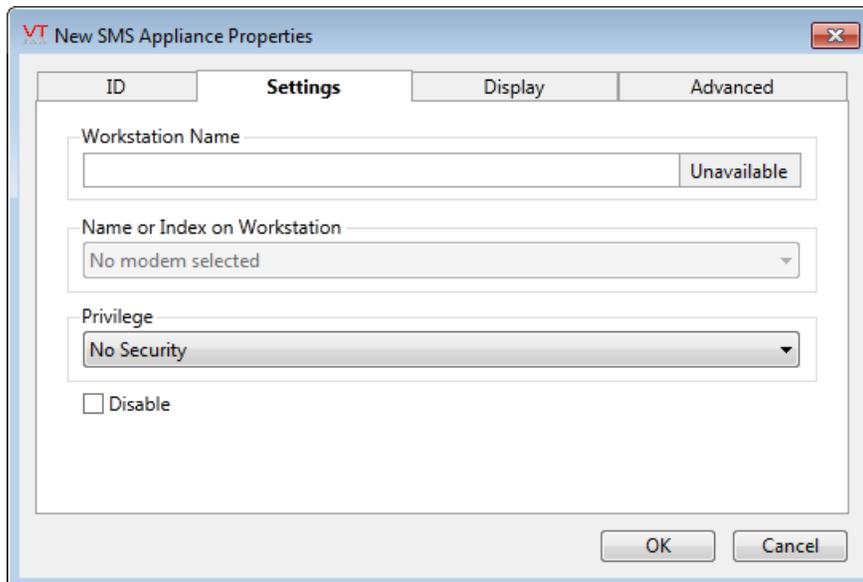
Workstation Name

The name of the computer, running the VTScada application, to which the SMS appliance is attached.

Name or Index on Workstation

The named workstation will be checked for devices. If more than one is found, you may select between them using this drop-down list.

Privilege

Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.

In this case, "operation" refers to the ability to disable or enable the tag via the context menu. It does not affect whether or not SMS messages are sent while a user without the privilege is logged on.

Application privileges are added using the Administrative Settings security dialog. Information on creating application privileges can be found in "Add Application Privileges Using the Administrative Settings Dialog".

Disable

Check this box to disable the device without removing it or deleting its configuration.

SMS Appliance Properties: Advanced tab

The fields in the Advanced tab are optional. These need not be set for most configurations and should only be configured by developers who are familiar with the configuration of SMS devices.

In rare instances, a device might not use a standard configuration of 9600 baud, etc. Serial port configuration fields are provided for use when working with such a device.
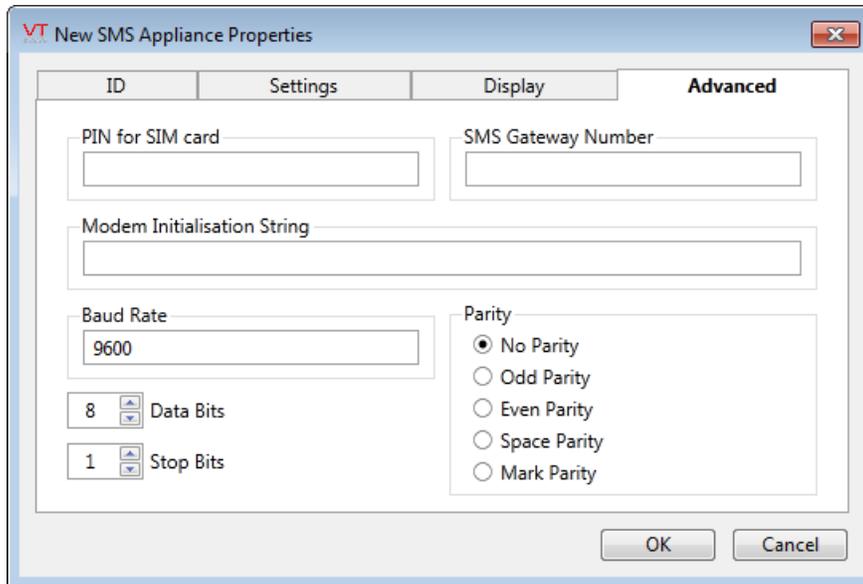
PIN for SIM Card

> If required, this will value will be assigned to you by your cellular provider.
> The password is pre-configured in most SIM cards.

SMS Gateway Number

> If required, this number will be provided by your cellular provider. This number is pre-configured in most SIM cards.

Modem Initialization String

> You may provide an initialization string, as specified in the user manual for your cell modem. This will be sent to the cell modem at the beginning of each initialization cycle.
> In most cases, you do not need to provide this string to the tag.

Baud Rate

> Change this field only if your device does not autobaud to 9600.

Data Bits, Stop Bits and Parity

> If your device does not autobaud to 9600, you might also need to configure these other serial communication parameters.

SMS Appliance Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
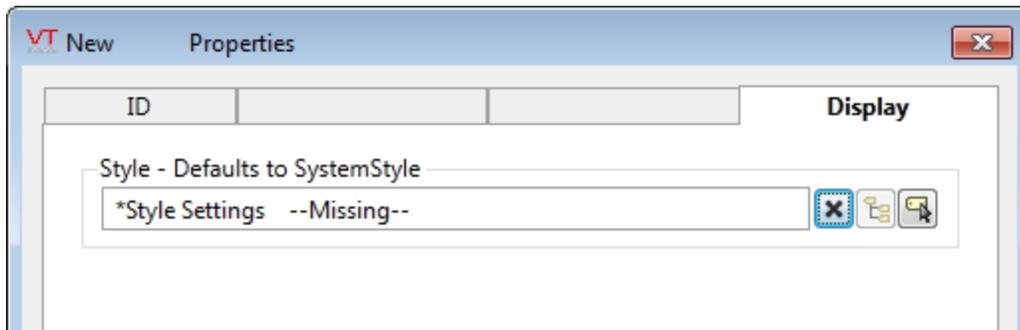
Related:

Style Settings Tags

SMS Appliance Tag Widgets

SMS Appliance Tags have a numeric value indicating their current state. The following widgets may be used to display this information:

...Alarm Priority Box

...Alarm Priority Icon

...Animated Image Widget

...Color Box Widget

...Image Change Widget

...LCD 5x7 Matrix Widget

...Equipment / Status Color Indicator

...Indicator Light Widget

...LCD 7 Segment Widget

...SMS Indicator Widget

...Multi-Color Widget

...Multi-Text Widget

...Numeric Value Widget

...Plot Data Widget

...Text Change Widget

Logging and Reporting Tags

Data logging tags are used to record the values of input and output tags to disk so that they can then be used in the Historical Data Viewer and reports. Reporting tags are used to specify groups of tags upon which to generate timely reports. Notebook tags enable users to add notes on the new Historical Data Viewer page.
The relationship between Historian tags and Logger tags is that Logger tags define when data is to be stored, while Historian tags define where the data is to be stored. Each tag for which data is to be stored must have a unique Logger tag. A single Historian tag is often enough for an application.

Related Information:

...Historian Tags. Looks after writing data to storage.

...Logger Tags. Links tags that do not have built-in logging to Historians. Enables configuration of logging frequency.

...Notebook Tags. Used to add notes on the Historical Data Viewer page.

...Report Tags. Generate a pre-configured report at the push of a button, or on a defined schedule.

...SQL Logger Group Tags. Provides support for SQL Logger tags, by holding the connection information for the data store.

...SQL Logger Tags. Writes data from a selected tag to the database configured in an SQL Logger Group tag.

...Transaction Logger Tags. Used to record a collection of historical values as a single transaction.

Historian Tags

Not counted towards your tag license limit.
The Historian Tag is used to write data that is to be logged to storage.
(The Logger tag controls only how often the Historian tag writes.)

Every VTScada application will have a default instance of a Historian tag named "SystemHistorian" for tag I/O values and one named SystemAlarmHistorian, used by Alarm databases. In many cases, these will be the only Historian tags that you use in an application. The information will be written to a VTScada proprietary database format unless otherwise configured.

> Warning: Do not use the same Historian to record both process values (tag I/O data) and alarm information.

The Historian tag's configuration panels provide access to only some of the available options. With a combination of Historian tags and related application properties, you can also configure the following:

- Save data to a proprietary database such as Oracle, Microsoft SQL Server, MySQL or SQLite instead of the VTScada database.
- Ensure that a redundant backup exists by writing to separate storage locations on two or more servers.
- Configure load sharing between servers.

The status of the Historian's connection to its data store can be viewed.

Historian Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area
> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.

- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Historian Properties: Datastore Tab

Choose how and where data will be stored.
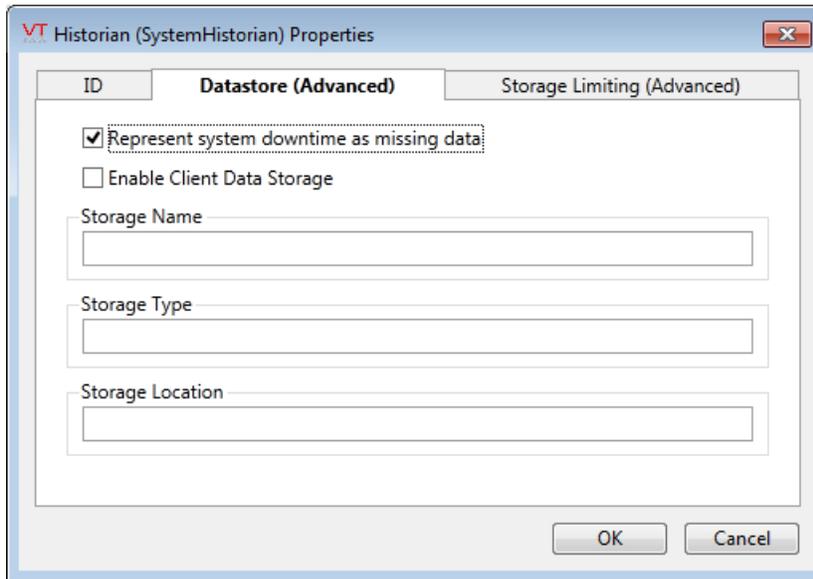
Represent system downtime as missing data

If checked, then system downtime will be represented as a gap in the data. If not checked, then the last recorded value will be taken as if it were in effect for the duration of the time that VTScada was not logging.
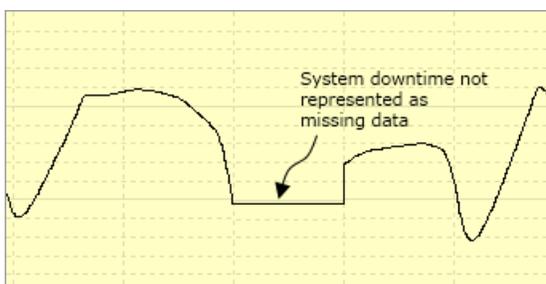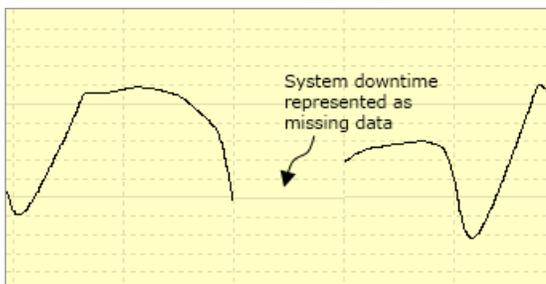
Enable Client Data Storage

Warning: Do not enable client data storage unless you have a clear understanding of the costs and benefits.
On a system with many clients, this feature can add a significant load to network communications.

This feature is intended for use with alarm database Historians, allowing isolated clients to modify the alarm system database and obtain alarm data from local storage. When enabled, Client Data Storage increases availability of the alarm system (especially in the event of a client becoming isolated from the network) at the cost of maintaining synchronization with all clients. Alarm historians tend to be relatively small (averaging one or fewer alarm or events per minute) when compared to the SystemHistorian, which may be logging thousands of records per second. It is entirely possible to use alarm Historians without this feature, but those clients will not be able to generate alarms and events when isolated from a server. They will also have to obtain alarm history from the server, which may require significant data transfer.

Warning: You are strongly advised to avoid using this feature with Historians that are logging I/O data.
If you insist upon doing so, ensure that drivers are configured with the Hold option set, to prevent isolated clients from logging invalids and thereby causing HDV gaps.

When enabled on any Historian tag, that tag will synchronize and retain its history files on all VTScada hosts, regardless of their presence in the server list. This generates extra synchronization and update traffic between all hosts and should only be used in the recommended circumstances such as alarm Historians, to allow alarm data to be available locally on all hosts.

As an advanced configuration option, you can specify client workstations that will not use this feature. To do so, use a tag parameter expression. Those clients will not need to be synchronized and will not store history locally, but will still receive data broadcasts.

> Note: If this feature is used, then it is essential that client storage be enabled on all workstations that are part of the server list.

The fact that a check box is used to enable or disable the parameter shows that it stores a Boolean TRUE (1) or FALSE (0) value. Therefore, the tag parameter expression must set a 1 or 0 depending on the identity of the workstation. You might use the WkStaInfo(0) function to test for the identity of each workstation, noting that the comparison is case-sensitive. The expression would be similar to that described in the topic, Historian Data Storage Options.

Related Information:

...Create or Remove a Parameter Expression

...Historian Data Storage Options

...Start Here for Scripting and Automation in the VTScada Programmer's Guide

Storage Name

Advanced Use Only. Enables a user-specified name to be used, rather than the tag's Unique ID. This provides control over the folder name that will be used in the data store.
If configuring an ODBC-based database, the use of this field is recommended, as the tag's Unique ID may not remain unique after being modified to a form that can be used by the ODBC driver.

Storage Type

> Advanced Use Only. May be "File" if using the VTScada data store. May be "ODBC" if configuring an ODBC-based database. Defaults to "File" if not otherwise specified and therefore need not be set if using the VTScada data store.

Storage Location

> Advanced Use Only. If using the VTScada data store, you may use this field to specify the path to a storage location other than the default. (C:\VTScada\AppName\Data)
> If using an ODBC-based database, this should be set to either the configured DSN (Data Source Name) of the database as follows:
> DSN=MyDSNName
> Or, you may use a connection string, in the form:
> Driver-
> =Serve-
> rBrand;Server=ServerName;Database=DBName;Uid=user;Pwd=pwd

Note: While it is possible to set the storage type and location using application properties, you are advised to use the tag configuration fields described here instead.

Related Information:
Historian Data Storage Options – More about storage options. How to configure logging to both the VTScada data store and a third-party database.
Query a 3rd-Party DBMS – Table structure

Historian Properties: Storage Limiting Settings Tab

Note: Historians used by Alarm Databases should not use storage limiting.

Choose whether to limit the data saved. If the data is to be limited, you can also choose whether to limit it by date, or by a maximum number of records.
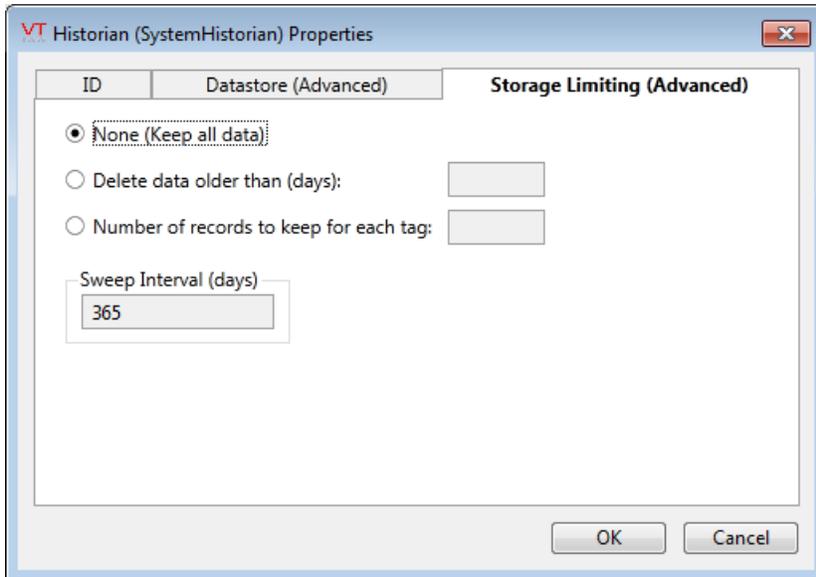
By default, no limit is set on the amount of data to be stored.

If you choose to delete data older than a given number of days, you will have the option of setting how many days that will be. The default is 365.
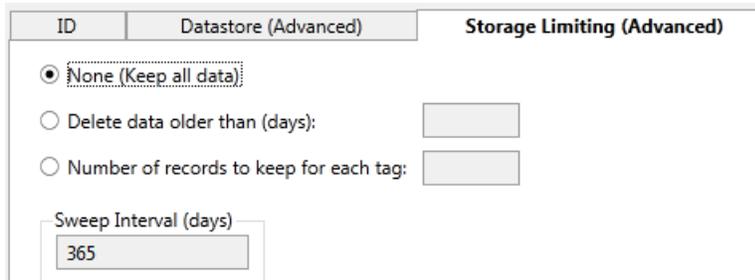
The sweep interval sets how often older data is deleted. When deleting data older than a set number of days, the sweep interval will always be exactly ¼ of the number of days specified. So, if deleting data older than 1 year, the actual removal of the older records will happen only 4 times per year. Between sweeps, data older than the set limit will accumulate.

If you choose a maximum number of records to keep, you can set the sweep interval to any number of days that you would like. Records bey‐ ond the set maximum number may accumulate between sweep intervals.

Widgets:

Historian Status – Displays information for all Historians, rather than being linked to a specific instance as widgets are.

Related Information:

...Logging and Reporting – Configuration – Advanced options.

...Alarm Data Logging – Details related to the alarm system.

...Historian Status – Monitor Historian activity.

Logger Tags

Not counted towards your tag license limit.
Logger tags are used to control when data is to be logged. They are required only for tags such as the Analog Input and Digital Input, which do not have logging built-in.
The Logger tag must be linked to a Historian tag, which will perform the actual recording of the data. The Logger's only job is to define when the value from the source tag is to be recorded.
Note that many tags have built-in logging through a connection to the Historian. As a general rule, the built-in logging is preferred when available. If you attempt to attach a logger to a tag that has its own historian connection, a warning will be displayed to discourage you from proceeding.
Logger tags themselves have no values and cannot be drawn on a page

Logger Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Logger Properties: Conditions Tab

The Conditions tab for the Logger tag properties folder consists of the attributes used to identify the tag whose value should be recorded and the logging mode to use when logging the tag's data. The logging mode can be a time period or for whenever the monitored tag's value changes.
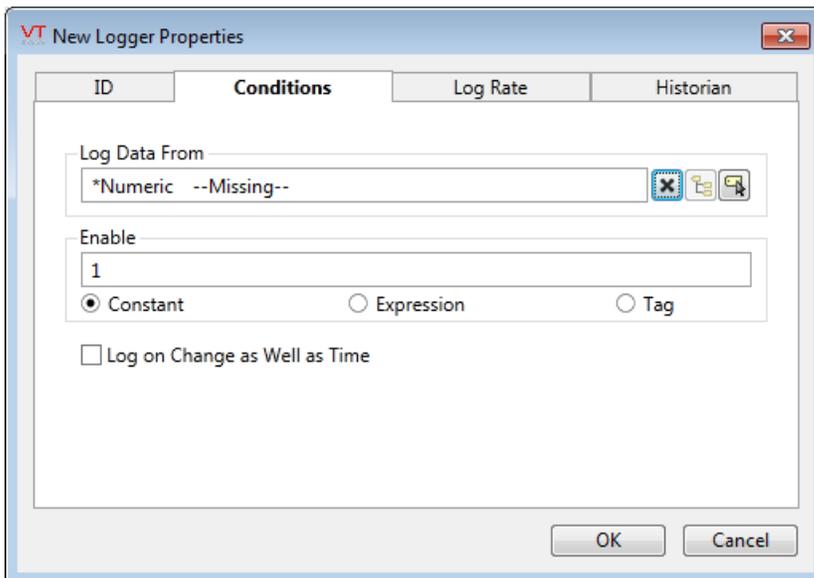
Log Data From

Select the tag whose values will be passed to a Historian by this Logger tag.

The Log Data From field can be used to associate this tag with a new or existing tag using the … button. The … button opens the Tag Browser, which displays only the existing tags for your application, and enables you to create a new tag using its New button. (Information on using the Tag Browser can be found in "Tag Browser").

The Log Data From field can be cleared using the X button to its right. Right-clicking the name of the tag that has been selected in the Log Data From field opens the tag properties folder for the selected tag.

Log On Change As Well As Time

The Log On Change As Well As Time check box enables you to specify whether the data for the specified tag will be recorded to the logger's .DAT file according to a specified time period alone, or also when the value of the tag being monitored changes.

If the Log On Change As Well As Time check box is selected, data will be logged to the .DAT file every time period as defined in the Interval property on this logger's Log Rate tab (see Logger Properties: Log Rate Tab), as well as when the value of the tag being monitored changes.

Enable

Enable or disable the logging of tag data by this logger using a constant value (either a 0, or a 1), an expression, or the value of a second tag.

If the Constant check box beneath the Enable field is selected:

- Enter a 1 to enable the logging of the data for the tag being monitored.

- Enter a 0 to disable the logging of the data for the tag being monitored.

Any expression or tag that returns either a 0 or 1 (or non-zero) can also be used. Please see Constant, Expression or Tag for help selecting which option to use.

Logger Properties: Log Rate Tab

The Log Rate tab of the Logger tag properties folder is used to identify the rate at which the data is recorded for the monitored tag.
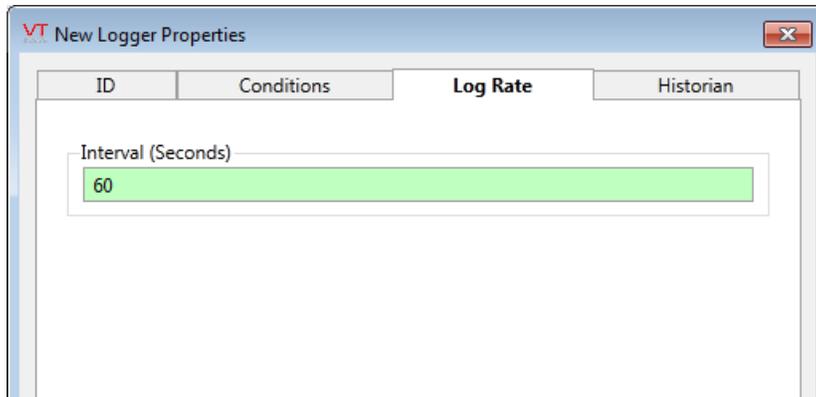
Logger Properties: Historian Tab

Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.
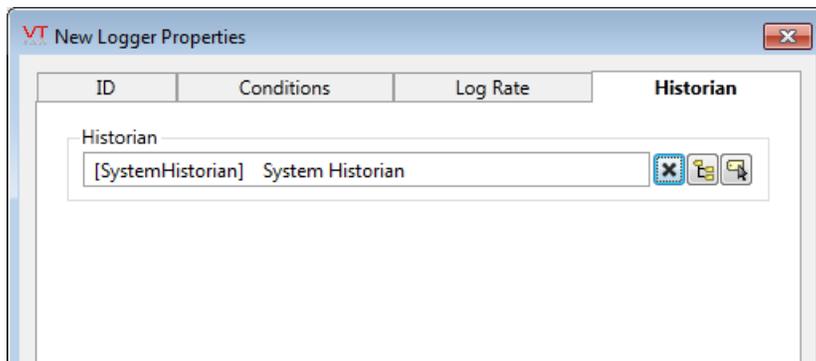
Notebook Tags

Not counted towards your tag license limit.

Notebook tags are used to store user-created notes. In essence, they provide a named link to an encrypted data file where operator notes are stored.

VTScada comes with two Notebook tags pre-configured: System Notes, which is the default for the Historical Data Viewer and Operator Notes, which is the default for the Operator Notes page.

You may create as many Notebook tags as required to organize your information. Note that every Alarm Database tag will have its own child Notebook tag, which will be created automatically. All notes created for an alarm are stored in the notebook associated with the matching alarm database.

Notebook Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Notebook Properties: SetupTab

Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer.

Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.
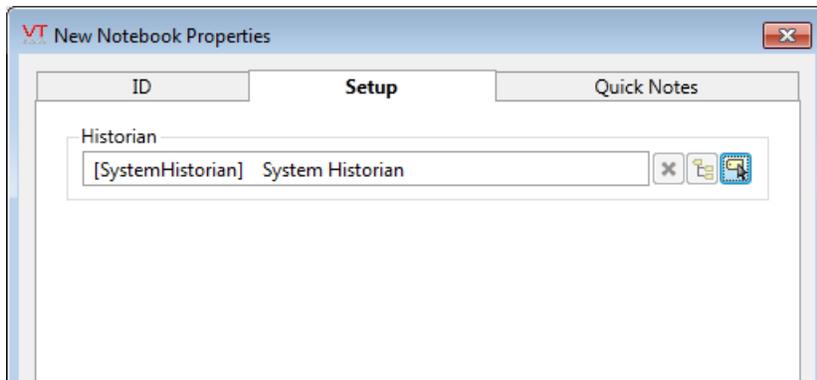
Notebook Properties: Quick Notes

Use this tab to create pre-written phrases that can be added to new Operator notes with a click. Each Notebook tag will maintain its own list of Quick Notes.

These can be templates with blanks for operators to fill in, or they can be often-used phrases or keywords to be used later when searching (example: [MAINTENANCE]).

By using a template, you ensure that information is organized in a consistent form within notes and that no details are forgotten. By providing often-used phrases, you can help operators by saving them the time they would otherwise spend typing the same phrase each time it's needed.

When operators add a Quick Note to a new note that they are writing, the text is copied to their editing field. No link is maintained between the Quick Note and the final version of the note that the operator saves.
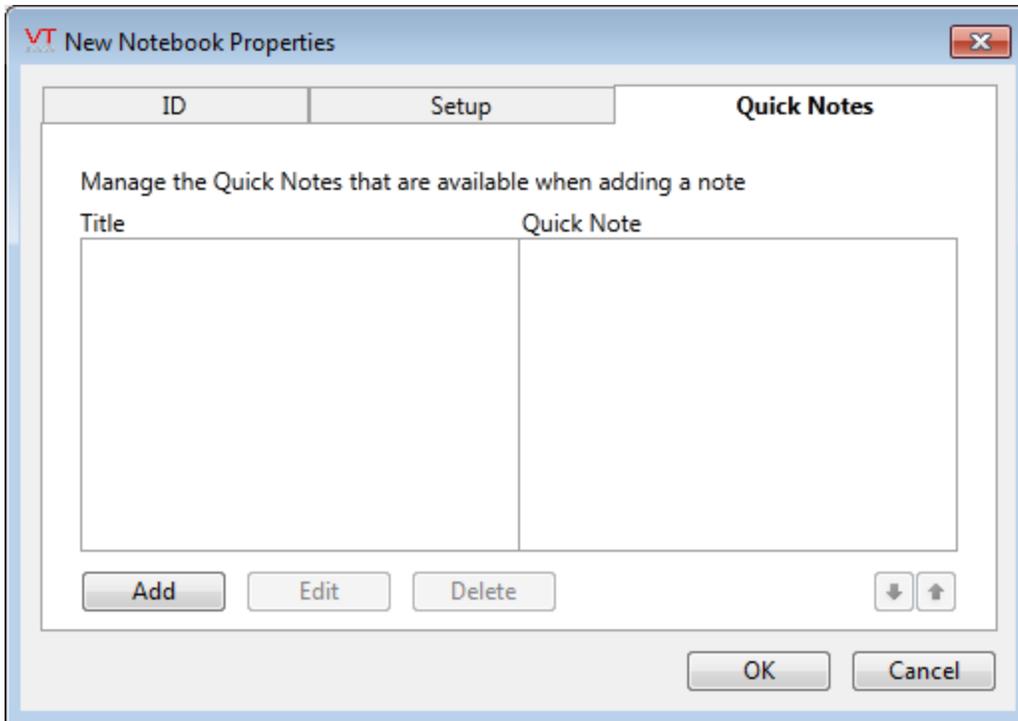
Add a Quick Note by clicking the Add button. Provide a title, which will be visible to operators when adding notes, and the text of the note.
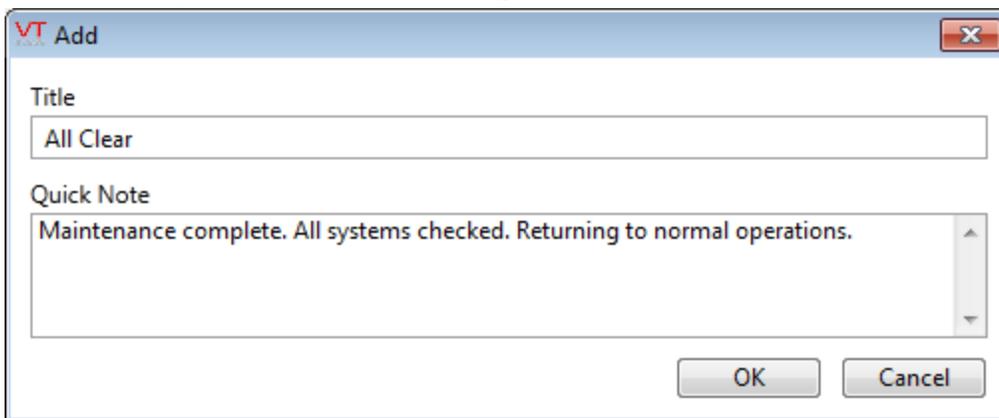
You may add, edit or delete Quick Notes at any time. No link is maintained between a Quick Note and a saved operator note, therefore the work that you do here has no effect on saved notes.

Notebook Tag Widgets

The following widgets are available for your application's Notebook tags:

Add Note Button

Note List Widget

Report Tags

Not counted towards your tag license limit.

The Report tag type is used to track the data of a user-defined group of tags to a user-defined output format at regular intervals.

VTScada also includes a built-in Reports page that enables the generation of reports, along with a set of report tools that enable you to build your own custom reports page. The difference between the default Reports page and Report tags is that the Reports page enables a one-time generation of a report, while Report tags enable you to configure a report to be generated on a regular basis (daily, weekly, or monthly), or whenever a related tag changes it value from 'false' to 'true'. An example of how to configure a Report tag can be found in "Use Report Tags to Generate Reports ".

A record will be added to the Events History each time a report is generated using the Report Tag. This enables operators to ensure that automated reports were created on schedule.

The value of the Report tag will be 1 when writing a report and 0 otherwise.

Report Properties: ID Tab

The Report tag's ID tab differs from the standard in that it includes an extra field. The Report Type drop-down list should be used to select the type of report that you want to generate using this tag. This may be one of:

- Daily Snapshot Report
- Daily Total Report
- Driver Communication Error Detail Report
- Driver Communication Summary Report
- Hourly Snapshot Report
- Hourly Total Report
- Standard Report

Other fields are standard for an ID tab...

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field.

You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

> Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.
>
> Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.
>
> The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Report Properties: Tag Tab

The Tag tab of the Report tag properties folder enables you to select tags whose data will be included in the report you wish to generate. The Tag tab enables you to filter the available tags by tag type or by area; so that you may more easily locate the tags you wish to incorporate in the report.
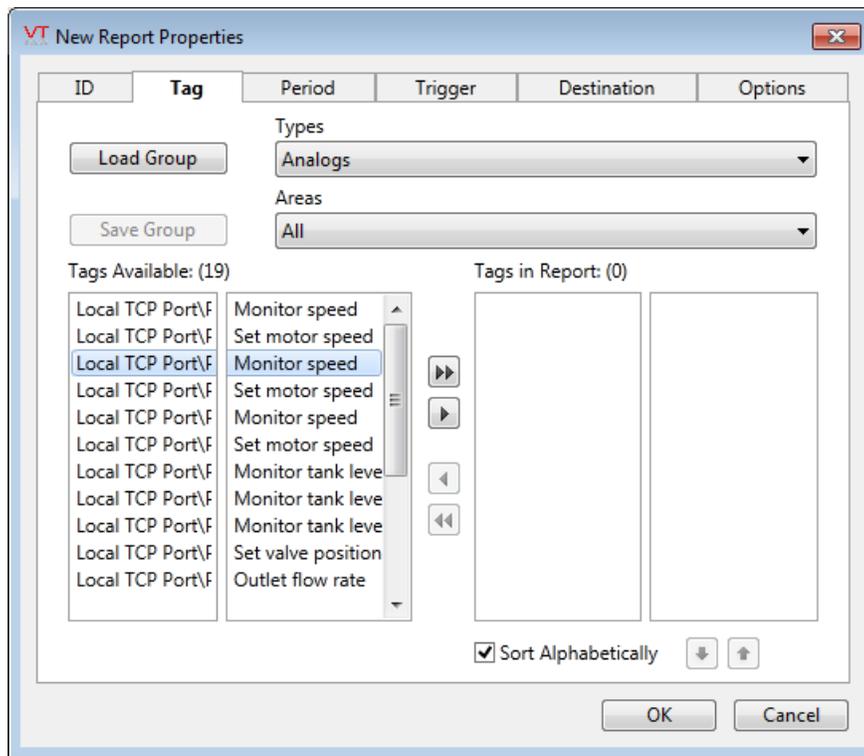
Load Group

The Load Group button enables you to load a set of tags that has previously been saved. A loaded tag group can then be used to generate a new report based on your application's requirements. Step-by-step instructions can be found in Saving and Loading Tag Groups for Reports.

Save Group

The Save Group button enables you to save the tags that have been configured for this report as a group that can later be reused for report generation. Step-by-step instructions can be found in Saving and Loading Tag Groups for Reports.

Types

Use this drop-down list to select the type of tags you wish to include in this report.

> Note: The Types drop-down list is set automatically according to the setting of the Report Type drop-down list on the Report tag's ID tab. You will rarely need to change it.

Areas

The Areas drop-down list enables you to filter the tags to be included in this report by their Area property.

Tags Available

The Tags Available list displays the names and descriptions of those tags that are available for inclusion in the reports to be generated by this tag. The $<$ button enables you to move a selected tag (or tags) between the Tags Available list and the Tags In Report list. The $<<$ button enables you to move all tags in the Tags Available list to the Tags In Report list.

Tags In Report

The Tags In Report list displays the names and descriptions of those tags you have selected to be included in this report. The $>$ button enables you to move a selected tag (or tags) between the Tags In Report list and the Tags Available list. The $>>$ button enables you to move all tags in the Tags In Report list to the Tags Available list.

Report Properties: Period Tab

Use the Period tab of the Report Tag properties folder to select a preset time period or configure a custom time period for the report to be

generated by this tag. Note: the Report Button widget provides an option for an operator to re-run the last scheduled report in the event that a system interruption prevented that report from being generated.

Last versus Previous:

The terms "Last" and "Previous" mean different things in the context of a report. "Last" refers to a time period ending at present (or at a defined end time if defining a custom period). "Previous" refers to the most recent full period where weeks end on Sunday night, days end at one second before midnight, and hours end one second before the top of the hour.

For example, suppose that the current time is 3:25 p.m. on a Tuesday. "Last 24 hours before trigger time" refers to the period from 3:25 p.m. Monday until 3:25 p.m. Tuesday.

"Previous calendar day" refers to the period from 12:00 a.m. Monday until 11:59 p.m. Monday.

"Current" refers to the time frames such as "so far today" or "so far this week".

Presets

> The Presets drop-down list can be used to select a pre-configured period of time for the data to be included in this report.
>
> If Custom is selected, further options will be available, as shown:

Period Type

The selection offers a choice between Previous Time Period or Duration and End Time.

Previous Time Period:

Select a time period (Days, Weeks, etc.) from the drop down list and enter a number in the field provided. "Previous" refers to the most recent time period, ending at a standard boundary such as the midnight, etc.

Duration and End Time:

Sets the length of time that should be included in the report, ending at a set length of time prior to the trigger event.

Report Properties: Trigger Tab

The Trigger tab of the Report tag properties folder enables you to select a trigger that will prompt the report to be generated. This trigger can be a daily, weekly, or monthly time, or any tag belonging to the numeric group.
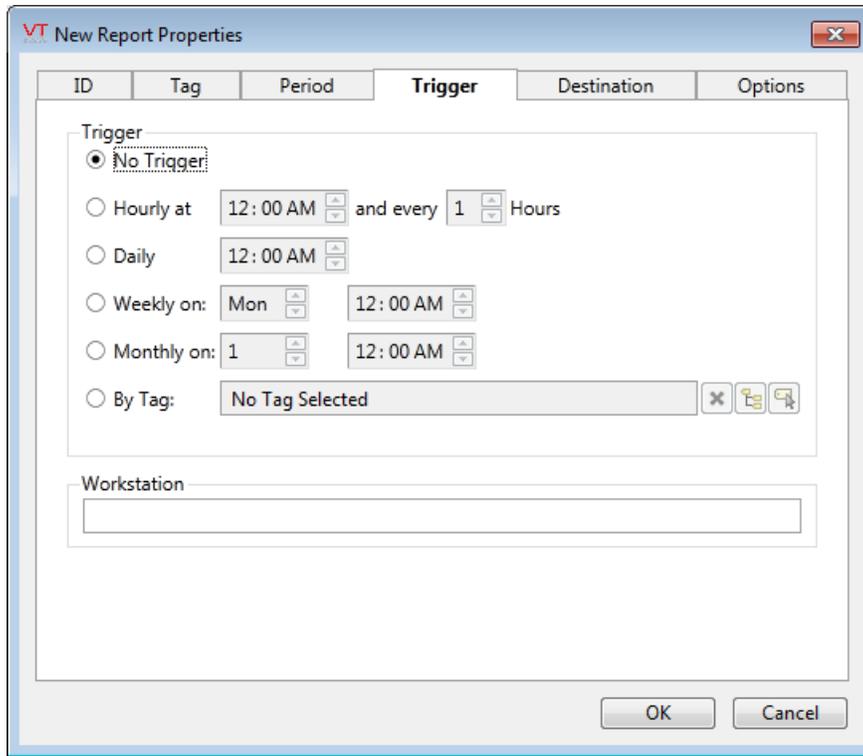
Trigger

The Trigger radio buttons enable you to select an event that will prompt the generation of this report. This can be one of:

- No Trigger

- Hourly at. (repeats every X hours, beginning at the specified time)

If X is 3, a specified time of 12:00, 3:00, 6:00 or 9:00 would all produce exactly the same result. You do not need to provide the earliest interval in the day.

If the system is offline when a report interval is due, that interval will simply be missed.

- Daily Trigger (at a user-defined time)
- Weekly Trigger (on a user-defined day at a user-defined time)
- Monthly Trigger (on a user-defined date at a user-defined time)
- By Tag (when the value of a specified tag changes from false to true). The Trigger Tag field is not enabled unless the By Tag radio button has been selected.

Trigger Tag

The Trigger Tag field enables you to select any tag with a numeric value whose value will be used to generate this report (when the value of the selected tag changes, the report will be generated). The Trigger Tag field only becomes enabled when the By Tag radio button is selected.

Right-clicking the name of the tag that has been selected in the Trigger Tag field opens the tag properties folder for the selected tag.

Workstation

The Workstation field enables you to enter the name of the workstation upon which this report will be generated. By default, the Workstation field is set to the name of the local workstation.

Report Properties: Destination Tab

The Destination tab of the Report tag's properties folder enables you to configure an output format and destination for the report to be generated by this tag.
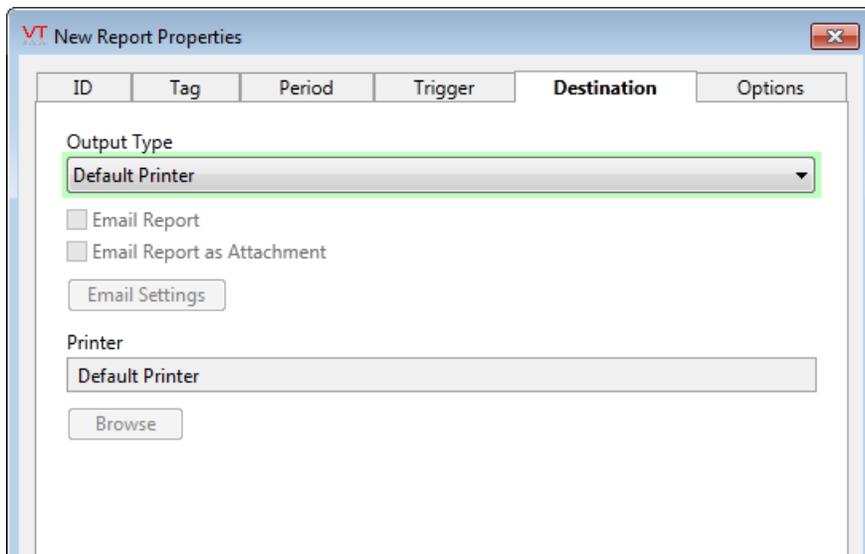
Output Type

The Output Type drop-down list enables you to select the format to be used to display this report. This can be one of:

- Default Printer: Prints the report to the printer configured for your PC under the Windows operating system.
- Printer: Enables you to specify the path to a printer other than the default printer configured for your PC under the Windows operating system.

- Text File: Generates the report as a plain, unformatted text file with the extension .TXT. This option enables the transmission of report data in the body of an email message (see, "Email Report").
- Screen Display: Generates the report in a window on your screen. This format enables the use of a Microsoft Excel template file to modify the way your report appears. Please refer to "Using a Microsoft Excel Template File to Generate Reports".

Note: A report that has had its Output Type configured as 'Screen Display', and for which the 'Use Excel to Display Screen Reports' check box has been selected will open as a regular screen display over VIC connections. The 'Screen Display Using Microsoft Excel' feature is not available for VIC connections at the present time.

- CSV File: Generates the report as a comma-separated value file with the extension .CSV. Such files are easily imported by a variety of different database software packages. This report output option enables the transmission of report data as an attachment to an email message (see "Email Report As Attachment").
- Formatted Excel XLS File: Generates the report as a formatted Microsoft Excel spreadsheet with the extension .XLS. (The report will feature a bold title and column headings.) This report output option enables the transmission of report data as an attachment to an email message "Email Report As Attachment").
- Plain Excel XLS File: Generates the report as a plain, unformatted Microsoft Excel spreadsheet with the extension .XLS. (Unlike the Formatted Excel XLS File option, the report will not feature a bold title and column headings.) This report output option enables the transmission of report data as an attachment to an email message (see "Email Report as Attachment").
- Access MDB File: Generates the report as a Microsoft Access database file with the extension .MDB. This report output option enables the transmission of report data as an attachment to an email message (see "Email Report As Attachment").

- ODBC Data Source: Enables you to specify an existing ODBC data source associated with an existing ODBC-compatible database file into which you wish the report data to be saved.

Email Report & Email Report as Attachment

When selected, the Email Report check box causes the report generator to email a copy of the report data to a specified address or set of addresses within the body of the email message. The maximum possible number of characters for the address is 255.

The Email option is available with most output types, provided that you have configured an email server.

Email Settings

After choosing to email a copy of the report, click this button to open the Email Settings dialog, where the recipients, subject, and message for the email can be configured.

Email addresses may be in the basic form, Name@Company.Com" or you may use either of the following longer forms. Multiple addresses may be entered, separated by semi-colons or commas. Do not add spaces within the angle brackets.

"Full Name" <Name@company.com>

Full Name <Name@company.com>

Destination

The destination field's label changes according to the selected report output type.

You may enter a path, printer, or ODBC data source for the report. (A path is not required for Default Printer or Screen Display output types). You may specify a path or printer using either the Browse button to navigate to the desired location, or by manually typing the path into the destination field.

Browse

Opens a dialog with which you can find and select a printer (when Printer is selected in the Output Type drop-down list) or a path to a Microsoft Excel template to be used to generate this report (when Screen Display is selected in the Output Type drop-down list). Instructions on utilizing Microsoft Excel templates to generate reports can be found in "Using a Microsoft Excel Template File to Generate Reports ".

Report Properties: Options Tab

The Options tab of the Report tag's properties folder enables you to con-
figure an output format and destination for the report to be generated
by this tag.
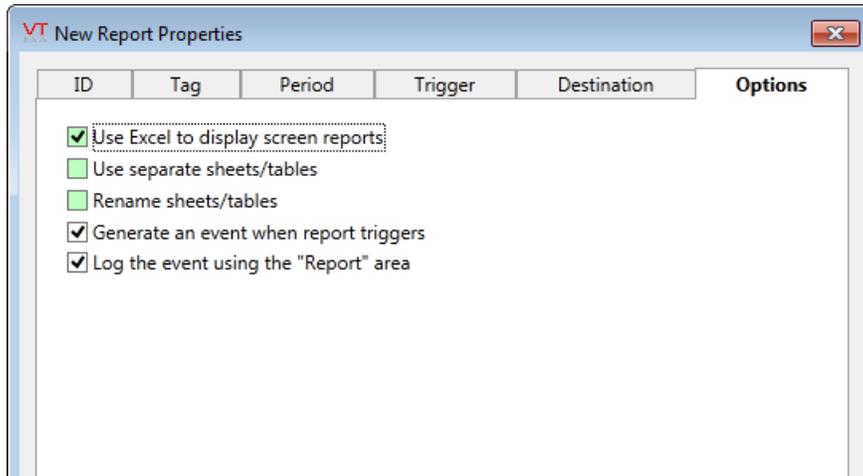
Use Excel to display screen reports

The Use Excel to display screen reports check box can be selected if you
wish Microsoft Excel to launch your report when the Screen Display report
output type is selected.

If you do not select the Use Excel to display screen reports check box, then a
simple window will be launched and the report data will be displayed within
it.

Note: A report that has had its Output Type configured as 'Screen
Display', and for which the 'Use Excel to Display Screen Reports'
check box has been selected will open as a regular screen display
over VIC connections. The 'Screen Display Using Microsoft Excel'
feature is not available for VIC connections at the present time.

Use separate sheets/tables

The Use separate sheets/tables check box can be selected if you wish
Microsoft Excel to display report data using separate spreadsheets or tables.
If you do not select the Use separate sheets/tables check box, then the
report data will be placed in a single spreadsheet or table.

Rename sheets/tables

The Rename sheets/tables check box can be selected if you wish the report spreadsheet(s) or table(s) to be renamed corresponding to the date and time at which the report was generated.

Generate an event

When checked, an event will be added to the application's history. Events can be viewed in the Alarm Page by selecting the History display option.

Log the event using the Report Area

The Generate an event option must also be selected. Causes the event to be associated with the system area, "Report". Otherwise, the event will be associated with the report tag's area.

Report Tag Widgets

The following widgets are available to display information about your application's Report tags:

Report Button Widget

Report Options Widget

Report Destination Widget

Report Tag List Widget

Reporting Period (Enhanced) Widget

Report Type Widget

SQL Logger Group Tags

Not counted towards your tag license limit.

SQL Logging will record, in an ODBC-compatible database, the values from any tag that could normally be logged.

The SQL Logger Group tag type works in combination with the SQL Logger tag. You use the SQL Logger Group tag to define a database and the frequency of data logging for all the SQL Logger tags associated with it. The SQL Logger tag is used to select which values are to be logged to that database. You may link one or more SQL Logger tags to one SQL Logger Group, recording one or more values into a single database.

Before You Begin:

Before you can log values to a database, you will need to have 1) a database containing two tables created using the template in the VTScada examples folder, and 2) an ODBC DSN configured for your database.

1. Setting up a database:

You can use any ODBC compatible database such as MS-Access, MS-SQL, MySQL, Oracle, etc. Instructions for how to install and configure a database program are beyond the scope of this documentation.
Once you have a database created, you will need two tables within it to hold the data that VTS will export. In the Example folder under VTScada (normally found as C:\VTScada\Examples) there are four files to help you with this task.

- SQLLoggerAccess.mdb
- SQLLoggerMySQL.SQL
- SQLLoggerOracle.SQL
- SQLLoggerSQLServer.SQL

The three files which end with the extension ".SQL", contain SQL commands to create the tables you will need in your database.
If you do not have a database and do not own a database program, you can still use this feature of VTScada by following these steps:

1. From the examples folder of VTScada, copy the file named SQLLoggerAccess.mdb to your application folder.
2. Rename it to match the data you plan to collect, being sure to keep the .MDB extension on the name.

You are now ready to proceed.

2. Configure an ODBC data source name (DSN)

This is created using the Windows™ Open Data Base Connectivity (ODBC) tool. A 64-bit version of this utility can be found in the Administrative Tools section the Windows™ configuration menu. A 32-bit version is available as "C:\Windows\Syswow64\odbcad32.exe". Instructions for using the Microsoft utility are beyond the scope of this topic.

This must be done on all potential servers for a networked VTS application. The DSN must be identical on each machine.

SQL Logger Group Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags

(note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

SQL Logger Group Properties: Settings Tab

The Settings tab holds the properties required to connect this tag to a database.
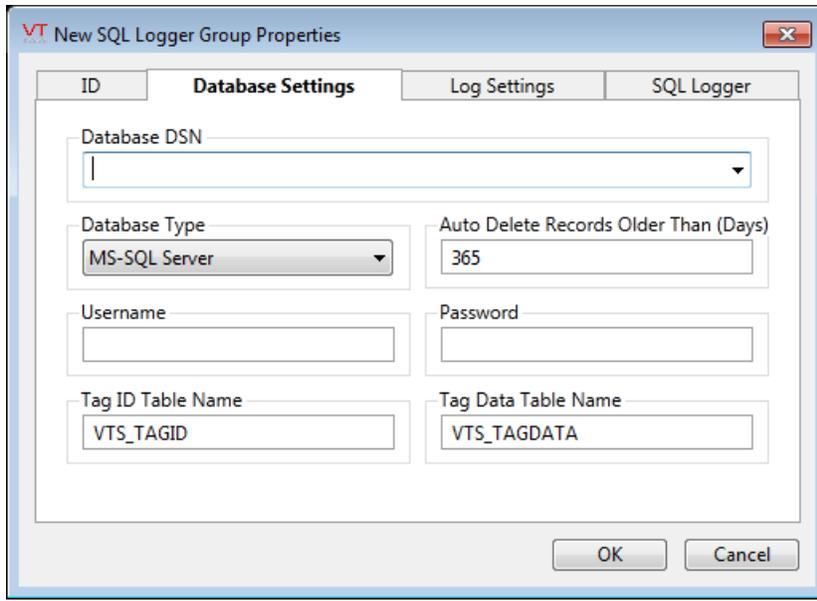
Database DSN

As described in the introduction to this topic, you must create and configure a database before using the SQL Logger Group tag.

The DSN (Data Source Name) must already exist on your system and be pointing to the database you created for this tag.

Database Type

Tell VTScada what type of database you are writing to.

> Note: CAUTION: The Database Type selection MUST match your database. Choosing an incompatible type will result in the tag failing to write values to your database due to differences in data format.

Auto Delete Records Older Than (Days)

Limit the size of the database by automatically deleting records older than (n) days.

You may choose to set this value to 0 to turn off the auto-delete feature.

Username, Password

The Username and Password fields should match whatever is required to connect to your database. If you are using MS Access and did not configure a user name and password, then you may leave these fields blank.

Tag ID Table Name, Tag Data Table Name

The two table name fields should not be changed under ordinary circumstances. The database configuration utility supplied with VTS/VTScada created these table names for you by default. (see SQL Logger Group Tag: Before you begin)

SQL Logger Group Properties: Log Settings Tab

Contains options to configure the timing for when data will be written to the database.

There are three choices of Log Type: Interval, Triggered and On Data Change. Note that all the tags attached to a particular SQL Logger Group through the associated SQL Logger tags will use these settings.
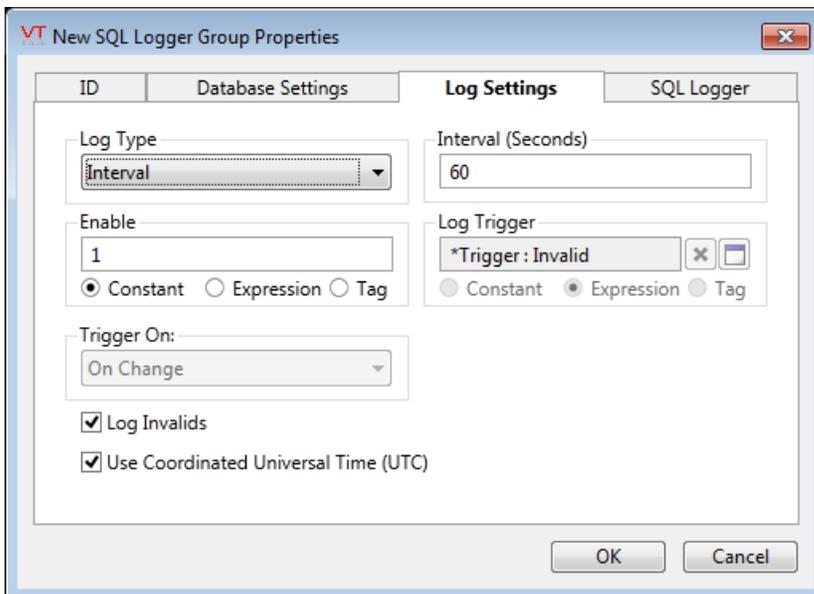
Log Type

- "Interval" – Record values every (n) seconds. Use this when you want continuous monitoring of the system, setting the interval to the number of seconds you want between each data output.

- "Triggered" – Write values only when some condition changes. The write could be triggered by a tag changing value, or you can create an expression to define precise conditions that will cause a value to be written. Note that the trigger does not need to be activated by the same tag whose value you are recording. You could potentially record the level of a holding tank every time a pump switches on or off.

When logging is to be triggered, you must also choose what the trigger will be. For testing purposes you can choose Constant, setting the value to 1 or 0. You can also write an expression that will evaluate to TRUE or FALSE (1 or 0), or you can select a tag that will have a value of 1 or 0.

Trigger On

If you have chosen Triggered as the Log Type, you also have the choice of whether the trigger happens on a Value Change, a Rising Edge (from FALSE to TRUE) or a Falling Edge (from TRUE to FALSE. This applies to Constants, Expressions and Tags.

Log Type: On Data Change

This option for log triggering passes more control over to the Logger tags attached to this Logger Group. Essentially, you are indicating that log values are to be written when the value of the tag you are monitoring changes. See the section on the SQL Logger tags for more information.

If using this log type, please read the following section: Log Invalids for an important note.

Log Invalids

You have the option of whether or not to record "invalids" as part of the data being written. An invalid data value is essentially a NULL.

Note that if you select On Data Change as your trigger and you have not chosen to log invalids, then if the data changes from a valid value to invalid (or vice versa), no value will be written. You must enable Log Invalids in order to capture these data changes.

Enable

Where triggering controls the timing of data output, Enable sets whether or not data is to be output at all. Where the two controls offer the same options of "Constant", "Expression" and "Tag", the difference may seem subtle, but it is fundamentally different. Enabling the logger group means that logging is switched on or off. Triggering means that (if logging is enabled) data values are written when the trigger signals them to be.

Note: If a SQL Logger Group is disabled, then all the SQL Logger tags attached to that group are automatically disabled.

Use Coordinated Universal Time (UTC)

A switch giving you the option to record the time stamp using either local time or UTC.

SQL Logger Group Properties: SQL Logger tab

Each Logger Group Tag must have one or more SQL Logger tags contributing data to it before it will be useful. As noted, the SQL Logger Group Tag creates a link to a database and sets a frequency of logging – it does not actually select what data is being logged. That job is assigned to the SQL Logger tag as described in the following section.
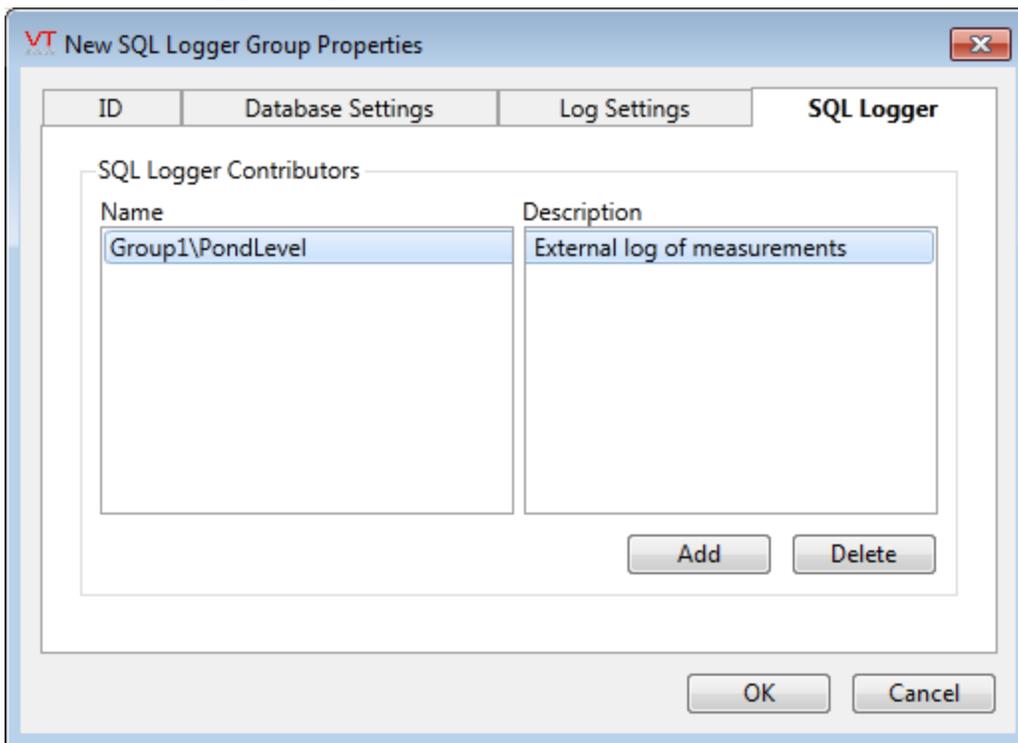
SQL Logger Group Tag Widgets

Only one widget is available to SQL Logger Group tags:
Show Stats Widget (ODBC)

SQL Logger Tags

Not counted towards your tag license limit.
The SQL Logger tag type is used to select where the data to be logged is to come from. Before using an SQL Logger tag you should create a SQL Logger Group tag as described in the preceding section. SQL Logger tags contribute values to SQL Logger Group tags.

Each record added to the database will include:

- An index value indicating which tag's value is being recorded
- A time stamp
- The value of the tag being logged at that moment
- A flag to show whether the value is 'manual data'
- A flag to show whether the value is 'questionable data'
- An alarm flag
- (Characteristics available in the Table of Type Characteristics.)

SQL Logger Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might

also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

SQL Logger Properties: Log Settings Tab

The Log Settings tab is where you select and control which tag's values you wish to log.
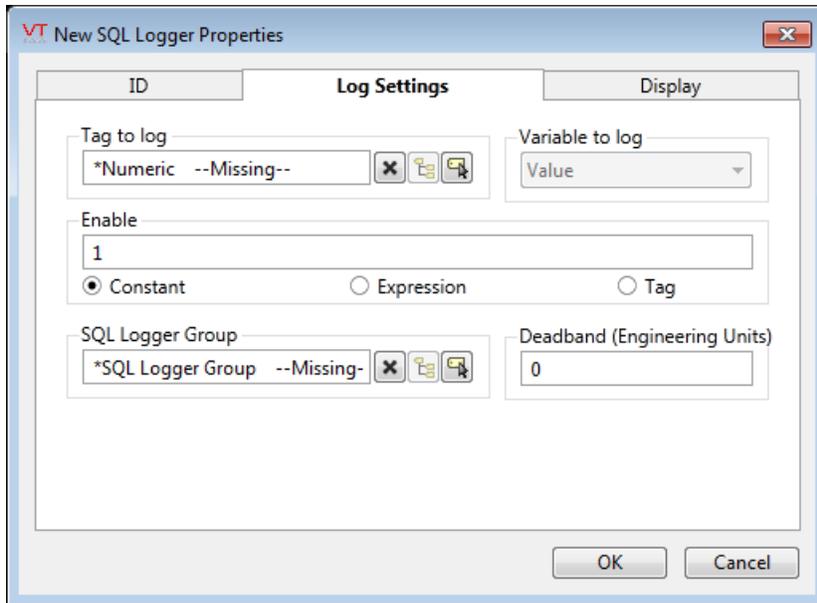
Tag to log, Variable to log

In the "Tag to log" selection area, simply choose any existing tag from your project. If that tag has multiple values which could be logged, you will have a choice in the "Variable to log" selection. In most cases, the variable will simply be "value".

Enable

Enable or disable logging of the selected value by entering a 1 for Enable and a 0 for disable. You also have the ability to toggle whether logging is enabled based on a script expression, or by selecting a tag: either the same tag you are logging the data values for, or any other. This provides you with the option of either logging continuously or logging only when a condition is met such as when an Alarm tag is activated.

SQL Logger Group

An SQL Logger Group must be selected. SQL Logger tags contribute their values to SQL Logger Group tags which then manage the actual output of the data to a database.

Deadband (Engineering Units)

If (and only if) the associated SQL Logger Group was set to trigger on data changes, it has essentially passed control to this tag to cause data writes to occur. If this is the case, then the Deadband field will be enabled and you will have the option of indicating how much the value must change by before another value will be written to the database.

SQL Logger Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
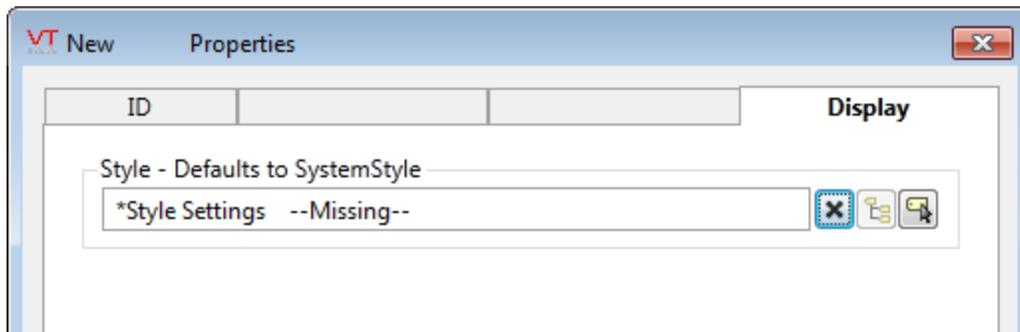
Related:

Style Settings Tags

SQL Logger Tag Widgets

The following widgets are available to display information about your application's SQL Logger tags:

Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Equipment / Status Color Indicator	Image Change Widget
Indicator Light Widget	LCD Bar Widget
LCD 7 Segment Widget	Left Bar Widget
Meter 1 Widget	Meter 2 Widget
Meter 3 Widget	Multi-Color Widget

Multi-Text Widget   Numeric Value Widget

Plot Data Widget   Right Bar Widget

Text Change Widget   Top Bar Widget

Two Color Bar Widget

Transaction Logger Tags

Not counted towards your tag license limit.
Transaction Logger tags can be used to record a collection of historical values as a single transaction. These are useful for material-handling applications that must record transfers of custody.
When the trigger changes to a non-zero value, all of the records in the history associated with the Analog Status, Digital Status, and String I/O tags that are immediate children of this tag, will be read and coalesced into a structure, which will be written as a single transaction record in the Historian associated with this tag. No other tag types will be included, nor will any tag that is below the level of the immediate children of the Transaction Logger.
The field names in this transaction snapshot will match the names of the tags being recorded, and are generated in the order of Analog Status, Digital Status and String I/O, with the fields being alphabetically ordered within types. All field names are stored with every transaction, since the child tags of the Transaction Logger may change over time.

The driver parameter of all the child tags of a Transaction Logger must be set to that Transaction Logger. Child tags that do not use this tag for their driver parameter will not be included in the transaction.
You are advised to use the History address of the input tags where possible, so that there will be a positive confirmation of the data being read via WriteHistory. This will avoid false positives on communication errors due to other users of the driver having comm errors concurrently with the transaction's polling.

Error Status

The value of the Transaction Logger tag shows its current error status, according to the following table:

Tag Value	Meaning
0	No error.
1	Communication Error.
2	For multiple-record devices (such as the ROC), the time stamps for consecutive records across all fields do not align.
3	Not all fields have the same number of records returned.
4	Driver became unavailable.

When there is an error, the I/O data corresponding to the poll attempt is written into the record, but these error records are not accessible via ODBC. In the case of the Communication Error, only one record will be written until the value becomes something other than 1. It is recommended practice to attach an alarm to each of your Transaction Logger tags.

Data Encryption

Stored data is encrypted using a default key (not published). If there is a reason to ensure that your data cannot be read by anyone using a standard installation of VTScada, you may wish to create your own encryption key. Do so by adding the property, TransactionEncryptKey to the System section of your Settings.Dynamic file, setting the value to a new key of your choice.

Data Retrieval

There is a single, virtual ODBC table called TransactionLog, which presents the data for all transactions by all Transaction Logger tags. This table will be the merged columns that match the tags in the query. The table's columns are defined by the child field tags that are present at the instant the query is made. There is a column, called TransactionLogger, that gives the associated Transaction Logger tag. Each record will have a 36-character GUID, stored in the column TransactionID.

The only method for retrieving logged transaction data is to use an SQL query. This may be done through code, calling the SQLQuery function, or through the ODBC Server. An example of an generic data retrieval query follows:

```
SELECT * FROM TransactionLog LIMIT 100;
```

Transaction Logger Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you

can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Transaction Logger Properties: I/O Tab

The I/O tab for the Transaction Logger tag consists of...
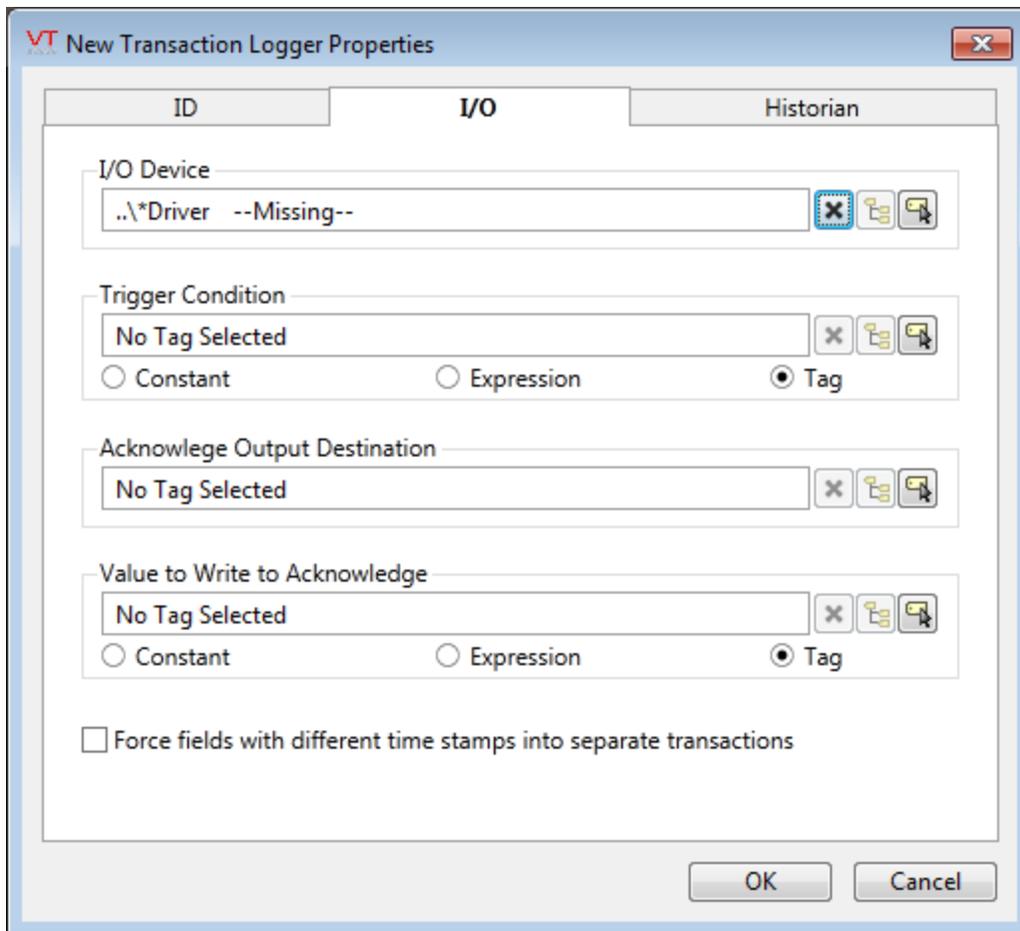
I/O Device

Select the communication driver tag from which data will be read.

By default, the tag will look for a parent tag that is a device driver (*Driver). If none is found, the text "--Missing--" will be displayed. The tag button to the right of the field opens the tag browser, from which you can either select an existing communication driver tag or add a new one. The X button will clear the field. Right-clicking on a tag in the field will open a dialog with which you can add or remove a Snapshot Expression, or open a selected driver's properties dialog.

Trigger Condition

May be any tag or expression that will change from a zero to a one, signaling when to write the transaction record.

Acknowledge Output Destination

Must be a tag. In most cases, the PLC will need to know that the transaction has been read so that the PLC can update the registers for the next transaction that may be in the queue. The output provides that acknowledgment to the PLC.

Value to Write to Acknowledge

Provide the value that is to be written to the acknowledge output destination. This may depend upon the error status of the transaction tag, in which case an expression should be used to generate the appropriate value.

Force fields with different timestamps into separate transactions

When selected, data returned for each field does not have to have the same number of records. Time stamps that do not match will be placed in separate transaction records.
This option assumes that the time stamps are in order in tag history. The transactions will be written with the time stamps corresponding to the data times in chronological order. Should be checked for the TBox chronological data.

Transaction Logger Properties: Historian Tab

Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Transaction Logger Tag Widgets

The following widgets are available to Transaction Logger tags:

Text Change Widget

Image Change Widget

Color Box Widget

Color Blink Widget

Animated Image Widget

Two Color Bar Widget

Plot Data Widget

Color Line Widget

Indicator Light Widget

Equipment / Status Color Indicator

Multi-Color Widget

Multi-Text Widget

Numeric Value Widget

Calculation and Inquiry Tags

These are tags whose value depends on some function of other tag values. Calculation tags and Function tags allow you to create your own expressions.

Other tags in this group provide inquiry functions that you may wish to use such as Counters, Totalizers, Alarm Status counts, etc.

Related Information:

...Analog Statistics Tags – Monitor average, maximum and minimum values of any tag having a numeric value.

...Calculation Tags – Monitor other tag values as part of whatever calculation you write.

...Counter Tags – Count the zero to non-zero transitions of a digital tag.

...Digital Statistics Tags – Monitor non-zero time and zero to non-zero transitions of any digital tag.

...Function Tags – An older technology for creating calculations, based on other tag values.

...History Statistics Tag – Any of seven calculations based on a numeric tag's value over a defined time interval.

...Rate of Change Tags – Monitor the rate at which another tag's value changes.

...Totalizer Tags – Create a running total of another tag's value over time.

...Script Tags – Activate a script module, based on a set-point that monitors another tag's value.

Analog Statistics Tags

Not counted towards your tag license limit.
Analog Statistics tags are used to monitor any tag having a numeric value. While they can use digital tags, they are more commonly used to collect statistics on Analog Input, Analog Status and other similar tags. Three statistics are gathered are gathered: minimum value during selected time frames, maximum value and average value, again during selected time frames.
The Analog Statistics tag works by launching a child tag for each statistic to be gathered. Each child tag may be included separately in a report or Historical Data Viewer plot.

Note: Select only the time frames required. A separate History Statistics tag will be launched for each statistic and time frame you choose.

Note: By default, the child tags are not logged. You must attach an Historian to each child tag for which you intend to save a record of the calculated statistics.

Analog Statistics Properties: ID Tab
The ID tab of every tag includes the same four elements. Name, Area,

Description and Help ID.

Name:

> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

> When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

> To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.
> The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).
> To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an exist-

ing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Analog Statistics Properties: Source Tab

Used to select the source tag for which statistics are to be calculated.

Source

The Source tab is used to select which tag is to be monitored.

Any tag with a numeric value may be selected as the source.

Exclude from Site List

Selected by default. You may choose to include this tag in the list of a Sites page as a folder. Operators may view a list of current statistics by clicking the link:

Analog Statistics Properties: Minimum Values Tab
Select the time frames, during which you wish to record the minimum value of the source tag. By default, only "Last Hour" and "Today" will be selected. Select all that you will need, remembering that a separate History Statistics tag will be launched for each.

Note: Statistics for the time periods "this week" and "last week" are affected by the application property, StartOfWeek. You may use that property to set the beginning of a week to any day of your choice.

"Today" means since the time 00:00 this day.
"This Week" means since 00:00 of the first morning of the week.
"This Month" means since 00:00 on the morning of the first day of the month.
"Yesterday" means the 24 day, prior to today.
"Last week" means the full week prior to the current week.
"Last Month" means the calendar month prior to the current one.
"Last 24 Hours" covers the 24 hours prior to "right now".

Analog Statistics Properties: Maximum Values Tab

Select the time frames, during which you wish to record the maximum value of the source tag. By default, only "Last Hour" and "Today" will be selected. Select all that you will need, remembering that a separate History Statistics tag will be launched for each.

Available time frames are the same as for the Minimum Values tab.

Analog Statistics Properties: Average Values Tab

Select the time frames, over which you wish to record the average value of the source tag. By default, only "Last Hour" and "Today" will be selected. Select all that you will need, remembering that a separate History Statistics tag will be launched for each.
Available time frames are the same as for the Minimum Values tab.

Analog Statistics Tag Widgets

The following widgets are available to display information about your application's Analog Statistics tags:
Tag List Widget

Calculation Tags

Not counted towards your tag license limit.
The Calculation tag type is used to hold an expression, with the resulting value of the tag being the result of the expression. This tag type replaces the Function tag.

Related Information:

...Creating Expressions – VTScada Programmer's Guide – Reference for the VTScada scripting language.

Calculation Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area
> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Calculation Properties: Calc Tab

The Calc tab holds the expression for the calculation to be performed by this tag. While a choice of Constant, Expression or Tag is provided, you would normally use only Expression.
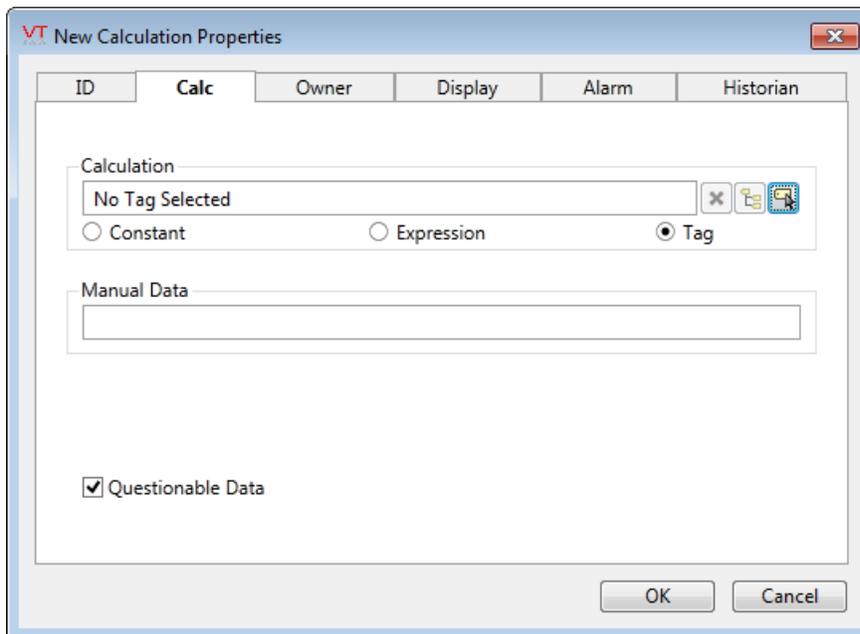
Calculation

The Calculation field enables you to specify a constant numeric value, an expression, or reference another tag that will be the calculation run by this tag. The Calculation field corresponds to the selected associated check box. Select Expression and click on the [...] button to open the expression editor.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark within any linked widgets.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

Calculation Properties: Owner Tab

This tag can be used in an owner/contributor structure where multiple contributor tags can supply their values to an owner tag. There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create, using the VTScada scripting language.
The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[...] To Value check box is used to set the value of this contributor tag as the nth element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected,

the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

.

Calculation Properties: Display Tab

Use the options in this panel to set display characteristics when the tag is represented on a page by widgets or plotted in the Historical Data Viewer.
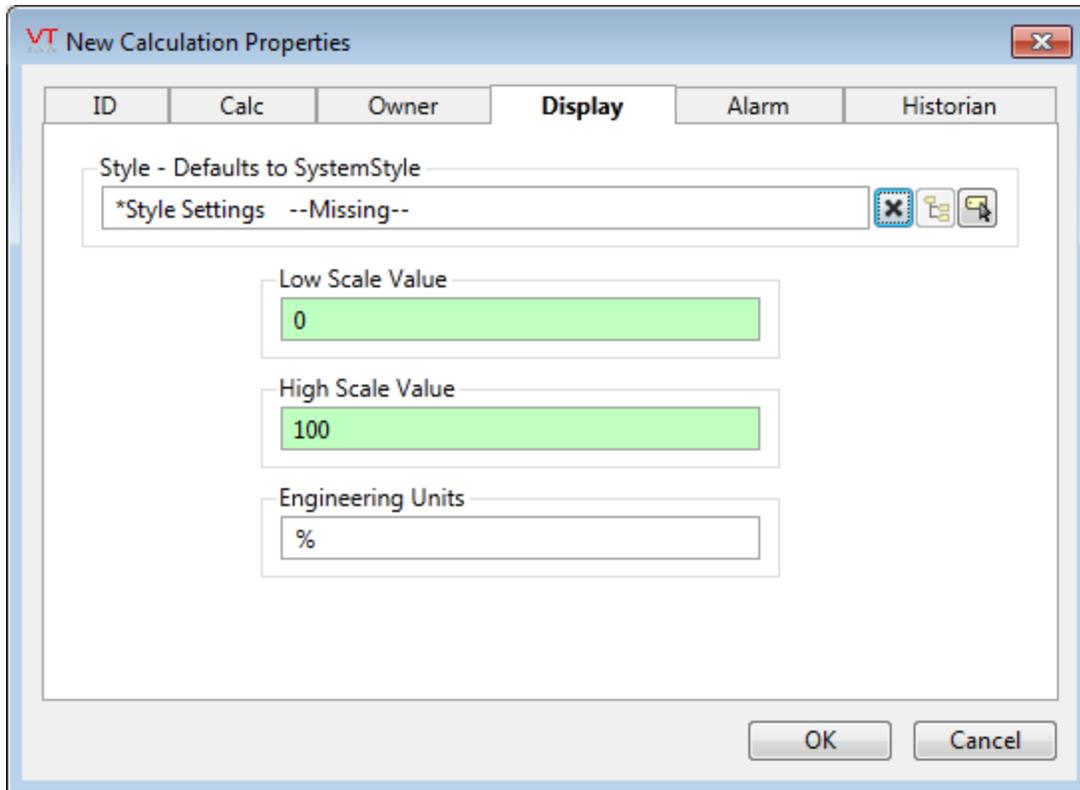
Style

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, which is the default style tag that is automatically part of every new VTScada application.

Related:
Style Settings Tags

Low and High Scale Values

Note: These fields have no effect on the value of the Calculation tag. They do affect the logging rate (if enabled) and

default scale ranges when this tag is represented in the HDV or various widgets.

Set the Low Scale Value and High Scale Value to define the default scale range for the HDV trend view and certain widgets, such as the various meters.

Engineering Units

Provide the units of measure that the calculated value represents. This text will be used by widgets that include engineering units as part of the display. Possible values for this field include "rpm", "degrees C", "%", etc.

Calculation Properties: Alarm Tab

If you want this tag to trigger an alarm, use the Add button to open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.
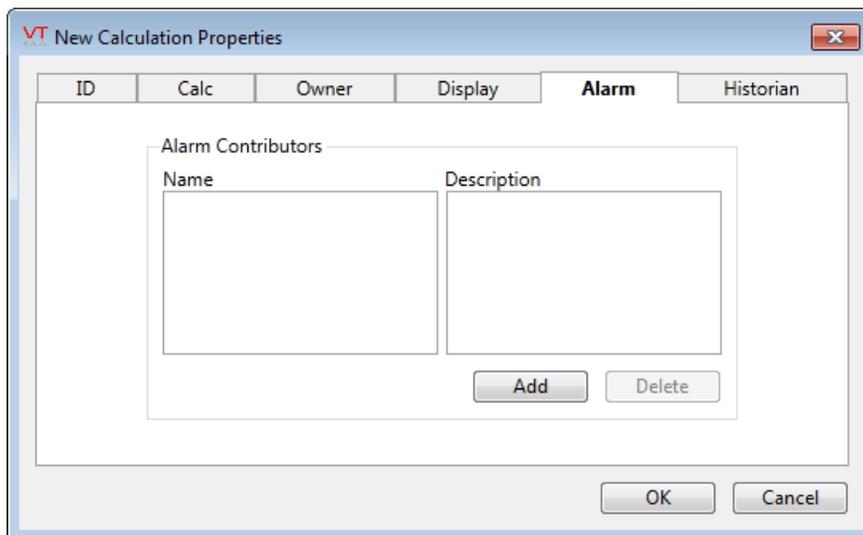
Calculation Properties: Historian Tab

New calculation tags will have a Historian attached automatically. If you are updating a legacy application from a version of VTScada that did not have this option, it is possible that some or all Calculation tags may have

been configured to log via an attached Logger tag. If so, you should leave the current Logger configuration in place.

Note: in previous versions of VTScada, unlogged calculation tags could be plotted in the HDV, where the plot would start at the moment the tag was added to an HDV display and continue until the next restart. Such temporary logging is not possible if the Calculation tag has an attached logger since the values are now being permanently logged. Turning off the Enable Logging feature does not restore the previous behavior. If you have a use for only temporary logging, remove the Historian connection.
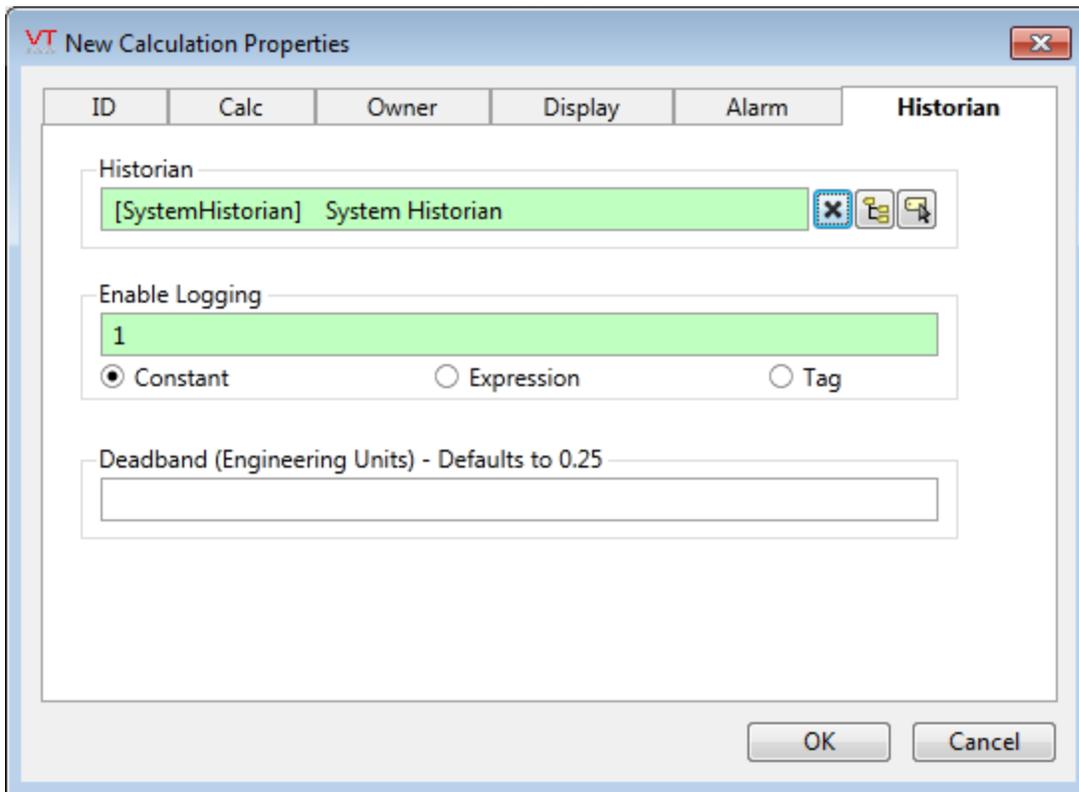
If you select a Historian tag, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historians are described in the topic, Historian Tag.

> Note: Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Enable Logging

If values should be logged only while certain conditions are true, you can tie the Enable Logging option to any tag or expression that will change from zero (logging disabled) to non-zero (usually 1; logging enabled).

Deadband (Engineering Units)

If a Historian is selected, then the value of this tag is automatically logged on change. Use the deadband range to set a minimum amount by which the tag's value must change before a new value is written to the log file. This will dampen out minor changes of value, shrinking the size of the log file by excluding system noise.

If not specified here, the deadband will be the difference between the high and low scale values (set on the Display tab), multiplied by the value of the application property, DefaultCalculationDeadbandFractionOfFullScale. That property defaults to a quarter of a percent (0.0025). If Low Scale Value and High Scale Value have not been set, the deadband calculation will use a scale range of 100.

Calculation Tag Widgets

The following widgets are available to display information about your application's Calculation tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
Compass 1 Widget	Compass 2 Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Equipment / Status Color Indicator	
Image Change Widget	Indicator Light Widget

LCD Bar Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Glass Gauge Widget	Classic Gauge Widget
Retro Gauge Widget	Cockpit Gauge Widget
Aqua Gauge Widget	Army Gauge Widget
Backlit Gauge Widget	Nautical Gauge Widget
Quarter Arc Gauge Widget	Linear Gauge Widget
Antique Gauge Widget	LCD Ring Widget
LCD Arc Widget	LCD Meter Widget
Multi-Color Widget	Multi-Text Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget

Counter Tags

Not counted towards your tag license limit.
Counter tags provide a means to count events such as pump starts or equipment cycles. A count is recorded each time the source changes from 0 to non-zero.

> Note: An invalid value for the source is not assumed to mean a 0 or off state. A transition from invalid to 1 will not be counted.

The Counter tag retains its value between application restarts and uses the last known valid source value when counting. For example, if the data source was 1 (i.e. running) when VTScada stopped and is 1 when VTScada restarts, it is assumed that the equipment continued to run while VTScada was offline and the counter is not incremented. Similarly, if the source was 0 (i.e. off) when VTScada stopped and is 1 when VTScada restarts, it is assumed that the equipment started at least once while VTScada was offline and the counter is incremented by 1.

Note: Do not attempt to attach a Logger tag to a Counter tag. Counter tags have a built in logger.

Also: Note that clicking on a Counter tag, drawn on a page, will not cause the Historical Data Viewer window to open.

Counter Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Counter Properties: Settings Tab

All options to configure the counting behavior of the Counter tag can be found in the settings tab.
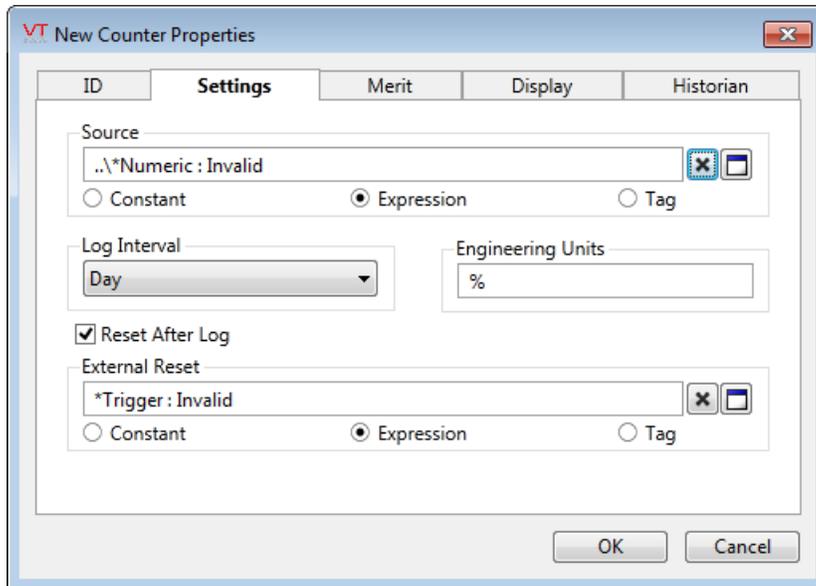
Source

The source can be any tag or calculation whose value will change between 0 and a non-zero value. The count will be increased each time the source changes from 0 to a non-zero. Constants are not useful here, but are a standard part of this type of selector.

Log Interval

The Counter tag will log its value at the end of each interval specified. This interval goes by the clock, not by running time. If, for example, the log interval is set to "Day" the tag's value will be logged at midnight, not at the end of 24 hours of operation. If the log interval is set to "Hour" the tag's value will be logged each hour on the hour. "Weekly" means midnight Sunday night and "Monthly" means at midnight before the first day of the month.

Reset After Log

You can select whether or not the counter should be reset to 0

after its value is logged.

Engineering Units

Text that states what the Counter tag is counting (e.g. Starts).

External Reset

Specify a tag or calculation to force a reset of the counter. Note that the current value of the tag will always be logged before a reset. The reset will occur when this source changes from a 0 or false value to a non-zero or true value. A change from invalid to true does not cause a reset.

Counter Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.
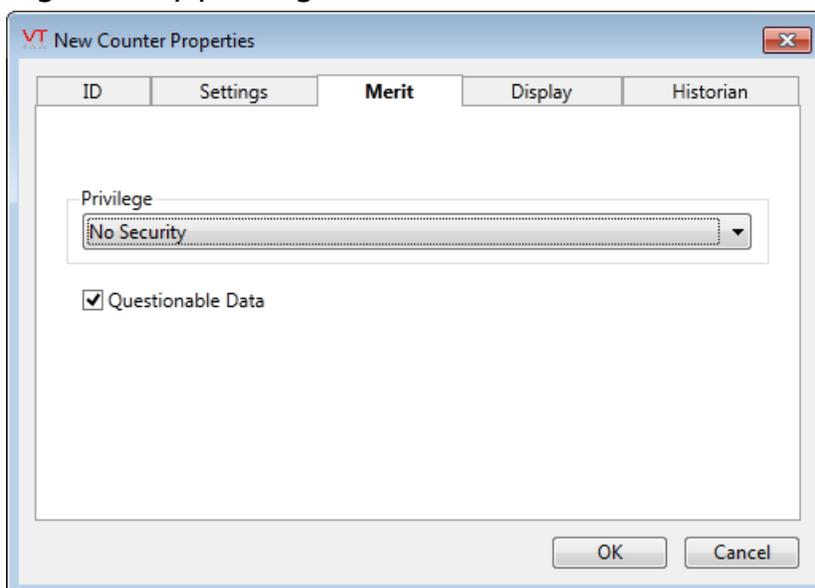
Counter Tag Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
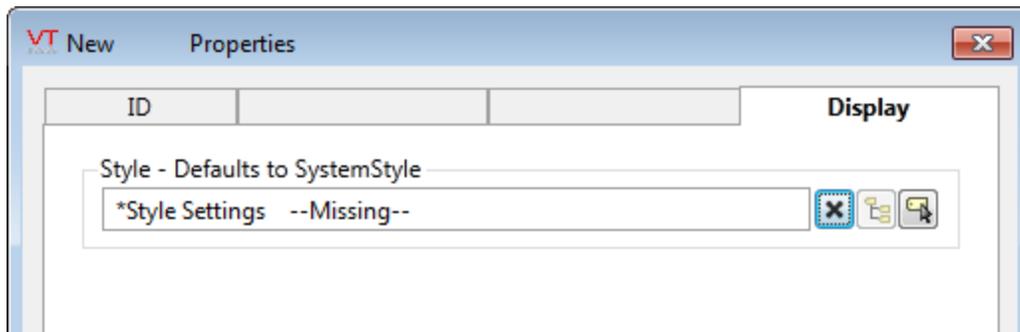
Related:

Style Settings Tags

Counter Properties: Historian Tab
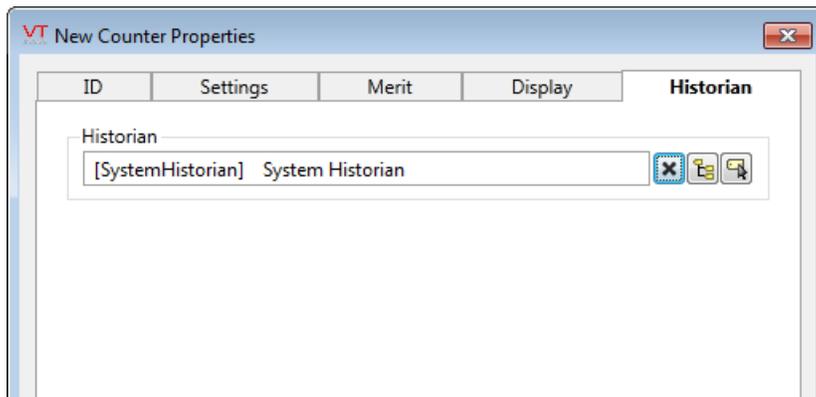
Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are

described in the discussion of the Historian Tags.

> Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Counter Tag Widgets

The following widgets are available to display information about your application's Counter tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Color Blink Widget
Color Box Widget	Color Fill Widget
Color Line Widget	LCD 5x7 Matrix Widget
Draw Text Widget	Equipment / Status Color Indicator
Image Change Widget	Last Logged Value Widget
LCD 7 Segment Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Numeric Value Widget	Multi-Color Widget
Multi-Text Widget	Numeric Entry Widget
Plot Data Widget	Reset Button Widget
Reset Target Widget	Set Analog Value Widget
Text Change Widget	Two Color Bar Widget

Digital Statistics Tags

Not counted towards your tag license limit.
Digital Statistics tags are used to monitor digital tags including Digital Input, Digital Status, Pump Status, Alarm, Roster, etc.
Statistics gathered are user-configurable and include number of starts (zero to non-zero transitions) during selected time frames and total run time (non-zero time) during selected time frames.

The Digital Statistics tag works by launching a child tag for each statistic to be gathered. (Counter tags, Totalizer tags and History Statistics tags) Each child tag may be included separately in a report or Historical Data Viewer plot.

Note: Select only the time frames required. A separate tag will be launched for each statistic and time frame you choose.

Note: By default, the child tags are not logged. You must attach an Historian to each child tag for which you intend to save a record of the calculated statistics.

Note: Do not rely on this tag for statistics of on time or starts that occurred prior to the creation of this tag. The Historian Statistics child tags will use the logged data from the monitored tag but the Totalizer and Counter tags will not.

Digital Statistics Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you

can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Digital Statistics Properties: Source Tab

Used to select the source tag for which statistics are to be calculated.

Source

The Source tab is used to select which digital tag is to be monitored. Selection is limited to tags with numeric values, but only tags whose values have a clear zero to non-zero transition should be selected. Examples include Digital Input, Digital Status and Alarm tags.

Elapsed Time Meter

Your controller may be capable of recording the running time of the equipment being monitored. This is often called the Elapsed Time Meter. If your equipment supports this feature then an analog input (or analog status) tag can be created to read the value from the Elapsed Time Meter. You may then list this value beside the other statistics in your page graphics or reports.

On-Count Units

Similar to the Engineering Units field of analog tags, this value is used for display purposes only. Use it to describe what is

being counted.

Exclude from Site List

Selected by default. You may choose to include this tag in the list of a Sites page as a folder. Operators may view a list of current statistics by clicking the link:
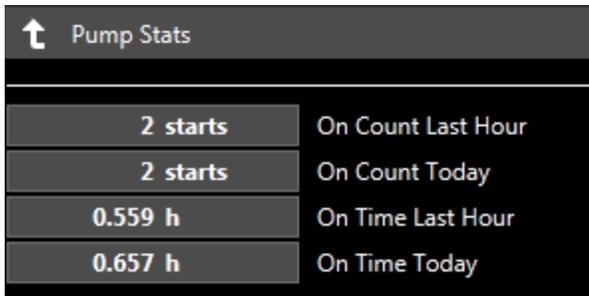
Digital Statistics Properties: On-Count Statistics Tab

The On-Count Statistics tab is used to select the time frames for which you want to gather zero to non-zero transitions of the source tag. By default, only "Last Hour" and "Today" will be selected. Select all that you will need, remembering that a separate tag will be launched for each.

Note: Statistics for the time periods "this week" and "last week" are affected by the application property, StartOfWeek. You may use that property to set the beginning of a week to any day of your choice.

- "Today" means since the time 00:00 this day.
- "This Week" means since 00:00 of the first morning of the week.
- "This Month" means since 00:00 on the morning of the first day of the month.
- "Yesterday" means the 24 day, prior to today.
- "Last week" means the full week prior to the current week.
- "Last Month" means the calendar month prior to the current one.
- "Last 24 Hours" covers the 24 hours prior to "right now".

Digital Statistics Properties: On-Time Statistics Tab

The On-Time Statistics tab is used to select the time frames for which you want to gather the non-zero (running) time of the source tag. By default, only "Last Hour" and "Today" will be selected. Select all that you will need, remembering that a separate tag will be launched for each.

Note: Statistics for the time periods "this week" and "last week" are affected by the application property, StartOfWeek. You may use that property to set the beginning of a week to any day of your choice.

- "Today" means since the time 00:00 this day.
- "This Week" means since 00:00 of the first morning of the week.
- "This Month" means since 00:00 on the morning of the first day of the month.
- "Yesterday" means the 24 day, prior to today.

- "Last week" means the full week prior to the current week.
- "Last Month" means the calendar month prior to the current one.
- "Last 24 Hours" covers the 24 hours prior to "right now".

Digital Statistics Tag Widgets

The following widgets are available to display information about your application's Digital Statistics tags:
Tag List Widget

Function Tags

Not counted towards your tag license limit.
The Function tag type is used to perform mathematical and logical calculations using numeric values and, optionally, the values of other tags. A Function tag can accept up to four tag values or numeric values as properties, and the result of the calculation becomes the value of the Function tag. Other tags can use the value of a Function tag.
The output of a Function tag with no configured function parameters is undefined. It is not guaranteed to be any specific value, not even invalid.

> Note: Function tags are an older technology, maintained for backward-compatibility. You are advised to use Calculation tags for most applications.

Function Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Function Properties: Operator Tab

The Operator tab identifies the operation that will be performed on the inputs for this tag.
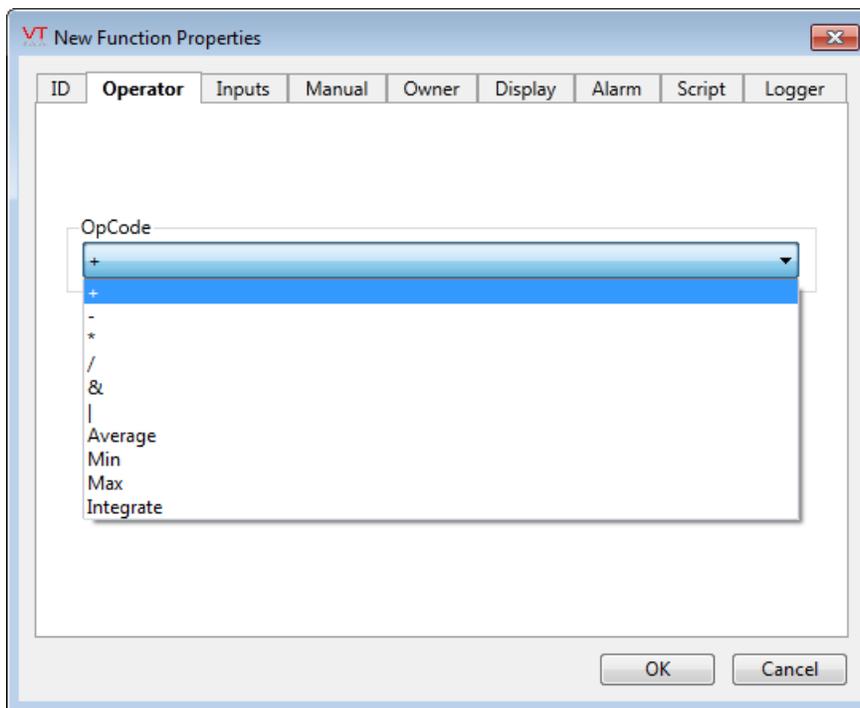
OpCode

The OpCode drop-down list enables you to specify the type of operation to be performed on the values associated with this Function tag.

The parameters (values) that will be used for the selected operation, are configured on the tab, Inputs.

- Parameters that are not configured have no impact on the resulting value of the function.
- Parameters whose value evaluates to Invalid will cause most function operations to return Invalid.

The supported operations are:

+	All configured parameters are added. Any parameter that contains a tag or expression that evaluates to Invalid will result in the Function tag having the value, Invalid.
−	Parameter 1 – Parameter 2 – Parameter 3 – Parameter 4 (disregarding any unconfigured parameters). Any parameter that contains a tag or expression that evaluates to Invalid will result in the Function tag having the value, Invalid.
*	Parameter 1 * Parameter 2 * Parameter 3 * Parameter 4 (disregarding any unconfigured parameters). Any parameter that contains a tag or expression that evaluates to Invalid will result in the Function tag having the value, Invalid.
/	Parameter 1 / Parameter 2 / Parameter 3 / Parameter 4 (disregarding any unconfigured parameters). Any parameter that contains a tag or expression that evaluates to Invalid will result in the Function tag having the value, Invalid.
&	Logical AND. Three results are possible: TRUE, FALSE or Invalid. * To return a value of TRUE, all configured parameters must evaluate to TRUE. * If any configured parameter is FALSE, then the result will be FALSE. * If no configured parameters evaluate to FALSE, but at least one evaluates to Invalid, then the result will be Invalid.
|	Logical OR. Logical Three results are possible: TRUE, FALSE or Invalid. * If any configured parameter is TRUE, then the result will be

TRUE.

* If all configured parameters are FALSE, then the result will be FALSE.

* If no configured parameters evaluate to TRUE, but at least one evaluates to Invalid, then the result will be Invalid.

Average	Average of Parameters 1 through 4 (excluding any undefined values).
Max	Maximum of Parameters 1 through 4 (excluding any undefined values).
Min	Minimum of Function Parameters 1 through 4 (excluding any undefined values).
Integrate	Performs time integral of Parameter 1 * Parameter 2, every Parameter 3 seconds. * If Parameter 2 is not defined, it defaults to a value of 1. * If Parameter 3 is not defined, it defaults to a value of 1 second. For example, if Parameter 2 and Parameter 3 were not defined, the calculation would be Parameter 1 * 1 every 1 second).

Function Properties: Inputs Tab

The Inputs tab identifies the numeric values or tag values that will be factors in the calculation.
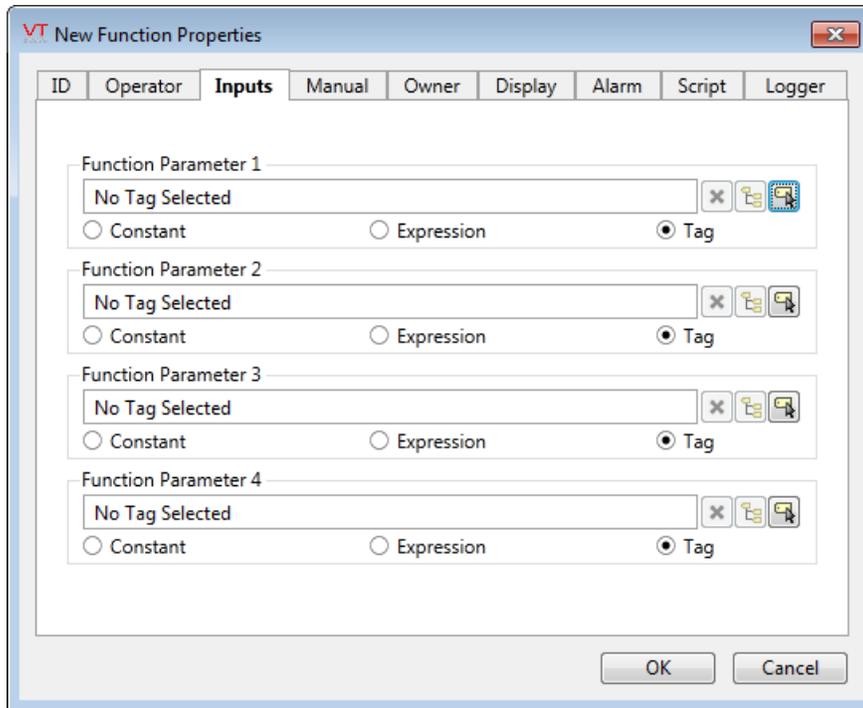
Function Parameter X

The Function Parameter 1 through to Function Parameter 4 fields enable you to specify up to four factors in the calculation performed by this tag.

You may use a constant, expression or tag as the source of the value for each parameter. Parameters that are not configured are essentially ignored, in that they will not have an effect on the result of the calculation.

Parameters that are configured, but which evaluate to Invalid, will have an effect on the calculation as noted in the description of each operational code. Note that tags can have an invalid value when there is an error in communication. Many expressions can return a value of Invalid. For example, if attempting to divide by zero.

Note: You cannot select a Function tag as one of its own function parameters.

Function Properties: Manual Tab

The Manual tab contains the Manual Data and Questionable Data properties. The Manual Data field can be used to enter a user-defined value for this tag that will override the value of its calculation. Use the Questionable Data check box to flag this tag's data as being questionable in the event that you feel the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is monitored for validity.

Manual Data

Enables you to set a user-defined value for this tag, rather than reading the value from the associated I/O device (i.e. this user-defined value overrides the data incoming from the equipment).
If this field is anything other than a blank, or an invalid number, then the value from the I/O device is ignored and this value will be used instead.
The widget used by any tag with Manual Data will include an exclamation mark.

Note: If the value entered into the Manual Data field is a 0, the actual value reported by the equipment is also overridden, as 0 is considered a valid value. Therefore, if it is your intention to resume reading data from the I/O device, you must clear the Manual Data field entirely.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

Function Properties: Owner Tab
This tag can be used in an owner/contributor structure where multiple

contributor tags can supply their values to an owner tag.

Note: There is no specific "owner" tag type, rather an owner tag is typically a custom-designed tag that is created using VTScada scripting code.

Owner

Specify a tag to which this contributor should supply its data. An owner tag is one which you must design and then create, using the VTScada scripting language.

The owner tag may keep track of different aspects of each contributor's data, from the presence of a user-defined manual data value, to questionable data, according to the configuration of the check boxes appearing beneath the Owner field. These check boxes also determine the way that this contributor tag's value should be used in the owner tag's calculations.

Set Owner\DataX(...) to Value

When selected, the Set Owner\DataX[…] To Value check box is used to set the value of this contributor tag as the nth element in the owner tag's array. You may choose to set this contributor tag's value in more than one of the owner tag's array elements if required.

Set Active/Unack. Priority

An owner tag may keep track of the alarm priority and status of its contributors. When selected, the Set Active/Unack. Priority check box causes the owner tag to keep track of the priority of the contributor's active alarm (or records an Invalid if the contributor is not in an alarm state). Selecting the Set Active/Unack. Priority check box also causes the owner tag to record whether or not the alarm has been acknowledged.

Record Use of Manual Data

An owner tag may keep track of the number of contributor tags that are providing manual data (user-defined values), rather than reading data from their I/O device. When selected, the Record Use of Manual Data check box is used to increment the owner's count of the number of tags that are contributing manual data by 1 when manual data has been provided for this contributor, and decrement this count by 1 when no manual data value has been specified.

Record Data Quality

An owner tag may keep track of the quality of the data for each of its contributors. When selected, the Record Tag Quality check box is used to increment the owner tag's count of the number of tags that are contributing quality data by 1, and decrement this count by 1 when this contributor is not

supplying quality data.

Record Tag Validity

An owner tag may keep track of the questionable status of the data for each of its contributors. When selected, the Record Tag Validity check box is used to increment the owner tag's count of the number of tags that are contributing questionable data by 1, and decrement this count by 1 when this contributor is not supplying questionable data.

Function Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
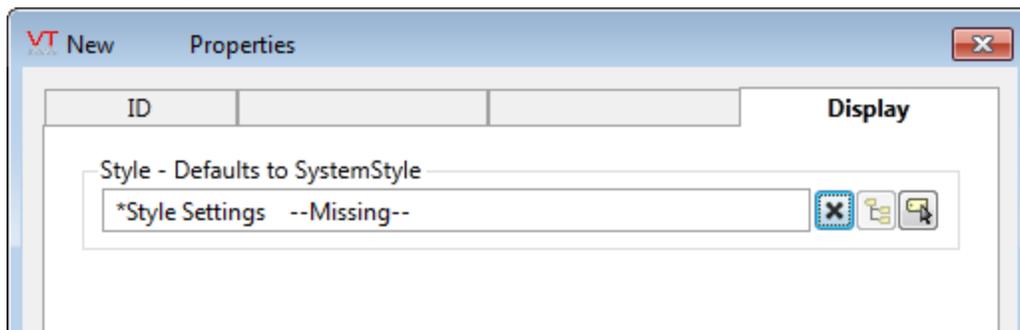
Related:

Style Settings Tags

Function Properties: Alarm Tab

If you want this tag to trigger an alarm, use the Add button to open a configuration panel for a new Alarm tag. The triggered-by field for the new alarm will automatically be linked to this tag's value. The new alarm tags will be created as children of the current tag.

Function Properties: Script Tab

Optionally, link one or more Script tags to this tag. A Script tag provides a means of creating a procedure, using VTScada's programming language, that will run whenever this tag's value changes.

Function Properties: Logger Tab

The logger tab enables you to associate a single Logger tag with this tag. The Logger tag works with a Historian to this tag's data to disk so that it can be plotted on the Historical Data Viewer page. The new logger tag will be created as a child of the current tag.

Note: Only one Logger tag can be directly associated with a single input or output tag. If you need to have multiple loggers with different logging rates, please refer to Using Function Tags to Create Multiple Data Logs of an I/O Tag

Function Tag Widgets

The following widgets are available to display information about your application's Function tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Box Widget	Color Fill Widget
Color Line Widget	LCD 5x7 Matrix Widget
Equipment / Status Color Indicator	
Image Change Widget	Indicator Light Widget
LCD 7 Segment Widget	Left Bar Widget
Meter 1 Widget	Meter 2 Widget
Meter 3 Widget	Multi-Color Widget
Multi-Text Widget	Numeric Value Widget
Plot Data Widget	Right Bar Widget
Text Change Widget	Top Bar Widget
Two Color Bar Widget	

History Statistics Tag

Not counted towards your tag license limit.

Each instance of a History Statistics tag will perform 7 statistical calculations, using the data collected from a numeric tag's value over a defined time period. History Statistics tags complement, rather than replace, Totalizer tags and Counter tags. The result of only 1 calculation will be shown, but you may display other calculations by linking other History Statistics tags to the first – these will simply take the result of the requested calculation from the first tag rather than performing calculations of their own.

The available calculations include:

- Time-weighted average – average value over a specified length of time.
- Minimum value during period
- Maximum value during period
- Value at start
- Count of zero to non-zero transitions (e.g. equipment starts)
- Non-zero time (example: total time that a pump was running during the period)
- Totalizer, calculated as the sum of the values logged during the time period. Note that this calculation differs substantially from that used by Totalizer tags.

Any numeric tag may be used as the data source, so long as its value is logged. The exception is if the data source is another History Statistics tag. A warning will appear if you attempt to use a data source that is not being logged.

Statistics are collected over a given duration of time (the Manual Data Period option in the tag's configuration). You provide the length of the time period and the frequency at which the display is to be updated. Time durations fall on regular clock intervals. For example, a manual data period set for the latest 1 hour, updated every 15 minutes, will use logged data for the 60 minutes prior to the most recent quarter-hour mark.

Use care if attempting to use a Digital Output tag as a data source for counting 0 to 1 transitions (equipment starts). While the output tag may be writing a 1 to the equipment, the value of the Digital Output tag itself will only be 1 if it is configured for a pulse duration, or if it has a feedback configured and there is a delay between when the output writes and the feedback shows that the value has been written. If neither of these is true, then the Digital Output tag's value will remain 0 and you will not get the number of times that it wrote a start value to the equipment.

The calculated value of the History Statistics tag can be displayed on a page using any of the Analog widgets. While a History Statistics tag can be selected for inclusion in the Historical Data Viewer (HDV), clicking on one that is drawn on a page will not cause the HDV to open. Note that if you want to save the values calculated by the History Statistics tag, you will need to select a Historian to use for the built-in connection.

History Statistics Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

History Statistics Properties: Source Settings Tab

Use this tab to select the logged tag whose value is to be used for the calculation, and the particular calculation to be performed by this instance of the History Statistics tag.

Every standard VTScada tag has only one source variable that can be used for the calculation: its value. Custom tag types may expose other variables that can be used in the calculation.
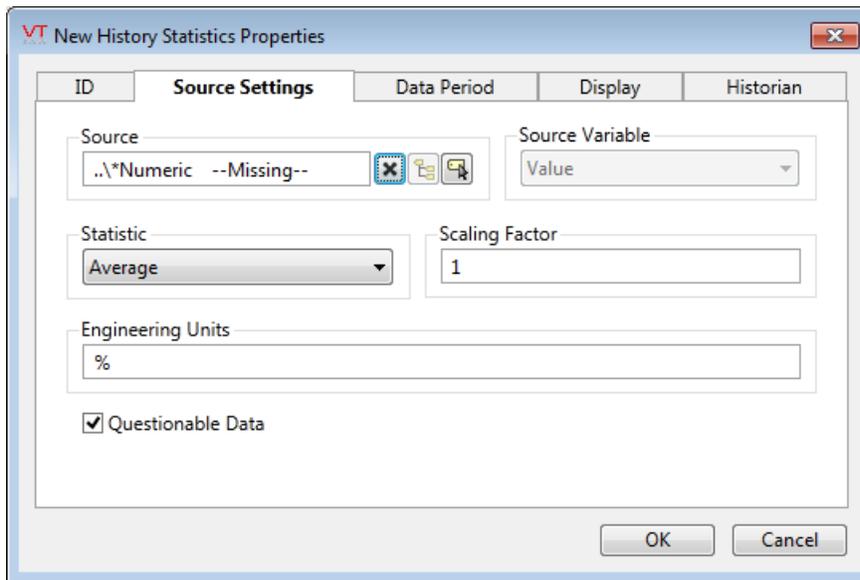
Source

Select the tag whose value (or other source variable) is to be used in the calculation. All numeric tag types are available. By default, the nearest Numeric parent tag will be selected. You can select another History Statistics tag here if you wish to display a different one of its 7 calculations. Selecting another History Statistics tag as the data source will disable the Data Period options for this tag instance.

Source Variable

For all standard VTScada tags, the only source variable is the tag's value. Custom tags, written specifically for your application, may make other variables available. If so, the drop-

down list will be populated with the available choices.

Statistic

Select the calculation to be performed from the following list:

- Average – time weighted average over the data period.
- Minimum value during the data period.
- Maximum value during the data period.
- Value at start of the data period.
- Zero to non-zero transitions during the data period.
- Non-zero time. Example: total time that a pump was running during the period.
- Totalizer – the sum of the values collected during the data period.

Difference from the Totalizer tag:
The Totalizer tag calculates a time-weighted integral of the values collected. The totalizer option of the History Statistics tag provides a sum of the values within the time interval. Both have their uses.

Scaling Factor

The calculated value will be multiplied by the factor you specify in this field. Use for applying a scaling factor to the value, for example if correcting for raw values that are measured in terms of minutes or hours instead of seconds. Defaults to 1.

Engineering Units

Text that will indicate to operators what the value of this tag means.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that

its data is marked for extra monitoring.

History Statistics Properties: Data Period Tab

This panel is used to set the length of the calculation period, and the rate at which the calculation is updated.

If another History Statistics tag has been selected as the data source, all fields in this panel will be disabled. The source History Statistics tag will control the data period.
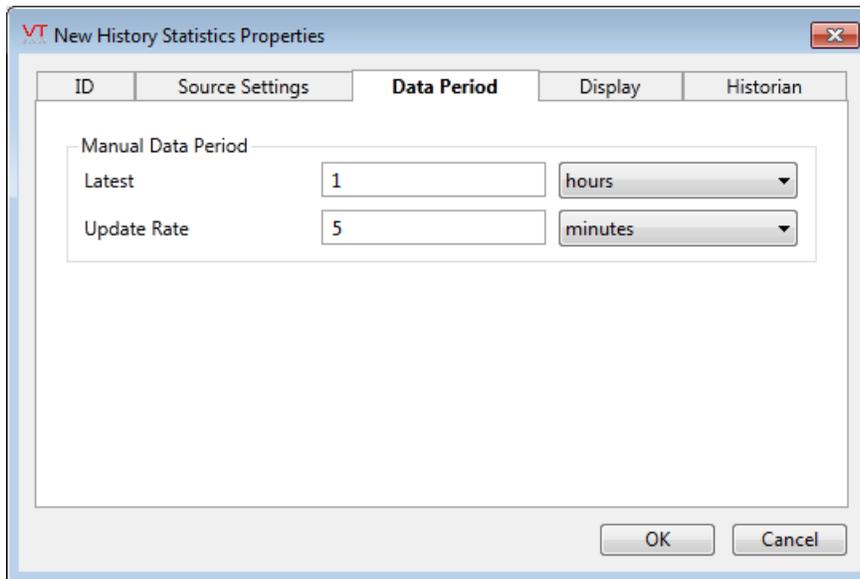
Latest

Use the text field to set a number of time units. The statistic calculation will use data from this length of time, measured to the nearest clock interval as controlled by the update rate.

Update Rate

Sets the frequency at which the statistic is recalculated. The calculation will always use data from the full length of time specified by the Latest field, ending on the clock interval specified by this update rate. For example, if the update rate is 5 minutes, the update will occur at the 5 minute mark, 10

minute mark, etc.

The update rate must not be greater than the duration of time over which the statistic is being calculated, and can be no smaller than 0.1% of that time period.

History Statistics Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
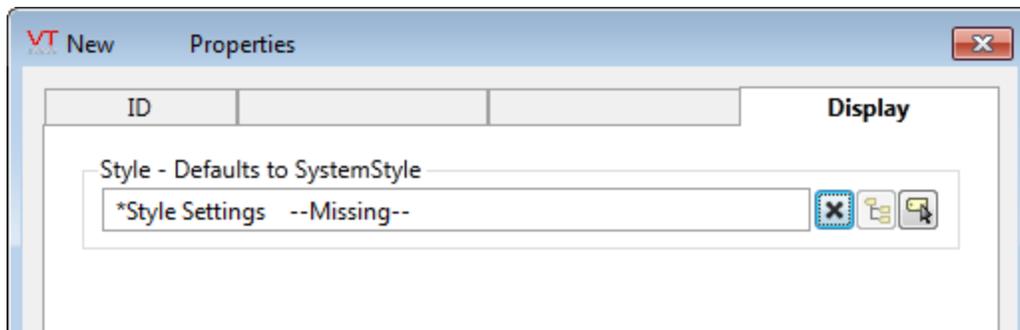
Related:

Style Settings Tags

History Statistics Properties: Historian Tab

Optionally, link a Historian tag to this History Statistics tag in order to save a record of the summary data calculated by the History Statistics tag. Logging is done at the interval configured in the data period tab. In a remote application, logging is broadcast to all machines running the application.

Note that the History Statistics tag uses logged data for its calculations, thus logging the result of those calculations is redundant. Doing so may simplify later report generation, but at the cost of increased storage.
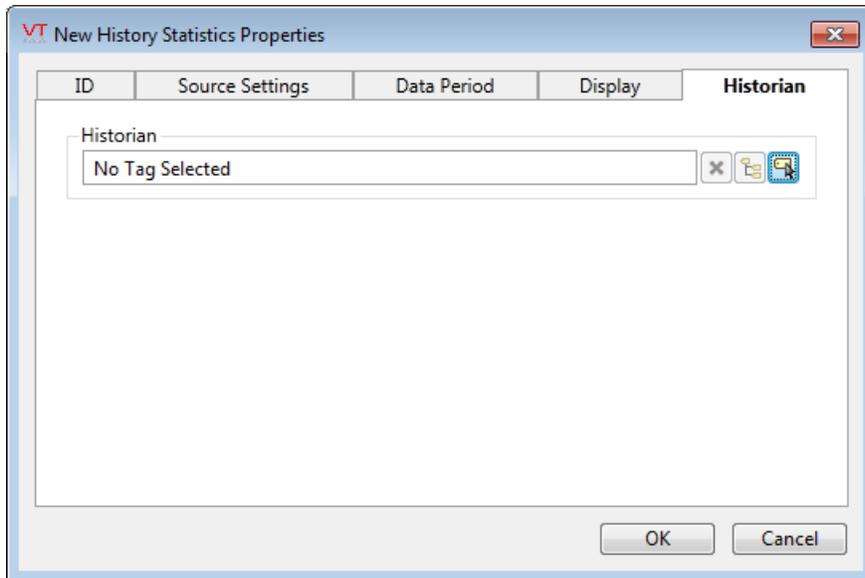
History Statistics Tag Widgets

The following widgets are available to display information about your application's History Statistics tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Blink Widget	Color Box Widget
Color Fill Widget	Color Line Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Equipment / Status Color Indicator	
Image Change Widget	Indicator Light Widget
LCD Bar Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Glass Gauge Widget	Classic Gauge Widget
Retro Gauge Widget	Cockpit Gauge Widget
Aqua Gauge Widget	Army Gauge Widget
Backlit Gauge Widget	Nautical Gauge Widget
Quarter Arc Gauge Widget	Linear Gauge Widget

Antique Gauge Widget	LCD Ring Widget
LCD Arc Widget	LCD Meter Widget
Multi-Color Widget	Multi-Text Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Text Change Widget
Top Bar Widget	Two Color Bar Widget

Network Status Tags

Not counted towards your tag license limit.
These are used to inquire about the connection status between two computers (usually a server and a workstation). A 1 is returned for a good connection, 0 for no connection and Invalid if no information can be determined.

Network Status Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
>   Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area
>   The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Network Status Properties: Settings Tab

The Settings tab for a Network Status tag is used to specify which work-stations you want to monitor. You must provide the names of the machines to monitor. IP and Subnet are required only if separate connections between the computers are possible. In this case, you must tell VTScada which connection to monitor.

Machine 1 Name

Specify the name of one of the two workstations, the connection between which is to be monitored. You may enter the name of the workstation in this field, or select the name of the workstation from the drop-down list.

Machine 1 IP

The Machine 1 IP field enables you to enter the IP address of the first workstation to be monitored by this tag.

Machine 1 Subnet

The Machine 1 Subnet field enables you to enter the subnet for the first workstation to be monitored by this tag.

Machine 2 Name, IP and Subnet

These three fields match those of Machine 1. They enable you to specify the name, IP address and subnet of the second workstation to be monitored by this tag.

Manual Data

The Manual Data property enables you to enter a value for this tag, rather than reading the value from the monitored workstations. In this instance, the Manual Data property may be set to 0, indicating that the workstation is not connected; or 1, indicating that the workstation is connected.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

Network Status Properties: Alarm Setup Tab
Use the Alarm Setup tab for a Network Status tag to configure alarm conditions for this tag.
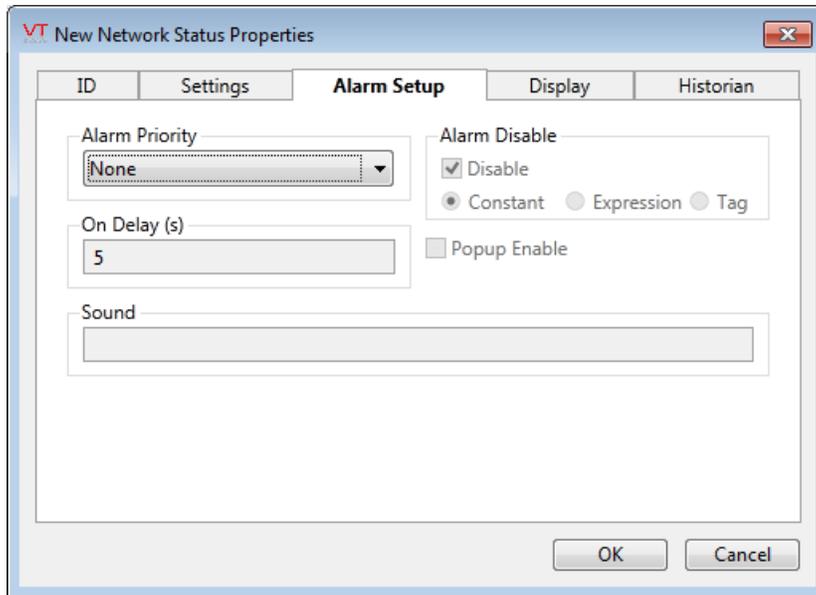
Alarm Priority

The Alarm Priority drop-down list enables you to select the priority of the alarm for this tag, should it enter an alarm state. The available priorities are:

None (alarm not configured)

0 – Event

1 – Critical

2 – High

3 – Warning

4 – Notice

On Delay

The Delay field enables you to enter the amount of time (in seconds or fractions of a second) that the system will wait before triggering an alarm for this tag. This tag must therefore be in an alarm state for the amount of time specified in the Delay field before an alarm will be indicated.

Sound

The Sound field enables you to identify what sound will be played when this alarm is triggered. The Sound field can be set to a 0, 1, or to the name of a .WAV sound file to be played.
If the Sound field is set to 0, no sound will be played when this alarm is triggered.
If the Sound field is set to 1, an alarm sound whose properties are configured on the associated Alarm Priority tag will be played.
If the Sound field identifies the name of a .WAV sound file, it will override any alarm sound configured for the associated Alarm Priority tag. When specifying a sound file, you must enter its name and extension (e.g. MySound.wav). The specified sound file must be a .WAV file, and must be stored in the application directory. If the specified sound file is not found, the alarm will revert to using tones as specified in the associated Alarm Priority tag.

Alarm Disable

The Alarm Disable check box enables you to disable and enable the alarm for this tag. Alarm disabling is typically used in situations where routine maintenance is being performed, or when you are aware that another interruption in communications will occur for a period of time. In such a situation, an alarm can be disabled until the maintenance is complete and communications are reestablished.
May be linked to a tag or expression for suppression by design.

Pop-up Enable

If the application property AlarmPopupsEnable is set to 1, then checking Pop-up Enable will result in a pop-up dialog being displayed whenever the alarm is triggered. It is strongly

suggested that this feature be used sparingly.

Network Status Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
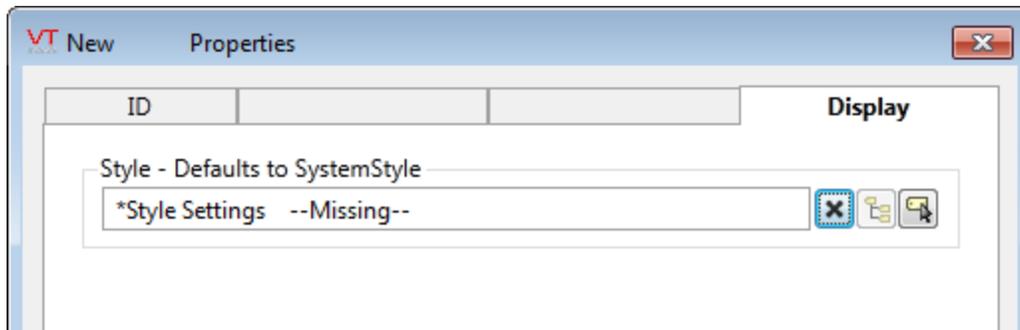
Related:

Style Settings Tags

Network Status Properties: Historian Tab
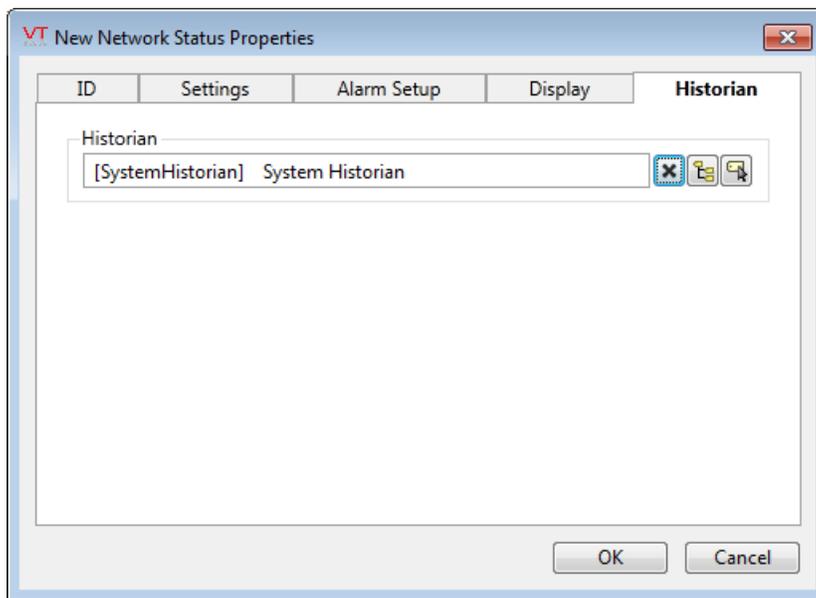
Historian

> If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Network Status Tag Widgets

The following widgets are available to display information about your application's Network Status tags:

...Alarm Priority Box

...Alarm Priority Icon

...Station and Site Tags

...Color Blink Widget

...Color Box Widget

...Equipment / Status Color Indicator

...Image Change Widget

...Indicator Light Widget

...Network Link Widget

...Plot Data Widget

...Text Change Widget

...Two Color Bar Widget

Rate of Change Tags

Not counted towards your tag license limit.

Rate of Change tags are used to display how quickly another tag's value is changing. They can be used to watch for system problems such as leaks and blockages.

The Rate of Change tag calculates the first derivative of another tag's value. In general terms, this is defined as the change in value divided by the change in time. A steady flow has a rate of change of zero.

> Note: The source tag's value must be logged in order for the Rate of Change tag to work properly.

The rate of change will be positive if the source is increasing in value, and negative if the source tag's value is decreasing. The tag can be configured to report the absolute value instead, thus showing a positive value for any rate of change.

For example, the water level in a large holding tank would normally change slowly unless it started to leak. A tank feeding a process would be expected to have a level decreasing at a minimum rate unless the outlet became blocked. A Rate of Change tag can be configured for either situation in order to detect departures from an expected rate of change. Alarms built into the tag can be triggered when the rate is too fast or too slow.

Note that in remote applications, the Rate of Change tag information is distributed from the server to workstations on a regular interval (default: 60 seconds) or when the value changes by a given percentage (default: 5%). These values are controlled by variables in your configuration file.

Related Information:

See variables: RateOfChangeTagRPCThreshold, and RateOfChangeTagRPCInterval in the VTScada Admin Guide.

Rate of Change Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Rate of Change Properties: Settings Tab

The Settings tab is used to select which tag is to be watched and how often it is to be checked.
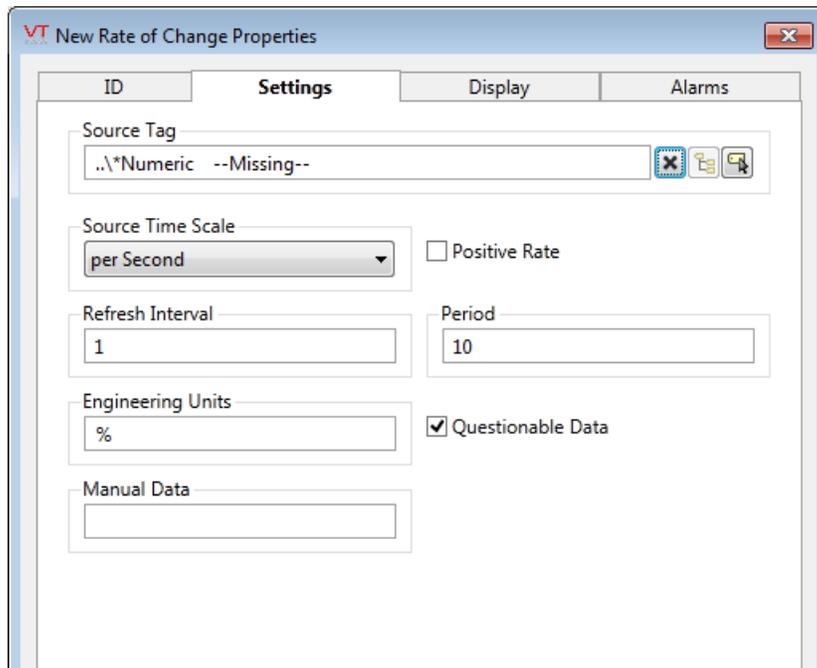
Source

Use this option to select the tag to be monitored for changes in value.

If you would like to monitor the rate of change of some combination of tags or a calculated value, you should create a Calculation tag with the expression you need and use it as the source.

Source Time Scale

Sets the measurement rate for the change. Available options include: per Second, per Minute and per Hour. For example...

Source	Source Time Scale	Rate of Change Expressed in:
Level (inches)	Hours	inches/hour
Pressure (psi)	Minutes	psi/min
Flow (liters/second)	Seconds	l/s/s

Positive Rate

When checked, the absolute value of the rate of change will be shown. Otherwise, a decreasing value in the source tag will be displayed as a negative rate of change.

Refresh Interval and Period

- The Refresh Interval sets how often the rate of change is calculated. This value is always measured in seconds.
- The Period defines the length of time over which the change is to be measured. It is also measured in seconds.

The rate of change is roughly defined as the amount by which the value has changed during a given period, divided by the length of time in the period. To state it more accurately, the rate of change is the calculated derivative (dy/dx) of the plotted value.
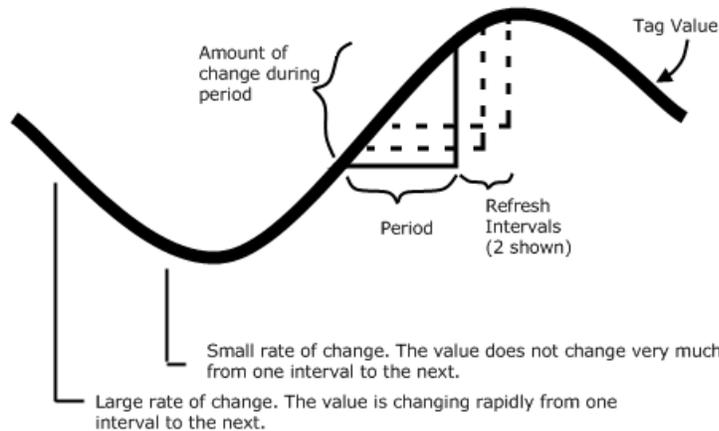
Note: The rate of change calculation is affected by data noise and transient spikes. The period should always be set to a value larger than the frequency of data noise. To avoid alarms due to transient noise, set a delay on the alarm. (see: Rate of Change Properties: Alarms Tab.)

Engineering Units

Used by various widgets. Set this field to display the units of the rate of change.

Questionable Data

Use this field to flag the tag's data in the event that you suspect the values it is reporting might not be correct, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.

Manual Data

This optional field will provide a constant value that will be used instead of input read by the communication driver. It is commonly used when testing a new tag while the application is not attached to a live data source. Tags that are using manual data are marked on screen by a flashing exclamation mark

within any linked widgets.

Rate of Change Properties: Display Tab

The controls in the display tab are used only in connection with the Historical Data Viewer (HDV). The Displayed Value Minimum and Maximum set the initial limits of the y-axis for this tag when it is viewed in the HDV.

These fields do not scale the data.
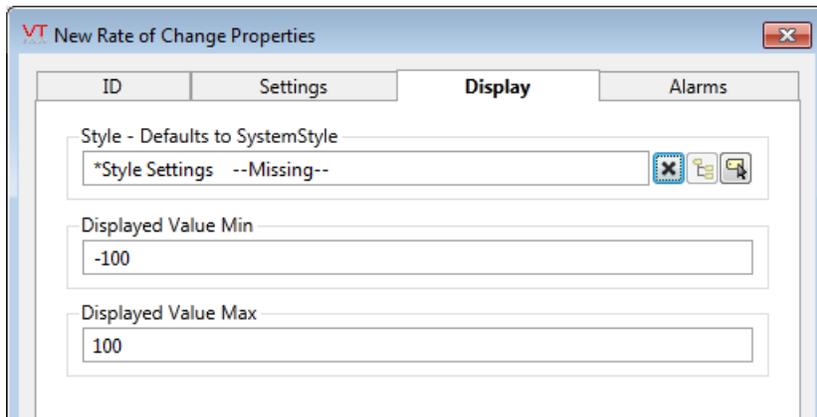
Rate of Change Properties: Alarms Tab

You can set alarms to notify operators when the rate of change is either too slow or too fast.

High alarms are triggered when the rate matches or exceeds a set point. Depending on whether the Positive Rate option is checked on the Settings tab, the Low Alarm can be used either to warn on large negative changes, or to warn when the rate of change falls below a desired minimum.
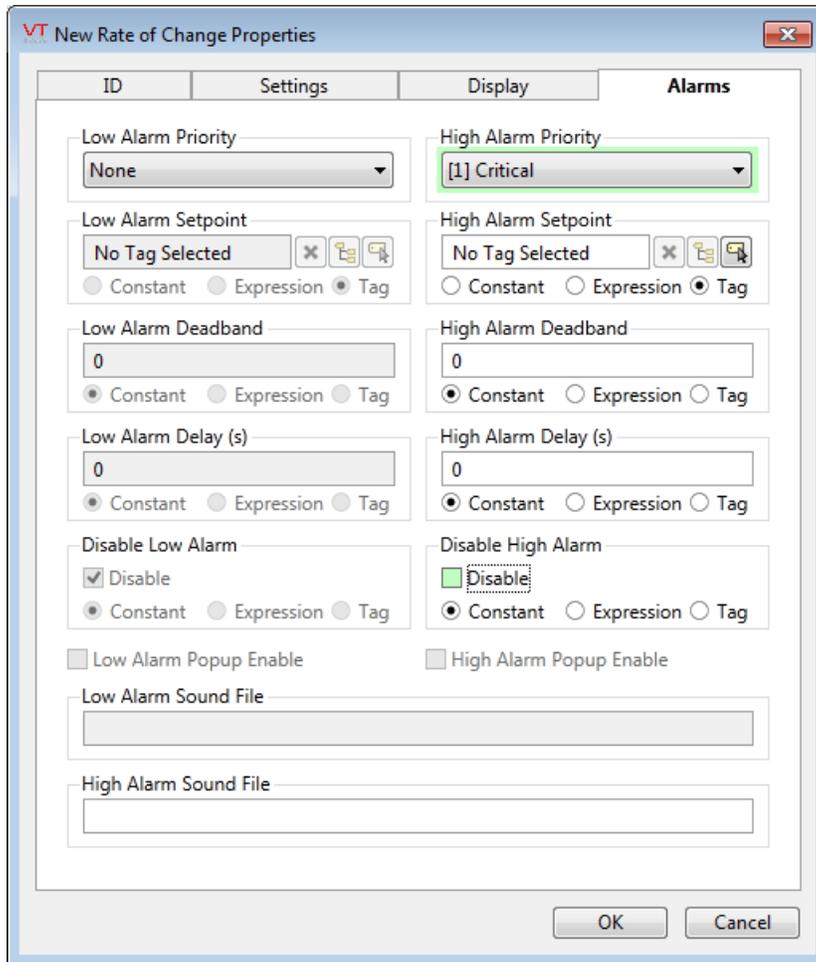
Low and High Alarm Priority

The Low and High Alarm Priority drop-down lists are used to select the priority of the alarms that will be triggered for this tag. The default priority is None, indicating that neither alarm has been configured. The available priorities are:

None
0 – Event
1 – Critical Alarm
2 – High Alarm
3 – Warning Alarm
4 – Notice Alarm

If you have defined your own Alarm Priority tags, those will also be available for selection.

Low and High Alarm Setpoints

Use the Low Alarm Setpoint field to enter the lowest value that this rate of change value will be permitted to attain, after which a low alarm will be triggered. Correspondingly, the High Alarm Setpoint is used to set a value at which the high alarm should be triggered.
If the Positive Rate option is selected on the Settings tab, all rates of change will be positive. In this situation, the high alarm is used to warn of excessively high rates of change (whether the changing value is rising or falling). The low alarm could be used to warn of change rates that are too slow.
If the Positive Rate option is not selected, the high alarm will warn only of excessive change rates for values that are rising. The low alarm set point should be set to a negative value to warn of excessive change rates for values that are falling.
The source for these fields may be either a constant, the result of an expression, or the value of a tag. Please see the topic: Constant, Expression or Tag for more information on selecting between these three choices.

Low and High Alarm Deadbands

If noise in the system causes a value to fluctuate above and below the set point, the alarm will repeatedly switch between active and inactive status. By using the deadband, you can dampen out these fluctuations by setting an amount by which the tag's value must retreat back past the set point before the alarm state is considered to no longer be active.

For the low alarm, this means that the value must rise <dead-band> units above the set point. For the high alarm, the value must fall by <deadband> units below the set point.
The deadband is used only for changing a current alarm's active status. It does not affect the set point.

Low and High Alarm Delays

Noise in the value being monitored may cause transient spikes or drops in the reported rate of change. To avoid having alarms triggered by system noise, you can set a delay in seconds. The value must meet or exceed the alarm set point for the length of the delay time before an alarm will be activated.

Disable Low or High Alarm

The Disable … Alarm field is used to specify whether either the low alarm or high alarm for this tag is enabled. Disabling of alarms is typically used in situations where you wish to avoid false alarms. For example, in the event that routine maintenance is being performed on the equipment represented by this tag, the alarm can be disabled until the maintenance is complete and communications are re-established.
The value can be provided via any of a constant, an expression, or a tag, where 0 means "not disabled" and 1 means "disabled. Please see the topic: Constant, Expression or Tag for help selecting which to use.

Low Alarm Pop-up Enable & High Alarm Pop-up Enable

If the configuration variable AlarmPopupsEnable is set to 1, then setting either the Low Alarm Pop-up Enable or the High Alarm Pop-up Enable, will result in a pop-up dialog being

displayed whenever the respective alarm is triggered. It is strongly suggested that this feature be used sparingly.

Sound

The Sound fields can be used to set the sound that will be played when either the low alarm or the high alarm is triggered. These fields can be set to blank, 0, 1, or to the name of a .WAV sound file to be played.
If a Sound field is set to 0, no sound will be played when this alarm is triggered.
If a Sound field is set to blank or 1, an alarm sound whose properties are configured on the associated Alarm Priority tag will be played.
If a Sound field contains the name of a .WAV sound file, it will override any alarm sound configured for the associated Alarm Priority tag. When specifying a sound file, you must enter its name and extension (e.g. MySound.wav). The specified sound file must be a .WAV file, and must be stored in the application directory. If the specified sound file is not found, the alarm will revert to using tones as specified in the associated Alarm Priority tag.

Rate of Change Error Dialogs
The period must be set to a value that is larger than the refresh interval for the Rate of Change tag to work properly. Attempting to set either value such that this is no longer the case will result in the following error dialog:
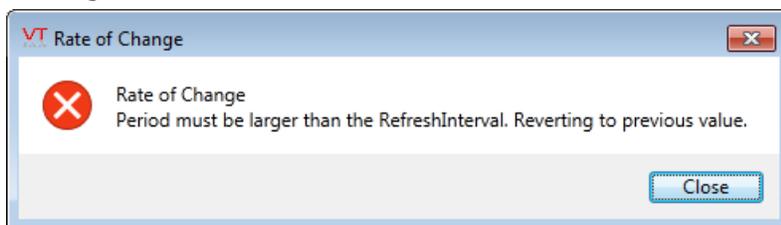
Acknowledge the dialog by clicking on the Close button, then adjust the complementary value (either a larger period or a smaller refresh interval) before continuing to set the value you were first attempting to set.

Rate of Change Tag Widgets

The following widgets are available to display information about your application's Rate of Change tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image WidgetColor Blink Widget	Color Box Widget
Color Fill Widget	Comm Line Widget
LCD 5x7 Matrix Widget	Draw Text Widget
Equipment / Status Color Indicator	Image Change Widget
Indicator Light Widget	LCD 7 Segment Widget
Meter 1 Widget	Meter 2 Widget
Meter 3 Widget	Glass Gauge Widget
Classic Gauge Widget	Retro Gauge Widget
Cockpit Gauge Widget	Aqua Gauge Widget
Army Gauge Widget	Backlit Gauge Widget
Nautical Gauge Widget	Quarter Arc Gauge Widget
Linear Gauge Widget	Antique Gauge Widget
LCD Ring Widget	LCD Arc Widget
LCD Meter Widget	Multi-Color Widget
Multi-Text Widget	Numeric Value Widget
Plot Data Widget	Text Change Widget
Two Color Bar Widget	

Totalizer Tags

Not counted towards your tag license limit.

The Totalizer tag provides a means of accumulating the sum of another tag's (or calculation's) value over time. To put it more precisely, the totalizer integrates a value over a time period.

The totalizer looks for its source to represent values "per second", "per minute" or "per hour". It is important to select a source time scale that matches the source values. For example, when recording total flow from a meter that is measuring flow rate per second, selecting the "per minute" source time scale is guaranteed to produce results that are incorrect.

You can also use the totalizer to record running time. For this, the source value must be a 1 when the equipment is running and a 0 when stopped. Given a source time scale of "per second" this will add 1 second, per second while the equipment is running and 0 seconds per second while stopped.

An invalid is not taken to mean that the value (or running state) of the tag is 0. If the source's value goes to invalid, it will be assumed that the last known value remains in effect and that value will continue to be used for the integration.

Values are added to the Totalizer tag at the end of each elapsed source time scale. This will not necessarily be "on the second" or "on the minute".

Do not attach a Logger tag to the Totalizer. At the end of each log interval, values are logged using the built-in connection to a Historian. You have the option of performing a reset of the Totalizer tag's value after each write to the log file.

You can cause a reset to be done at times other than the set logging interval, either by providing a reset button for the operator to use, or by configuring a tag to use an external reset. The current value is always logged before a reset is done.

Note: If the totalizer is drawn with a Set Analog Value tag or a Numeric Entry tag, two things will happen when the operator uses either widget: the totalizer will log its current value and it will reset to whatever value is set by the widget instead of 0.

It should be noted that a plot of the Totalizer tag on the Historical Data Viewer will show a constant value from one log interval to the next, with an instant change between values at each log interval.

The current value of the Totalizer tag is saved between VTScada restarts.

Note: Do no attempt to attach a Logger tag to a Totalizer tag. Totalizer tags have built-in logging.

Difference from the History Statistics tag, totalizer option:

The Totalizer tag calculates a time-weighted integral of the values collected. The totalizer option of the History Statistics tag provides a sum of the values within the time interval. Both have their uses.

Totalizer Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Totalizer Properties: Settings Tab
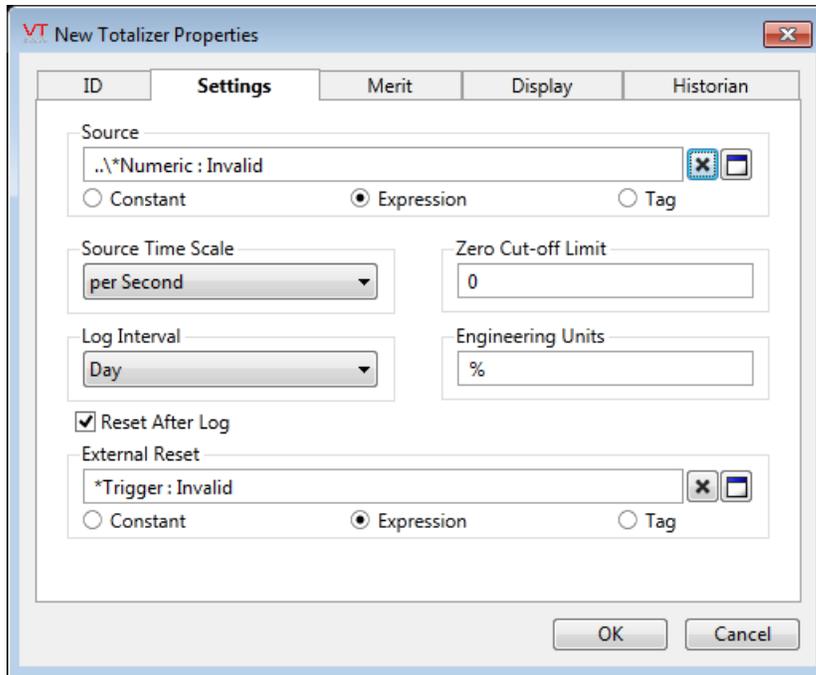
Source

The source can be any tag, calculation or even a constant. Whatever value is provided here will be the value that is totaled. In most cases, this will be an analog status or Analog Input tag, but there are no restrictions on the source. You might use a digital input to collect a total of equipment running time. If, for example, you specify a constant value of 1, the totalizer will record the total time that the application is running.

Source Time Scale

While there is no limitation on what can be used for the source, a Totalizer tag will commonly be used with equipment such as a flow meter. Flow meters may provide their values in units of flow per second, flow per minute or flow per hour. Selecting the correct time scale is of critical importance to collecting an accurate total.

Log Interval

The Totalizer tag will record its current total to a log file at the end of each interval specified. This interval goes by the clock, not by running time. If, for example, the log interval is set to "Day" the tag's value will be logged at midnight, not at the end of 24 hours of operation. If the log interval is set to "Hour" the tag's value will be logged each hour on the hour. Weekly means midnight on Sunday night and Monthly means midnight starting the first day of the month.

Reset After Log

You can select whether or not the totalizer should be reset to 0 after its value is logged.

Zero Cut-off Limit

To allow for a source that does not output an exact 0 when there should be no value to record, you can set a zero cut-off limit. This value is taken as a range, above and below zero, within which the totalizer takes 0 to be the source's value. Note that this is a range above and below zero, not a low value cut-off. Negative input values are possible from the source tag and are totaled as such.

Engineering Units

Text stating what the Totalizer tag is totaling (e.g. Gallons, Liters...).

External Reset

Specify a tag or calculation to force a reset of the totalizer. Note that the current value of the tag will always be logged

before a reset. The reset will occur when this input changes from a 0 or false value to a non-zero value. A change from invalid to non-zero does not cause a reset.

Totalizer Properties: Merit Tab

This tab contains the Questionable Data check box. Use this field to flag this tag's data in the event that you suspect the values it is reporting might not be accurate, or when this tag has initially been created and you wish to ensure that its data is marked for extra monitoring.
This tab also contains the Security Privilege field. Select an application security privilege from this drop down if you wish to limit the operation of this control to only those operators who have been granted the matching security privilege.
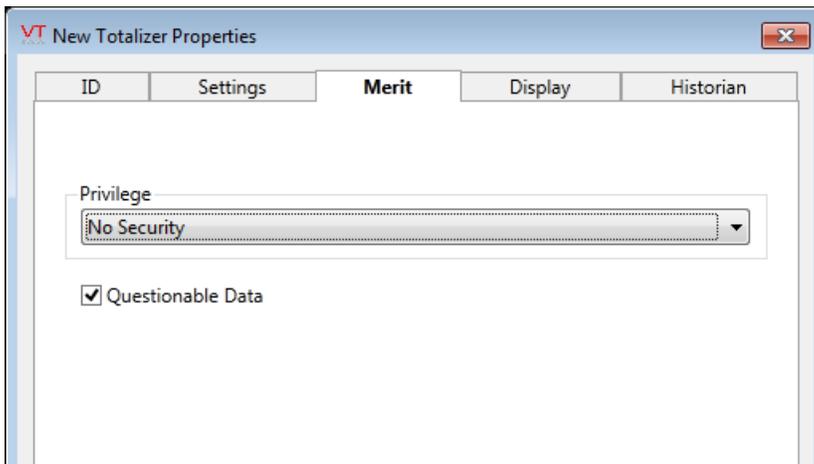
Totalizer Properties: Historian Tab
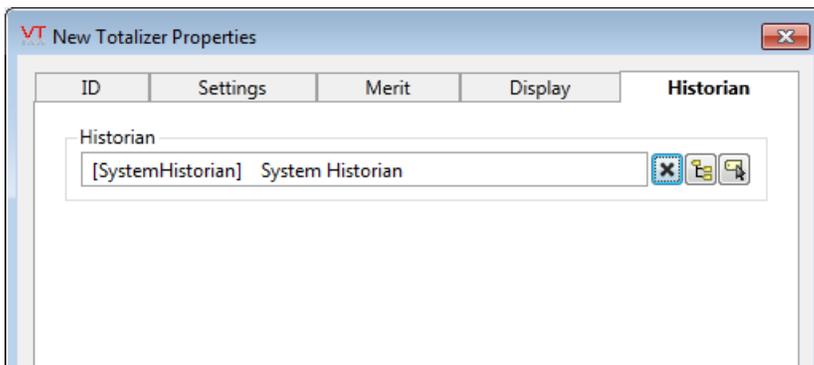
Historian

If a Historian tag is selected, this tag's run-time values will be saved for use in reports and the Historical Data Viewer. Historian configuration and advanced logging options are described in the discussion of the Historian Tags.

> Note: There are consequences if you change the selected Historian tag after you have begun collecting data. If you switch to a new Historian (perhaps for organizational or load sharing purposes), the data collected for this tag by the previous Historian will become inaccessible.

Totalizer Tag Properties: Display Tab

When this tag is represented on screen by widgets that are able to use a Style Settings tag, you can save development time by choosing the Style Settings tag that holds the correct display configuration for this tag instance.

The default configuration will search for the first parent tag of the Style Settings type. If there is none, (message reads, "--Missing--") it will use System Style, the default style tag that is automatically part of every new VTScada application.
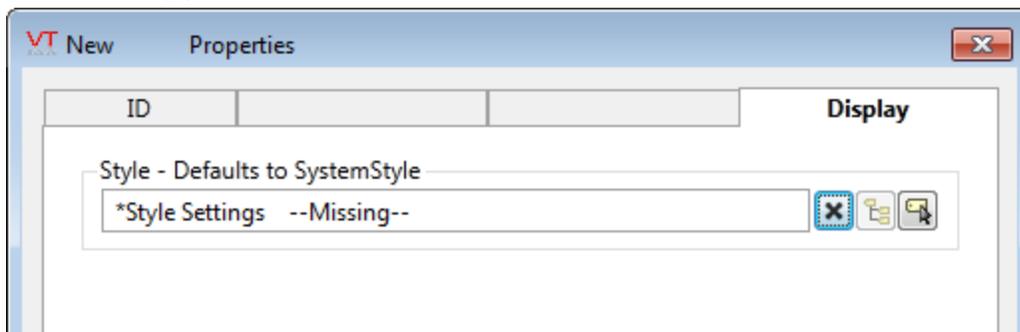
Related:

Style Settings Tags

Totalizer Tag Widgets

The following widgets are available to display information about your application's Totalizer tags:

Alarm Priority Box   Alarm Priority Icon

Animated Image Widget   Color Blink Widget

Color Box Widget	Color Fill Widget
Color Line Widget	LCD 5x7 Matrix Widget
Draw Text Widget	Elapsed Time Widget
Equipment / Status Color Indicator	Image Change Widget
Indicator Light Widget	Last Logged Value Widget
LCD 7 Segment Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Multi-Color Widget	Multi-Text Widget
Numeric Entry Widget	Numeric Value Widget
Plot Data Widget	Reset Button Widget
Reset Target Widget	Set Analog Value Widget
Two Color Bar Widget	Text Change Widget

Script Tags

Not counted towards your tag license limit.

Script tags monitor the value of another tag and pass that value to a module (VTScada script code). The module performs a calculation (whatever you program it to do) and sets the script tag's value to the result.

A module is a block of VTScada scripting code - a small program. You will need to have some familiarity with the VTScada scripting language in order to make use of a Script tag.

The module for a script tag must have two parameters, as follows:

```
(
    PointObj { object value of AI to monitor };
    ScriptObj { Script Object };
)
```

PointObj is the tag whose value is monitored and used by the Script tag. This matches the first field, "In Tag Scope" found in the Execute tab of the Script tag's configuration panel.

ScriptObj is used internally to link to your script tag. Whatever calculation is performed in your module must assign a value to ScriptOb-

j\value. This enables the module to pass its calculated value back to your Script tag.

Example:

The following module will perform a first order filter on the value of a given Analog Status tag. A first order filter takes 90% of the old value and 10% of the new, thus dampening out sudden changes. It is designed to run the calculation every two seconds rather than on change, therefore if the status tag's value changes from one value to another and stays there, the script tag's value will eventually match the new value.

```
{============================ Filter Module
==============================}
{ This module is used to do a First order Filter of an input point }
{ Current Value = .1 * new value + .9 * previous value }
{ The result is stored into the script point's value variable }
{=====================================================================-
==========}
(
    PointObj { object value of Analog Status to monitor (input) };
    ScriptObj { Script tag object (output) };
)
[ { Local variable }
  PreviousValue { Previous Value };
]
Filtering [
    { ..every two seconds... }
    If AbsTime(1, 2, 0);
    [
        { Insure there is a valid value first time through }
        PreviousValue = PickValid(PreviousValue, PointObj\Value);
        { Work out filtered value and assign that to the script tag }
        ScriptObj\Value = .1 * PointObj\Value + .9 * PreviousValue;
        { Save for next time }
        PreviousValue = ScriptObj\Value;
    ]
]
{ End of Filter Module }
```

This module would be stored in your application folder, perhaps as FILTER.SRC.

As with all VTScada modules, it must be declared in your application's AppRoot.SRC file. The declaration links a module file to a name (variable) that VTScada can use. The declaration must be added to the PLUGINS section of AppRoot.SRC and would look similar to the following:

```
[ (PLUGINS) {===== Modules added to other base system modules =====}
   Filter module FILTER.SRC;
]
```

Finally, when configuring your Script tag, you would put name of the Analog Status tag to monitor in the In Tag Scope field, and the declared variable, "Filter" in the Launched Module field.

Script Tag Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For

serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Script Tag Properties: Execute Tab

The Execute tab of the Script tag properties folder contains the properties required to identify the tag in which the script is to be executed, and the name of the script to be used.
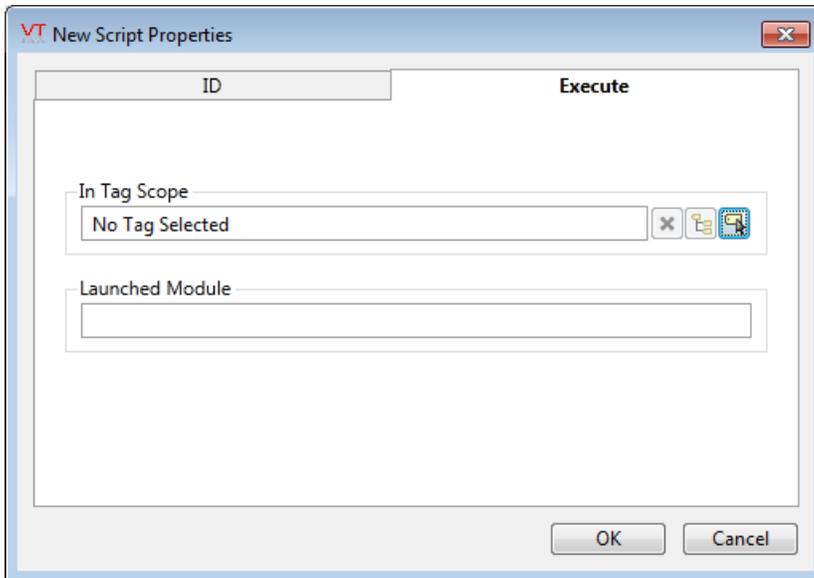
In Tag Scope

The In Tag Scope field enables you to specify the tag whose value will be used by your script module. Use the browser button to select the tag that will provide a value to be passed to the Launched Module

Launched Module

Provide the declared name of the module this Script tag will run. (See notes in the introduction of this topic). The name to enter here must be the variable that you declared in the application's AppRoot.SRC file, for the module that will calculate a value for this Script tag.

Script Tag Widgets

Script tags monitor the value of another tag and allow the execution of a script when the monitored tag's value changes. The widgets associated with Script tags enable you to display the value of the completed script. The following widgets are available to Script tags:

Alarm Priority Box   Alarm Priority Icon

Animated Image Widget	Bottom Bar Widget
Color Fill Widget	LCD 5x7 Matrix Widget
LCD Bar Widget	LCD 7 Segment Widget
Left Bar Widget	Meter 1 Widget
Meter 2 Widget	Meter 3 Widget
Multi-Color Widget	Multi-Text Widget
Numeric Value Widget	Plot Data Widget
Right Bar Widget	Top Bar Widget
Two Color Bar Widget	

Workstation Status Tags

Not counted towards your tag license limit.

Workstation status driver tags can be associated with one or more Analog Input tags to access data that the driver obtains from the Windows Management Instrumentation (an API in the Windows operating system that enables devices and systems in a network to be managed and controlled).

In order to access information about the workstation being monitored by a workstation status driver tag, follow these steps:

1. Create a workstation status driver tag. Set the Workstation Name drop-down list to the name of the workstation you wish to monitor. (You can create one workstation status driver tag for each workstation you wish to monitor.)

2. Create one Analog Input tag for each piece of data you wish to obtain from the workstation being monitored. (You can create multiple Analog Input tags and associate them with a single workstation status driver tag.)

3. Use the I/O Device field for each Analog Input tag to associate the analog input with the workstation status driver tag.

4. Set the Address field of each Analog Input tag to one of the available strings.

See: Workstation Status Driver Addressing.

Additional Configuration Requirements for Windows Vista

Windows Vista™ requires additional security configuration before you

will be able to access performance counter information (i.e. workstation status information). By default, an application will run using the same privileges as the person who started it. You can add the user account that the application runs as to the Performance Log Users group and then assign that group the right to "Log on as a Batch Job User".

For further information, refer to http://technet2.microsoft.com/windowsserver2008/en/library/8620ccc5-b054-48b4-b276-9f6c716954a71033.mspx?mfr=true

To add a user to the Performance Log Users group:

1. Click Start, click in the Start Search box, type compmgmt.msc, and then press ENTER.

2. Expand System Tools, expand Local Users and Groups, and click Groups.

3. In the list of groups, right-click Performance Log Users, and then click Add to Group.

4. On the General tab, click Add.

5. Type the name of the user you want to add, or click Advanced to search the directory for a user.

6. When you have finished adding users, click OK, and click OK again to close the Performance Log Users property page.

To assign the Log on as a batch job user right to the Performance Log Users group:

1. Click Start, click in the Start Search box, type secpol.msc, and then press ENTER. The Local Security Policy snap-in will open in Microsoft Management Console.

2. In the navigation pane, expand Local Policies and click User Rights Assignment.

3. In the console pane, right-click Log on as a batch job and then click Properties.

4. In the Properties page, click Add User or Group.

5. In the Select Users or Groups dialog box, click Object Types. Select Groups, and then click OK.

6. Type Performance Log Users in the Select Users or Groups dialog box, and then click OK.

7. Click OK again to close the property page.

Workstation status driver tags have been configured in such a way that they do not conform to the same server/backup server fail over process like other VTScada drivers (see "Remote Applications"). In essence, the PC being monitored by a workstation status driver tag is its own server. If the PC being monitored drops off the network, the workstation status driver tag will be unable to report data about it to the application.

Related Information:

Workstation Status Driver I/O Addressing

Workstation Status Driver Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Workstation Status Driver Properties: Workstation Tab

This tab is used to specify the workstation for which the tag should report data.

Workstation Name

The Workstation Name drop-down list enables you to select the name of the workstation that you wish to monitor (you may alternatively enter a Net BIOS name in this field). The Workstation Name field defaults to the local machine name when new workstation status driver tags are initially created.

> Note: Please refer to Workstation Status Driver Tag Addressing for instructions on the correct configuration of a workstation status tag and its associated Analog Input tags.

Workstation Status Driver Tag Widgets

The following widgets are available to display information about your application's workstation status driver tags:

...Alarm Priority Box

...Alarm Priority Icon

...Comm Indicator Widget

...Comm Line Widget

...Equipment / Status Color Indicator

...Gradient Color Change Widget

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...Numeric Value Widget

...Multi-Color Widget

...Multi-Text Widget

...Plot Data Widget

Workstation Status Driver I/O Addressing

A workstation status driver tag monitors a workstation and enables associated Analog Input tags to report and display data about the workstation.

In order to access information about the workstation being monitored by a workstation status driver tag, follow these steps:

1. Create a workstation status driver tag. Set the Workstation Name drop-down list to the name of the workstation you wish to monitor. (You can create one workstation status driver tag for each workstation you wish to monitor.)
2. Create one Analog Input tag for each piece of data you wish to obtain from the workstation being monitored. (You can create multiple Analog Input tags and associate them with a single workstation status driver tag.)
3. Use the I/O Device field for each Analog Input tag to associate the analog input with the workstation status driver tag.
4. Set the Address field of each Analog Input tag to one of the available strings.

Address Field Value	Information Returned from Workstation
AverageCPU	Running average of total system CPU
BatteryLevel	The percentage of the battery level remaining
BatteryLifetime	An estimation of the remaining battery life, in seconds
PowerStatus	0 == on battery 1 == on A/C 2 == on back-up power
BatteryState:High	1 if the battery's power state is high
BatteryState:Low	1 if the battery's power state is low
BatteryState:Critical	1 if the battery's power state is critical
BatteryState:Charging	1 if the battery is charging
BatteryState:NoBattery	1 if no battery is found
Expression:Memory()	Calls the Memory function, returning the current amount of memory acquired from the operating system heap on the named workstation.
FreeDiskSpace	Free disk spaces (bytes) for a logical disk that is addressable with a drive letter. (*) See note.
GDIObjects	GDI objects used by VTS
Handles	Handles in use by VTS
IOBPS	I/O bytes per second, including all disk reads/writes, as well as network traffic
Memory	VTScada memory usage in bytes, as seen by Windows. Does not decrease as memory is released.

SystemCPU	System CPU – all processes
Threads	Threads being used by VTS
VirtualMemory	Virtual memory used by VTScada (in bytes) as seen by Windows. Does not decrease as memory is released.
VTSCPU	CPU used by VTScada only

Note: (*) You must append the letter for the drive to be monitored to FreeDiskSpace (e.g. FreeDiskSpaceC for drive C, or FreeDiskSpaceD for drive D.

Station and Site Tags

Tags in this category create multi-I/O structures, designed for specific brands of hardware.

Related Information:

...MPE Duplexer Station Tags. Define a complete MPE Duplexer station.

...MPE SC Series Station Tags. Define a complete MPE SC Series station.

...MultiSmart Station Tags. Define a complete MultiSmart station.

...ScadaAce Site Tags   Add sites in a ScadaAce-based application

MPE Duplexer Station Tags

These create several child tags that do count towards you license limit, but as containers, are not themselves counted.
"MPE" stands for Motor Protection Equipment. A single MPE Duplexer Station tag includes all of the I/O required to monitor and operate a MPE Duplexer station.

MPE Duplexer Station Properties: ID Tab
The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field.

You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

MPE Duplexer Station Properties: Settings Tab
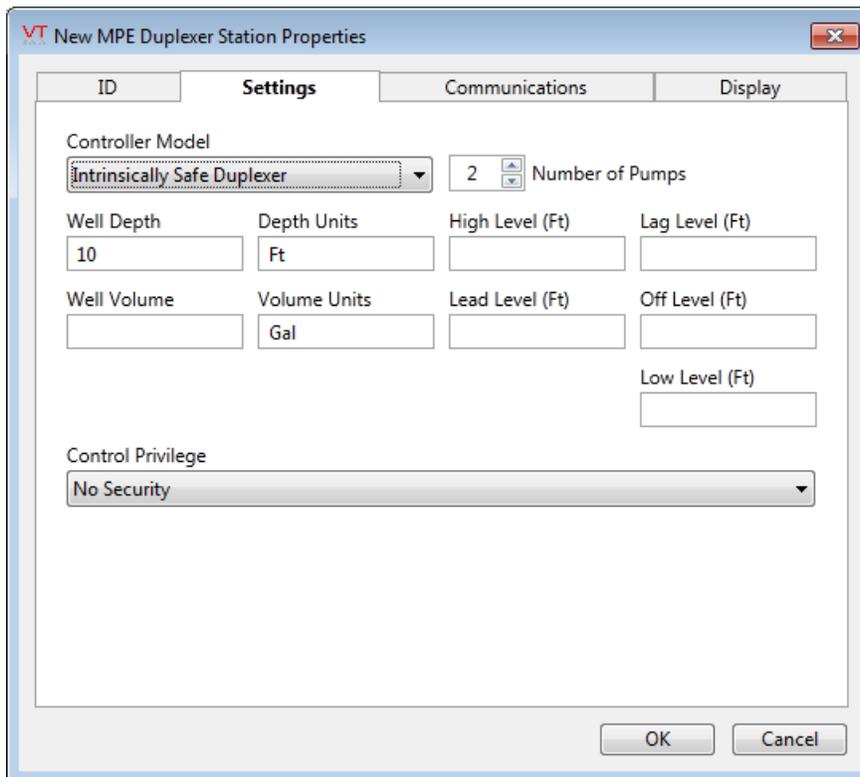
Controller Model

The options are, Duplexer or Intrinsically Safe Duplexer.
Defaults to Intrinsically Safe.

Number of Pumps

Set the number of pumps in the station, thereby controlling how many associated child tags will be created. Ranges from one to two.

Well Depth

Provide the depth of the well, as measured in the units specified in the next field. Setpoints must be with the range of the well depth provided.

Depth Units

Provide the system of units to be used for the depth of the well. This value is used only as a label on the display.

High Level (Ft)
Provide a number

Lag Level (Ft)
Provide a number

Well Volume
Provide the volume capacity of the well, as measured in the volume units specified in the next field.

Volume Units

Provide the system of units to be used for measuring the volume of fluid in the well. This value is used only as a label on the display.

Lead Level (Ft)

Provide a number

Off Level (Ft)

Provide a number

Low Level (Ft)

Provide a number. Used only if the Intrinsically Safe model has been selected.

Privilege

If you wish to restrict access to the controls of this station, select the application privilege to be associated with it. Only the operators who have been assigned a security rule that includes this privilege will be able to control the station.

MPE Duplexer Station Properties: Communications Tab
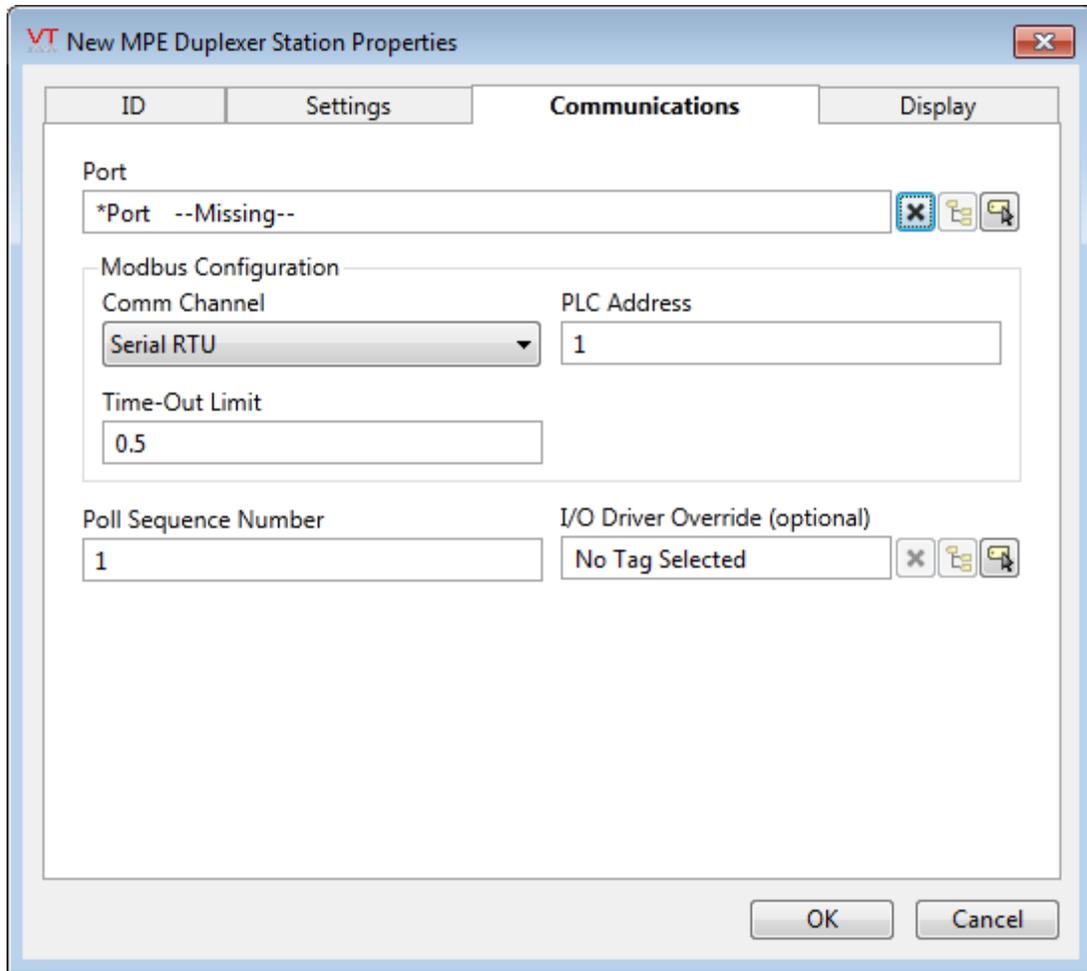
Port

Select the communications port to use for this station. By default, the nearest parent will be selected.

Modbus Configuration

The driver created for this station will be based on the Modbus Compatible Device . Use the fields in this section to configure the details of the Modbus protocol that will be used.

Station Number

Poll Sequence if this tag is using a Polling Driver. Also used

for sorting the data in a Pump Activity or Derived Flow report.

I/O Driver Override (optional)

Provides a way for you to select an existing driver tag to be used instead of the driver built-into the station.

Related Information:
Modbus Compatible Device Driver Tags

MPE Duplexer Station Properties: DisplayTab
The Display tab is used to define the placement (Latitude and Longitude parameters) of the MPE Duplexer station. Decimal values should be used rather than degrees, minutes and seconds.

In most cases, it is easier to set the location using the map interface of the Site Details page than to enter the latitude and longitude values here.

Custom Details Page

A pre-built page is provided for your MPE Duplexer station. This includes a display of all I/O values, alarms and settings that can be used.
If you prefer to build your own parametrized page, to be opened when an operator clicks on this MPE's site draw widget, then select that page in the drop-down list provided in this field.
Parameter: CustomDetailsPage

Custom Map Icon

This station will be displayed on all site maps in the application using a pin. You can create and use any icon you like in place of the standard pin. (Instructions are provided elsewhere in this document) If you have done so, you may use this field to select the icon that will represent this station on maps.
Parameter: CustomMapIconParm

Exclude from the site list

When checked, this tag will not be included on a Sites page.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).

Default zoom level.

MPE Duplexer Station Tag Widgets

The following widget are available:

Alarm Priority Box	Alarm Priority Icon
Alarm List	Site Icon Widget
Site Summary Widget	Site Alarm List
Site Details	Pump Control
Well Details	Well Flow
Well Plot	Well Setpoints
Controller Status	Station Summary
Alarm List	Station All Inputs
Station All Outputs	Station Comms
Station Faults	Station Stats
Pump List	Tag List

Analog Control List   Analog Input List

Digital Input List   Digital Output List

Related Information:
Sites & Maps
Site Details Page
Site Map
Create Custom Map Icons

MPE Duplexer Child Tags

A single MPE Duplexer Station tag includes all of the I/O required to monitor and operate a MPE Duplexer pumping station. The I/O tags, alarms and more are implemented as child tags of the MPE Duplexer Station.

> Note: It is recommended that you do not create instances of these tags, independently of a MPE Duplexer station.

MPE Duplexer Child Tag Widgets
These widgets are components of the overall MPE Duplexer Station displays. While they can be drawn independently, it is recommended that you use the MPE Duplexer Station widgets in order to create these components in the correct context.

Related Information:
MPE Duplexer Station Tags

MPE SC Series Station Tags

These create several child tags that do count towards you license limit, but as containers, are not themselves counted.
MPE stands for Motor Protection Equipment. This tag will support either of the SC1000 or the SC2000 controllers.

MPE SC Series Station Properties: ID Tab
The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

MPE SC Series Station Properties: Settings Tab

Controller Model

Specify whether this tag is to be used with the MPE SC1000 or the SC2000 controller.

Number of Pumps

Select the number of pumps that are installed at this station. Possible values range from one to four.

Well Depth

Provide the depth of the well, as measured in the units specified in the next field.

Depth Units

Provide the system of units to be used for the depth of the well. This value is used only as a label on the display.

Flow Units

Provide the system of units to be used for flow. This is required only if flow calculations are supported by the controller.

Well Volume

Provide the volume capacity of the well, as measured in the volume units specified in the next field.

Volume Units

Provide the system of units to be used for measuring the volume of fluid in the well. This value is used only as a label on the display.

Total Daily Flow Units

Provide the system of units to be used for total flow. This is required only if flow calculations are supported by the controller.

Depth Probe Present

Check to box to indicate that your well has a depth probe.

Station Backup Generator Present

Check to box to indicate that your station has a backup generator.

Flow Calculations Supported

Check to box to indicate that your controller supports flow calculations.

Pump VDF Control Supported

Check to box to indicate that your controller provides VDF control.

Privilege

If you wish to restrict access to the controls of this station, select the application privilege to be associated with it. Only the operators who have been assigned a security rule that includes this privilege will be able to control the station.

MPE SC Series Station Properties: Communications Tab
In this tab, you define the parameters for the Communications Port and the Modbus Driver that will be created automatically as part of this station.

Port

> Select the communications port to use. By default, the nearest parent will be selected.

Modbus Configuration

The driver created for this station will be based on the Modbus Compatible Device . Use the fields in this section to configure the details of the Modbus protocol that will be used. For detail, see: Modbus Compatible Device Tags.

Station Number

Poll Sequence if this tag is using a PollDriver. Also used for sorting the data in a Pump Activity or Derived Flow report.

I/O Driver Override (optional)

Provides a way for you to select an existing driver tag to be used instead of the driver built-into the station.

MPE SC Series Station Properties: Auxiliary DI Tab
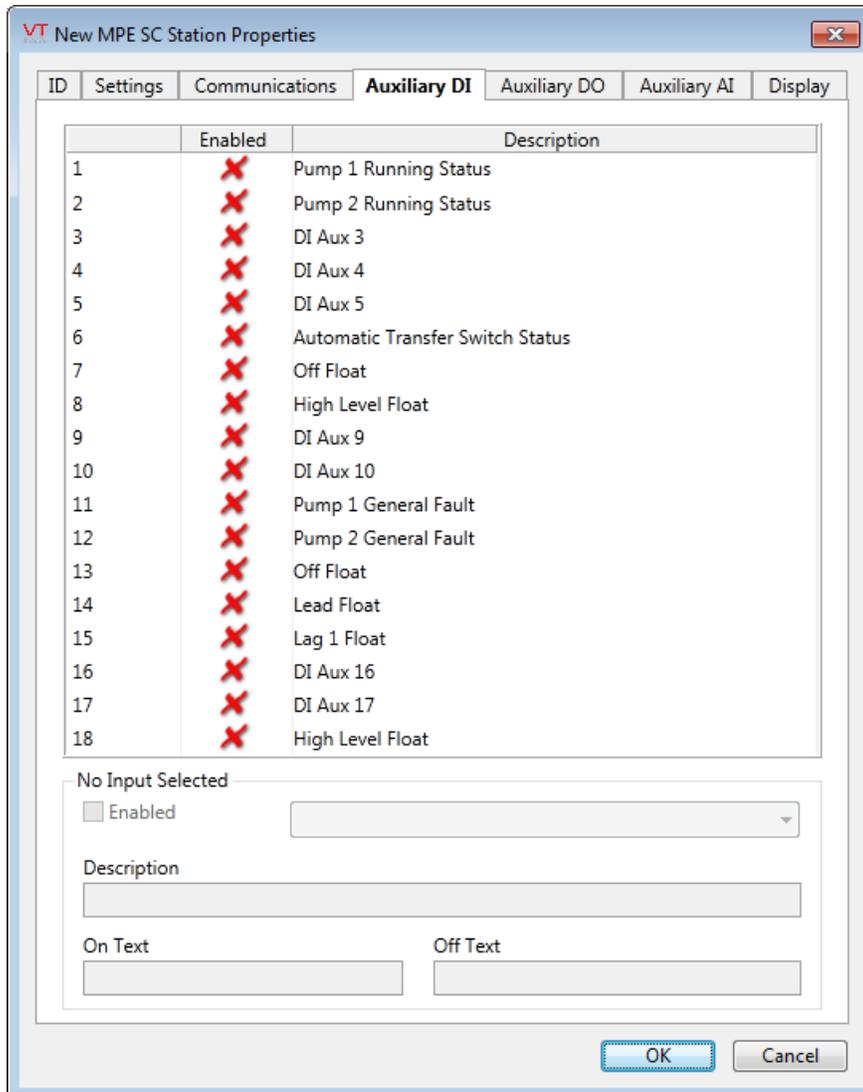
Provides an Enabled / Disabled option for each of the auxiliary input addresses of the station. The actual value being monitored by each input, N, can be configured using the drop-down list in the "Input N" section of the panel.

MPE SC Series Station Properties: Auxiliary DO Tab

You may define up to six auxiliary digital outputs for an SC2000 controller and up to five auxiliary digital inputs for the SC1000.
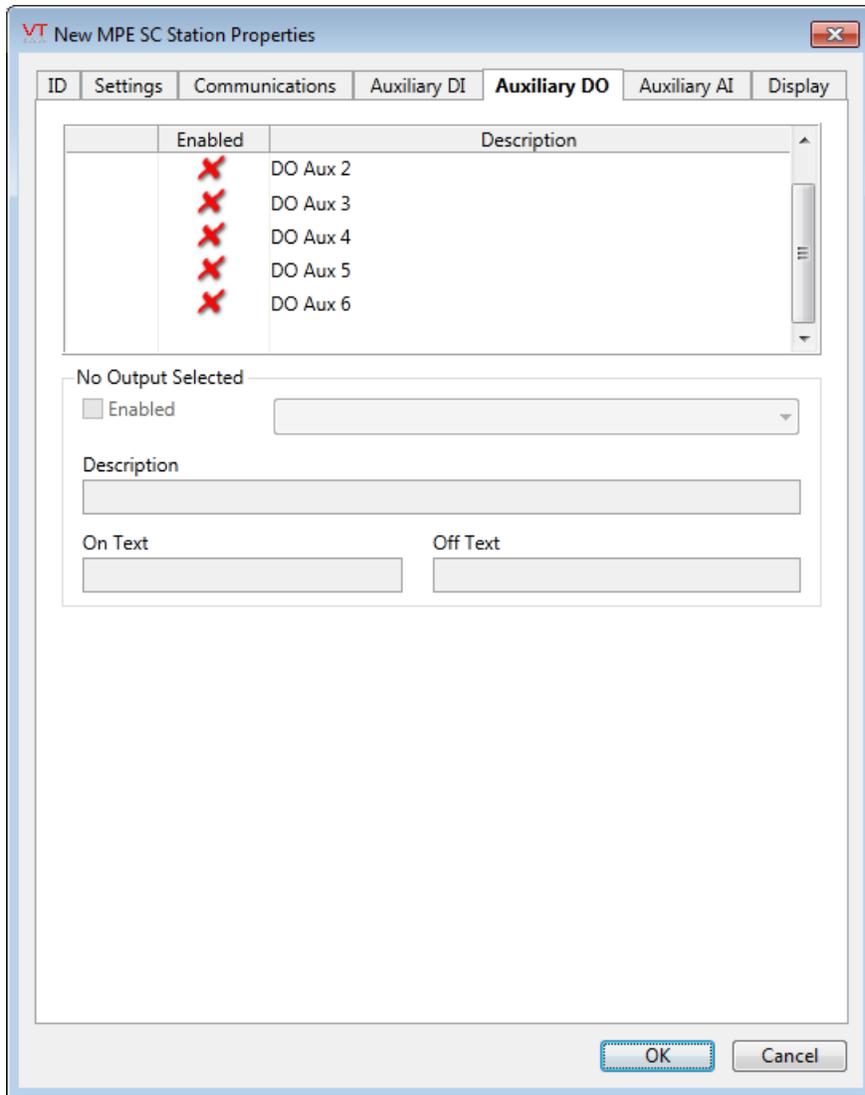
MPE SC Series Station Properties: Auxiliary AI Tab

You may define up to four auxiliary analog inputs for the SC2000 controller. The SC1000 does not support auxiliary analog inputs.
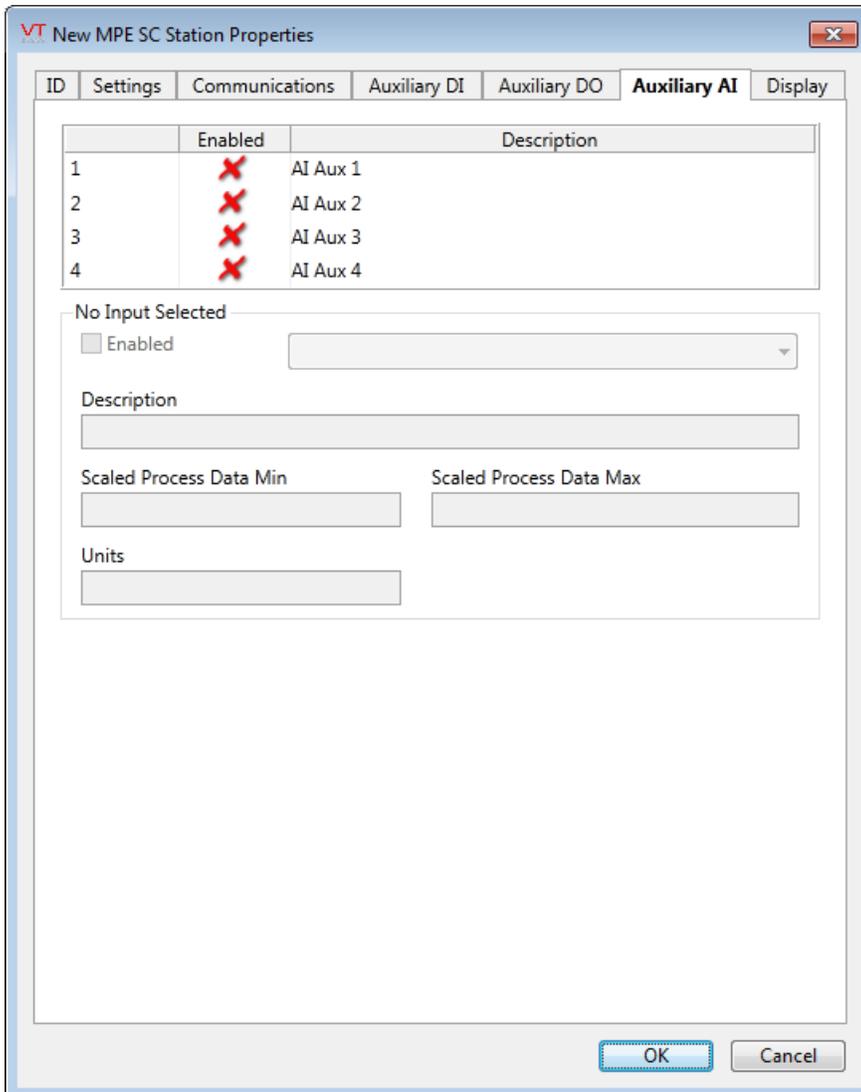
MPE SC Series Station Properties: Display Tab

The Display tab is used to define the placement (Latitude and Longitude parameters) of the MPE SC Series station. Decimal values should be used rather than degrees, minutes and seconds.
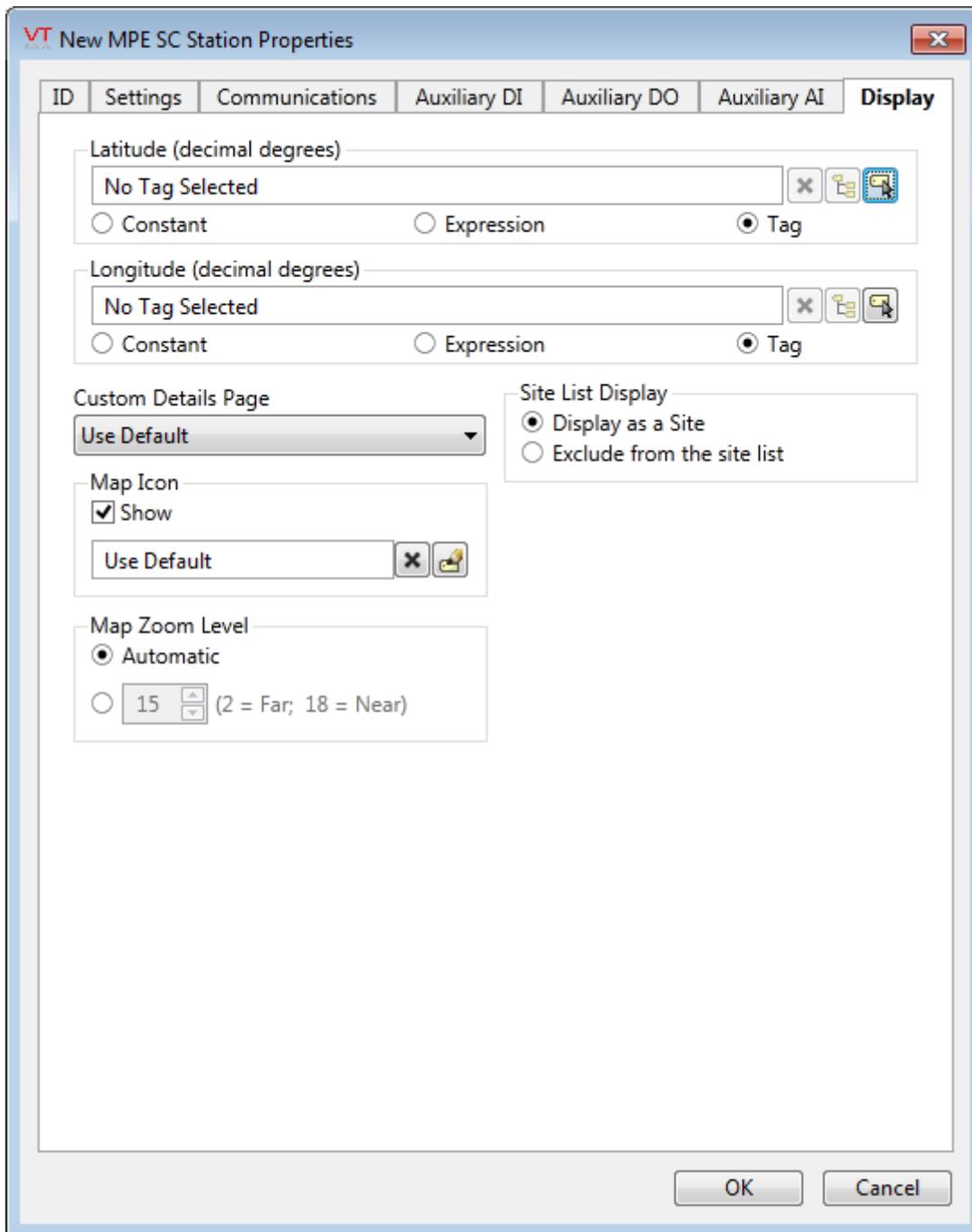
In most cases, it is easier to set the location using the map interface of the Site Details page than to enter the latitude and longitude values here.

Custom Details Page

A pre-built page is provided for your MPE SC Series station.

This includes a display of all I/O values, alarms and settings that can be used.

If you prefer to build your own parametrized page, to be opened when an operator clicks on this MPE's site draw widget, then select that page in the drop-down list provided in this field.

Parameter: CustomDetailsPage

Custom Map Icon

This station will be displayed on all site maps in the application using a pin. You can create and use any icon you like in place of the standard pin. (Instructions are provided elsewhere in this document) If you have done so, you may use this field to select the icon that will represent this station on maps.

Parameter: CustomMapIconParm

Exclude from the site list

When checked, this tag will not be included on a Sites page.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).

Default zoom level.

MPE SC Series Station Tag Widgets

The following widgets are available to display information about your application's MPE SC Series Station tags:

Alarm Priority Box	Alarm Priority Icon
Alarm List	Site Icon Widget
Site Summary Widget	Site Alarm List
Site Details	Pump Control
Well Details	Well Flow
Well Plot	Well Setpoints
Controller Status	Pump Alternation Control
Station Summary	Station Power
Alarm List	Station All Inputs
Station All Outputs	Station Comms

Station Faults	Station Stats
Pump List	Tag List
Analog Control List	Analog Input List
Digital Input List	Digital Output List

Related Information:

...Sites & Maps

...Site Details Page

...Site Alarm List

...Site Map

...Create Custom Map Icons

MPE SC Child Tags

A single MPE SC Station tag includes all of the I/O required to monitor and operate a MPE SC Series pumping station. The I/O tags, alarms and more are implemented as child tags of the MPE SC Station.
(See: Parent-Child Tag Structures)

> Note: It is recommended that you do not create instances of these independently of a MPE SC station.

MPE SC Child Tag Widgets
These widgets are components of the overall MPE SC Station displays. While they can be drawn independently, it is recommended that you use the MPE SC Station widgets in order to create these components in the correct context.

Related Information:
MPE SC Series Station Tags

MultiSmart Station Tags

These create several child tags that do count towards you license limit, but as containers, are not themselves counted.

A single MultiSmart Station tag includes all of the I/O required to monitor and operate a MultiSmart pumping station. Configuration is done in part through the standard set of VTScada tag configuration panels and in part through a MultiTrode-supplied XML file that describes your equipment. File names to look for are as follows:

- DNP3: dnpslave.xml
- Modbus: mbs.xml

MultiSmart Station Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

MultiSmart Station Properties: Settings Tab

Number of Pumps

Lift stations may have from one to six pumps.

Well Depth

Provide the depth of the well as measured in the Units value specified (meters, feet or inches). This value is used for the visual representation of the well and as a limit on the possible setpoint values.

Well Volume

Provide the volume capacity of the well, measured in the Units value specified (US gallons or Liters). This value is used by related reports.

Units

Select the system of measurement to be used. Choices are Metric or Imperial.

Level Units

Select how units for the well level and related tags are to be displayed. Choices are "m", "ft", "in", "%", or any custom unit that you care to type.

XML Slave Index

Select which slave from the XML file should be used by VTScada when generating the tag list. The is most often 1 (using the first and only slave in the file). The value represents a 1-based index. The spinbox will be limited to the number of slaves defined in the XML file.

Negative Offset

An offset to the level measurement, used to account for the distance from the bottom of the probe to the bottom of the well. Since the zero level in VTScada is relative to the bottom of the probe, failing to add the value of this offset can result in false alarms and incorrectly displayed values.

XML Configuration File and Import XML File

Click the Import XML File button to locate and import the configuration file for this station. A copy is made of the file that you import – no link is maintained to the original file. If the original is edited, you will need to import again in order to bring those changes into your application. File names to look for are as follows:

- DNP3: dnpslave.xml
- Modbus: mbs.xml

The selection list shows the files that have previously been imported.

Privilege

If you wish to restrict access to the controls of this station, select the application privilege to be associated with it. Only the operators who have been assigned a security rule that includes this privilege will be able to control the station.

MultiSmart Station Properties: Communications Tab

Port

Select the TCP/IP or Serial Port tag that is to be used for communication with the equipment. If the station is being created as a child tag of a port, the ancestor port tag will be selected automatically.

Protocol

Selection determines what type of communications driver will be created for use in this station. Choices are DNP3 or Modbus

Protocol Configuration

Configuration options vary according to the selected protocol type. For detail, see: DNP3 Driver Tags or Modbus Compatible Device Tags.

Station Number

Poll Sequence if this tag is using a Polling Driver. Also used for sorting the data in a Pump Activity or Derived Flow report.

I/O Driver Override (optional)

Provides a way for you to specify a specific, existing driver tag to be used instead of the driver built-into the station. In most cases this need not be selected.

MultiSmart Station Properties: DisplayTab

The Location tab is used to define the placement (Latitude and Longitude parameters) of the MultiSmart station. Decimal values should be used rather than degrees, minutes and seconds.

In most cases, it is easier to set the location using the map interface of the Site Details page than to enter the latitude and longitude values here.

Custom Details Page

A pre-built page is provided for your MultiSmart station. This includes a display of all I/O values, alarms and settings that can be used.
If you prefer to build your own parametrized page, to be opened when an operator clicks on this MultiSmart's site draw widget, then select that page in the drop-down list provided in this field.
Parameter: CustomDetailsPage

Custom Map Icon

This station will be displayed on all site maps in the application using a pin. You can create and use any icon you like in place of the standard pin. (Instructions are provided elsewhere in this document) If you have done so, you may use this field to select the icon that will represent this station on maps. Parameter: CustomMapIconParm

Exclude from the site list

When checked, this tag will not be included on a Sites page.

Map Zoom Level

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).

Default zoom level.

MultiSmart Station Tag Widgets

The following widgets are available to display information about your application's MultiSmart Station tags. In most cases, you need only use the Station Icon or the Station Summary. The remaining methods are components of the station page that will be created for you. Both the Station Icon the Station Summary link to this page.

If your application includes more than one station tag, you may prefer to draw a Station List, which is an automatically created display of Station Summary widgets, one for each station.

Alarm Priority Box	Alarm Priority Icon
Alarm List	Site Icon Widget
Site Summary Widget	Site Alarm List
Site Details	Pump Control
Well Details	Well Flow
Well Plot	All Well Setpoints
Well Setpoints	Controller Profile
Controller Maintenance	Alarm List
Custom Alarms	Station Summary
Station Power	Pump List
Station All Inputs	Station All Outputs
Station Comms	Station Faults
Station Stats	Tag List
Analog Control List	Analog Input List
Digital Input List	Digital Output List

Related Information:

...Sites & Maps

...Site Details Page

...Site Map

...Create Custom Map Icons

MultiSmart Station Child Tags

A single MultiSmart Station tag includes all of the I/O required to monitor and operate a MultiSmart pumping station. The I/O tags, alarms and more are implemented as child tags of the MultiSmart Station.

> Note: It is recommended that you do not create instances of these tags, independently of a MultiSmart station.

MultiSmart Station Child Tag Widgets

These widgets are components of the overall MultiSmart displays. While they can be drawn independently, it is recommended that you use the MultiSmart widgets in order to create these components in the correct context.

Related Information:
MultiSmart Station Tags

ScadaAce Site Tags

These create several child tags that do count towards you license limit, but as containers, are not themselves counted.

Available only in applications that are built on the ScadaAce layer.
A ScadaAce Site tag should be created using either the Add Site button or Manage Sites button in the Sites page of a ScadaAce application.
Note that, the first time you start a ScadaAce application, you will be prompted to create a gateway.

ScadaAce Site Properties: ID tab
When created using the Add button in a Motorola IP Gateway tag, the ID tab will not look like other ID tabs. The row of tabs, found across the top of most tag configuration panels, will be shown only after the ID tab fields have been filled in and you click OK.
Note that, clicking the OK button will cause two actions: The site tag will be created, and the configuration panel for this tag will be re-opened,

this time showing all the tabs. After that point, clicking the Cancel button will discard configuration changes, but will not abort the tag creation process.

Note that upon clicking OK, a series of other configuration tabs will be enabled. Of those, the Analog Inputs tab will be opened first.
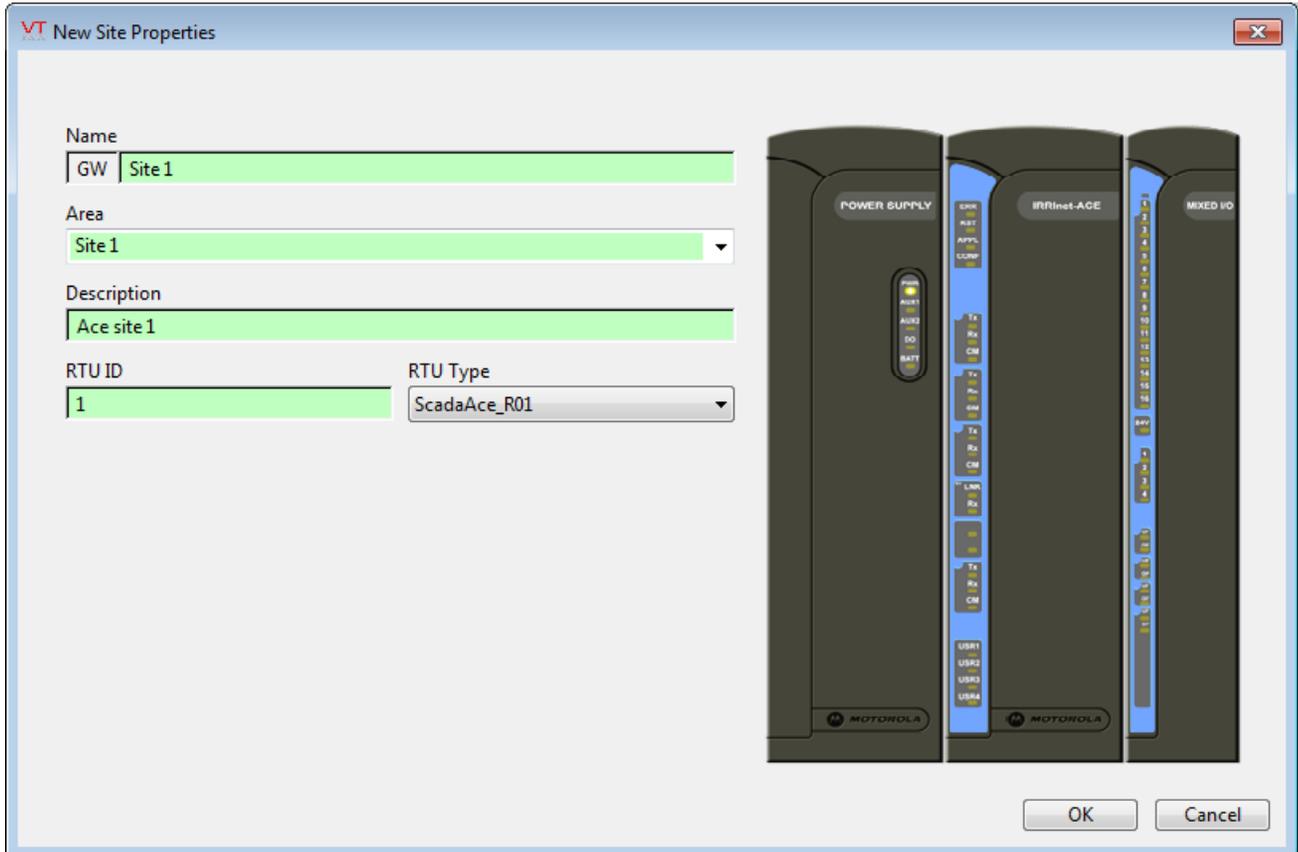
Name

Add a name, appropriate for the site being created.

Area

After pressing Tab or Enter to finish entering the name, the Area field will automatically configure itself to match the name that you provided. This is done so that, when viewing alarms on the Alarm Page, any alarms from different sites that happen to have the same description (a very likely scenario) can be told

apart by looking at the Area column. You are free to change the Area field if you wish.

Description

A description should be provided for every site.

RTU ID and RTU Type

The RTU ID value should match the identifier for the device at this site.

ScadaAce Site Properties: Settings tab

The Settings tab repeats two parameters from the ID tab: RTU ID and RTU Type. In most cases, you will have configured these values before creating the tag, but they may be created, or changed in this tab.
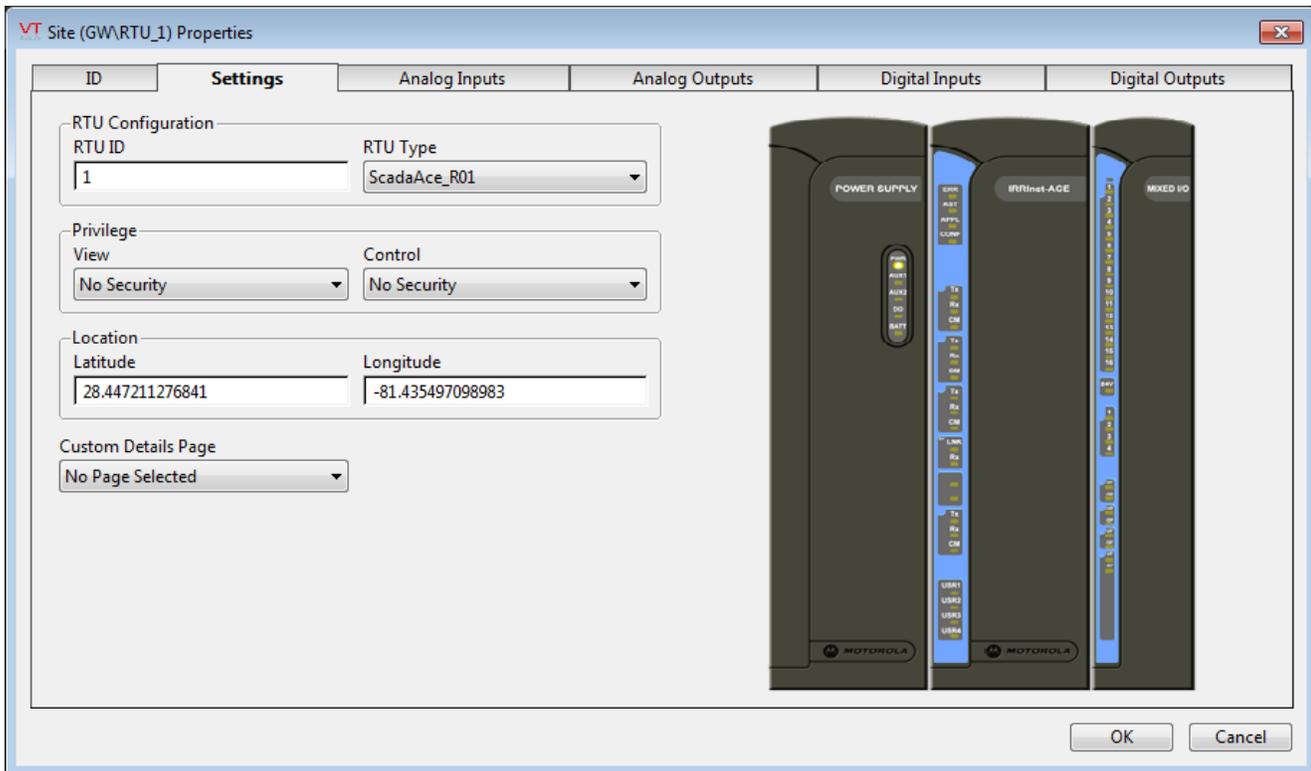
Two application privileges may be applied to a ScadaAce site: Operators with the View privilege may see the details of this site in a site list. Oper-

ators with the Control privilege may write values with this site's output tags.

You must create your own application privileges to use.

The location fields, Latitude and Longitude are most easily filled in by placing this site on a map. This can be done when creating the site, if you use the Add Site button in the Sites page. Or, you may locate the site after you have saved the configuration. Close all development dialogs and click on the site - a Site Details map will open, including a map. There, you may click the Update Location button to place or move the location pin.

Alternatively, you are free to type in the latitude and longitude coordinates for the site.

Selecting a Custom Details page is an advanced operation, requiring that developer create a customized page for this site. If so, it may be selected using this menu item.

Related Information:

Application Privileges.

ScadaAce Site Properties: Analog Inputs tab

Up to 25 Analog Input points may be configured for each site. Matching child tags of type, Analog Status, will be created for each row that you configure using the spreadsheet format. The format allows for rapid configuration of tags.

Unscaled values need not be provided since these values are known for the ScadaAce RTU. I/O addresses for each input are predefined by the ScadaAce system.

The I/O address of each tag is configured for you, following the format, B2:n-1:0 where n matches the number in the tag's name.

Click on the label <add>, in the description column to begin configuring each input. Note that you may tab between fields, and use the Enter key to finish an input and move on to the next row.

Related Information:

Analog Status Tags - Provides a full description of the configuration fields in an analog input.
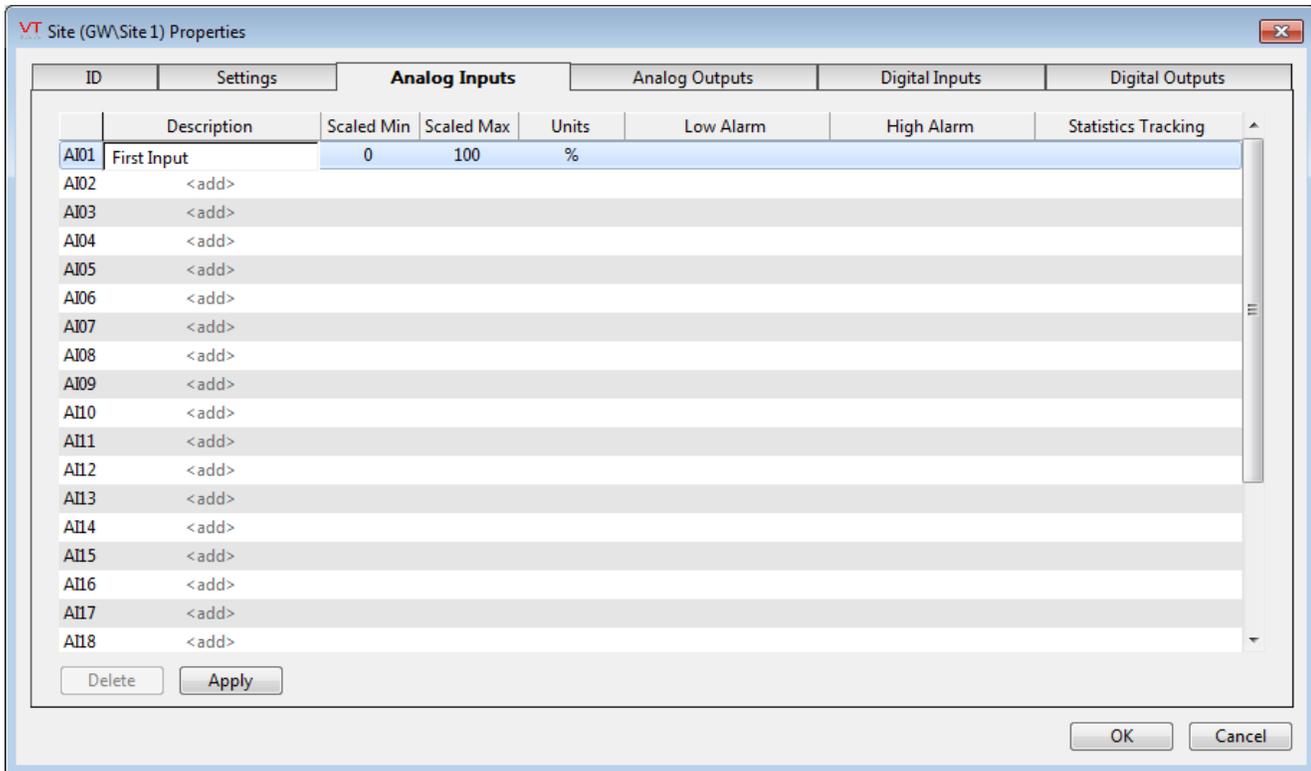
Description

Since the names are predefined, you must provide a description for each tag in order for operators to know what the value represents.

Scaled Min

The minimum value for this input, in terms of the engineering units being measured.

Scaled Max

The minimum value for this input, in terms of the engineering units being measured.

Units

Engineering units that the input represents.

Low Alarm

Upon first clicking in the Low Alarm field, the space will resolve into two inputs: The value, below which a low level alarm should be activated, and the priority of that alarm, presented as a drop-down list.
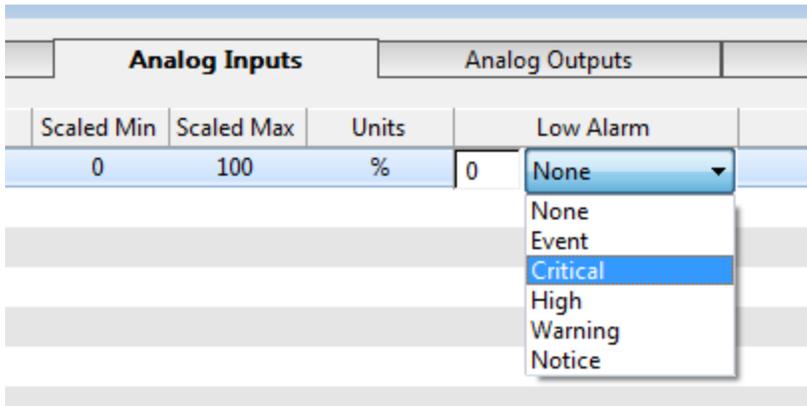
High Alarm

Configured in the same way as the Low Alarm. When the input's value goes above this setpoint, an alarm of the specified priority will be activated.

Statistics Tracking

Check in order to enable tracking of Minimum, Maximum and Average values of this input, over time. Two History Statistics tags will be created for each option you choose. Statistics over the past 24 hours and statistics for the time period, "yesterday".
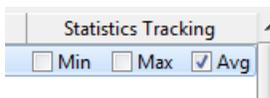
The Delete button will clear an I/O tag's configuration, effectively removing the matching child tag. It does not destroy the matching tag number, which may be used again later.

ScadaAce Site Properties: Analog Outputs tab

Up to four analog output tags (created using the Analog Control type) can be configured. Operators will be able to use these tags to write

values to the RTU.

The I/O address of each tag is configured for you, following the format, 6:n-1:0 where n matches the number in the tag's name.

Click on the label <add>, in the description column to begin configuring each output. Note that you may tab between fields, and use the Enter key to finish an input and move on to the next row.

Related Information:

Analog Control Tags – Provides a full description of the configuration fields in an analog output.

Description

Since the names are predefined, you must provide a description for each tag in order for operators to know what the value represents.

Scaled Min

The minimum value for this input, in human terms.

Scaled Max

The minimum value for this input, in human terms.

Units

Engineering units that the input represents.

ScadaAce Site Properties: Digital Inputs tab

Up to 96 digital input tags (created using the Digital Status type) can be configured. The are used to monitor whether points on the RTU are in state 0 or state 1.

The I/O address of each tag is configured for you, following the format, B1:n-1:0 where n matches the number in the tag's name.

Click on the label <add>, in the description column to begin configuring each input. Note that you may tab between fields, and use the Enter key to finish an input and move on to the next row.

Related Information:

Digital Status Tags – Provides a full description of the configuration fields in a digital input.

Description

Since the names are predefined, you must provide a description for each tag in order for operators to know what the value represents.

OFF text / ON text

These columns match input states zero and one, respectively. You may change the text as needed if state zero is defined as "ON" and state one is defined as "OFF" at your operation.

Alarm

Click in the alarm field to begin configuration. Use the first drop-down list to configure the state or state-transition that will activate an alarm. Using the second column, select the priority of this alarm.

Alarm Delay

Configure a time period, measured in seconds, for which the tag must remain in its activation state before an alarm is considered to have occurred.

Statistics

Tracking Check in order to enable tracking of On Time, or On Counts of this input, over time. Two History Statistics tags will be created for each option you choose. Statistics over the past 24 hours and statistics for the time period, "yesterday".
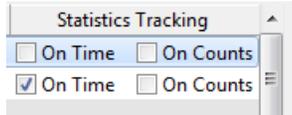
ScadaAce Site Properties: Digital Outputs tab

Up to 48 digital output tags (created using the Digital Control type) can be configured. These allow operators to issue control actions to the equipment.

The I/O address of each tag is configured for you, following the format, 5:n-1:0 where n matches the number in the tag's name.

Click on the label <add>, in the description column to begin configuring each output.

Related Information:

Digital Control Tags – Provides a full description of the configuration fields in a digital output.

Description:

Since the names are predefined, you must provide a description for each tag in order for operators to know what the value represents.

OFF text / ON text.

These columns match input states zero and one, respectively. You may change the text as needed if state zero is defined as "ON" and state one is defined as "OFF" at your operation.

ScadaAce Site Properties: Drawing Methods

Three widgets are available for the ScadaAce tag, Site Icon, Site Details, and Manage I/O.
In practice, it is not expected that you will need to use any of these. The Sites page, built into every ScadaAce-based application, will automatically add each site you configure.

Configuration Tags

Tags in this category relate to the appearance or structure of your application. This list includes Context tags, Font tags and the Realm Display Setup tag.

Related Information:

...Comm Link Sequencer Tags – Sequence requests for the same communication channel across multiple ports.

...Context Tags – Create a generic and customizable template that you can use to define any natural grouping in your application

...IP Network Listener Tags – Allows certain driver tags to listen for IP messages.

...Font Tags – Specify font and text style characteristics for your labels.

...Realm Display Setup Tags – Defines the level of display control, granted to operators connecting to an application from a VTScada Internet Client.

Comm Link Sequencer Tags

Not counted towards your tag license limit.
Comm Link Sequencer tags may be added when your application includes several device drivers that must share the same communication channel and that are themselves unable to determine when the channel is free. Newer radio devices that rely on a TCP/IP or UDP/IP port are particularly likely to experience this problem. This tag will automatically serialize access to the communication channel.

> Note: Use the Comm Link Sequencer tag only if you are experiencing communication problems that can be attributed directly to multiple devices attempting to use the same channel at the same time. If device access does not need to be serialized, then doing so will slow communication.

The Comm Link Sequencer is not required when multiple device drivers share the same port. Access to a single port is always serialized. The Comm Link Sequencer extends this sequencing across multiple port tags. No configuration is required beyond the creation of a Comm Link Sequencer tag as a parent (or ancestor) of the port tags that will use it.

You can monitor this tag by drawing it on any page. The value will match the number of pending requests.

Comm Link Sequencer Properties: ID Tab

No configuration is required beyond the standard Name, Area and Description. If adding more than one Comm Link Sequencer, do so by making each the ancestor of the port it is to sequence.

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

    Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

    The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Comm Link Sequencer Tag Widgets

The following widgets are available to display information about your application's Analog Input tags:

Alarm Priority Box	Alarm Priority Icon
Animated Image Widget	Bottom Bar Widget
Color Fill Widget	LCD 5x7 Matrix Widget
Draw Text Widget	LCD Bar Widget
LCD 7 Segment Widget	Left Bar Widget
Meter 1 Widget	Meter 2 Widget
Meter 3 Widget	Glass Gauge Widget
Classic Gauge Widget	Retro Gauge Widget
Cockpit Gauge Widget	Aqua Gauge Widget
Army Gauge Widget	Backlit Gauge Widget
Nautical Gauge Widget	Quarter Arc Gauge Widget
Linear Gauge Widget	Numeric Value Widget
Right Bar Widget	Top Bar Widget
Two Color Bar Widget	

Context Tags

Not counted towards your tag license limit.
Context tags are used as parents in parent-child tag structures. They provide a generic and customizable template that you can use to define any natural grouping in your application (region, station, city, sub-assembly, etc.). The parameter list for a context tag is left for you to configure. You may add whatever properties you wish. For example, you might add properties named Latitude and Longitude in order to be able to draw the Context tag on a Site Map, or you might create I/O address fields that can be used by child tags that you add to the Context tag. If you have one type of Context tag that is to be used for sites, you can add another to be used as a connector between those sites.

Your context tags can be turned into new tag-types by assigning them a type-name, then using the Create New Type command from the tag's right-click menu in the Tag Browser. All the child tags will be included as part of the new type definition. This feature is especially useful if the child tags are configured using parameter expressions so that their properties are based on values defined in the parent tag.

Context tags can be given site parameters, allowing them to be located on a site map. Related configuration parameters allow you to choose the icon that will represent the site and the page that will open when an operator clicks on that site. You can also configure how and whether the site will be shown in the Sites List page.

Note: Many other tag types also count as "sites", including Polling Drivers and Station tags.

If you have created one or more sites, you can choose to configure another context tag as a connector. Connectors are drawn as a line (you configure the line style) between two sites that have latitude and longitude values. Connectors can have any properties you care to assign and any child tags you want, thus you can use them to represent pipes or transmission lines configured with all the characteristics you require. Connectors are straight lines between two points. You can create a corner by adding a third site to your map, located between the first and second site, and configuring that intermediate site so that it is not displayed on the map. Add one connector between the first site and the intermediate site, then add a second connector between the intermediate site and the destination site.
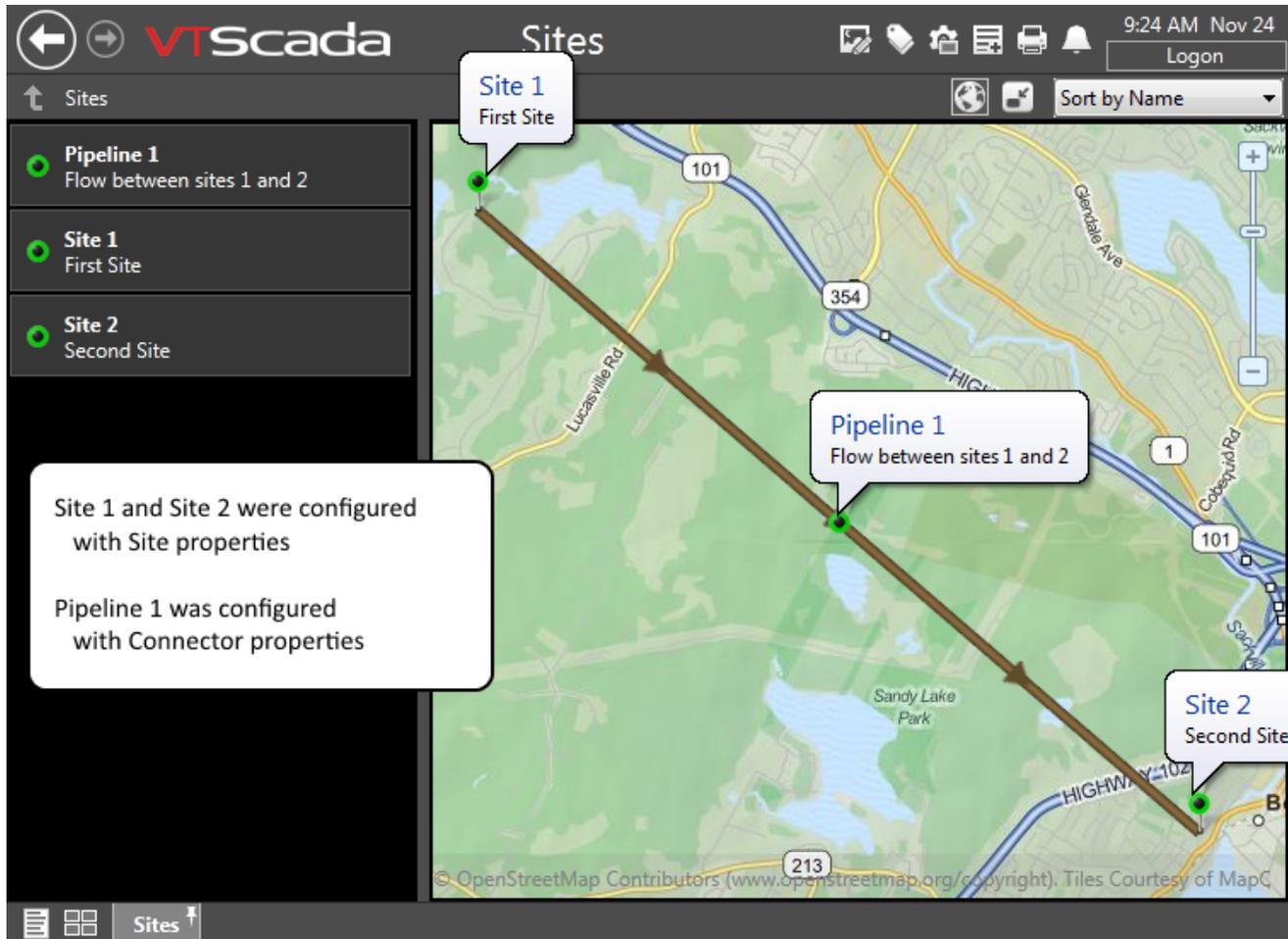
Configuration Notes:

Site-related properties or Connector-related properties are most easily added by clicking the Add Site Parameters button in the Settings tab. After adding the properties you will find it easier to configure site or connector properties using the tools in the Site Display tab.

- A Context tag can be a site or it can be a connector, but it cannot be both. If you click Add Site Properties, all connector-specific properties will be removed. If you click Add Connector Properties, all site-specific properties will be removed.
- If adding a Type name, ensure that it will be unique to your application, and will not conflict with any existing type such as "Pump" etc. It is best to add a prefix or suffix that is unique to your organization or application.

- When adding properties, ensure that property names do not conflict with VTScada reserved words such as "Address" or "Area". It is best to add a prefix or suffix to each property name that is unique to your organization or application.

Related Information:

Design Your Own Tags – Use your context tag as the basis for a new tag type.

Automated Tag Configuration – Use expressions in child tags so that they configure themselves based on the parent's configuration.

Design Your Own Widgets – Create new multi-part widgets for your multi-tag types.

Site Map – Mapping in general. Sub-topics relate to linking at tag, such as your context tag, to a map.

Context Properties: ID Tab

The Context Tag's ID tab contains an extra field named, "Type". If, after creating the context tag, you right-click on it in the tag browser and select "Create Type," you will create a new tag type, named after the contents of the Type field.

The name you provide for the type must be a single word and must not duplicate the name of any existing type of tag. An error dialog box will warn you if invalid characters are used, or if the name supplied cannot be used as the name of a new tag type.

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Context Properties: Settings Tab
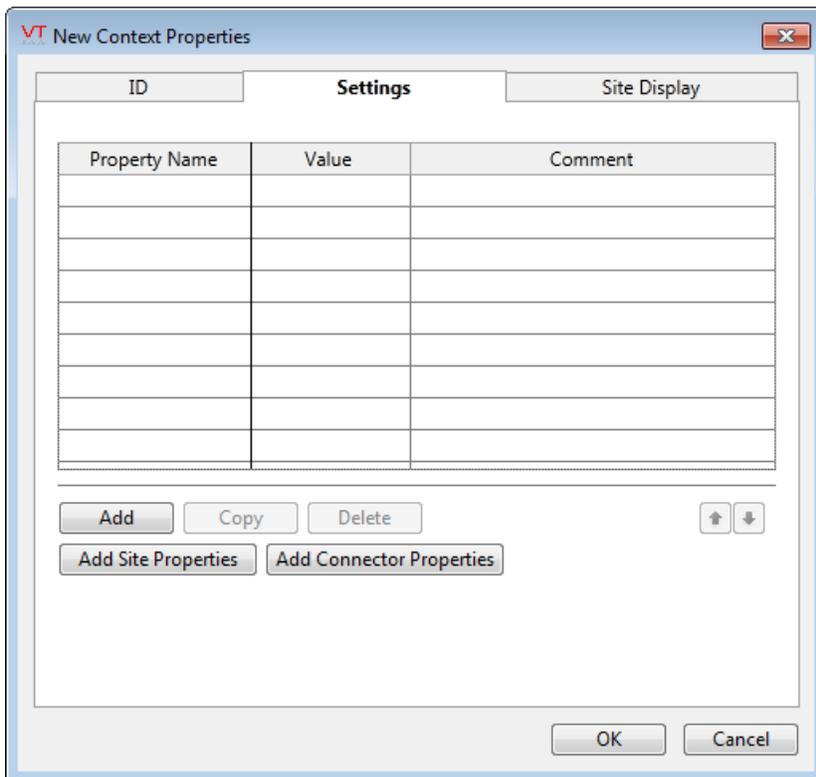
Context tags may be given a list of named properties. Each copy of the tag may then be given its own unique values for those properties. There is no limit on the number of named properties that you may create.

> Note: Property names may not duplicate VTScada reserved words. To be safe, adopt a naming convention that adds a distinct identifier to every name (for example "MA_Name" for My Application...)

By defining properties in a context tag, you provide a method for children of that tag to adapt their configuration values to the parent

instance via Parameter Expressions. For example, I/O addresses may be calculated from a BaseAddress value stored in the Context tag instance. To create a new property click Add, which opens the Add Property dialog:
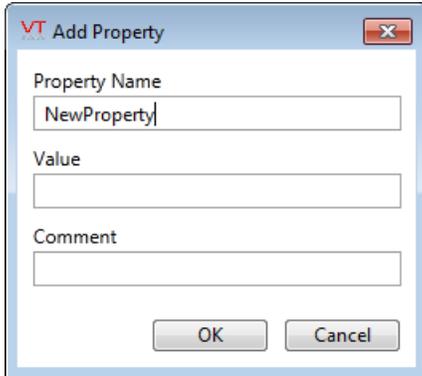
Property names must be unique, not just in the list of properties, but in the tag itself. VTScada properties such as Name, Area and Description may not be used. Attempting to do so will cause the "Name in Use" dialog to open.

Property names must include letters ("Property2" is valid, but "2" is not). The property name may *not* include a space or any of the following char-acters: ! , @ % ^ & * ( ) – ` ~ : ; " ' < > ? / { } [ ] | \ .

You can change any property name, value or comment by clicking on the field in the settings list. Editing is done in-place.

If you turn the context tag and its children into a new type using the Create New Type command in the tag browser, you can still edit the list of properties for the type definition as a whole. See: Edit Parent Type Properties.

Add Site Properties.

Note: Connectors are not sites, and sites cannot be used as connectors. Connectors do not have Latitude and Longitude properties. Sites do not

have other start and end sites.

To create connectors, you must have both tags configured as sites and tags configured as connectors.

Click this button to add the five properties that relate to site maps, Latitude, Longitude, CustomDetailsPage, CustomMapIconParm and SiteListDisplay. After adding the properties it is easier to assign values for the properties using the Site Display tab. *Connector-specific properties will be removed when you add site properties.*
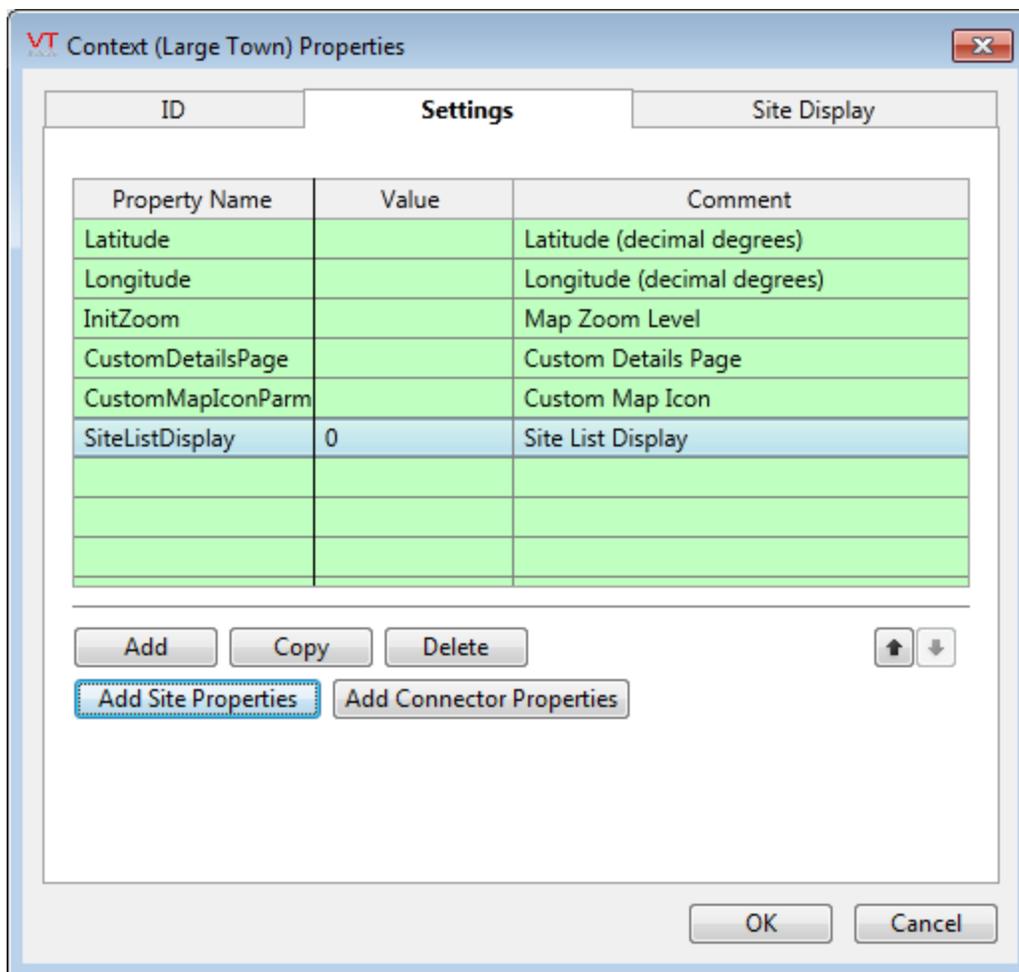
- Values for all five can be set using the Site Display tab.
- Latitude and Longitude are required if this context tag is to be added to a site map as a pin.

- SiteListDisplay is necessary if you wish to control whether the tag behaves as a "site" or a "folder" in the Site List page, or should not be shown at all. In the Site List, a "site" will open to a Site Details page when clicked. A "folder" will open within the sites list to show sub-sites (if any) and immediate I/O child tags.

- CustomDetailsPage is required only if you have or will create a custom page to display the child tags of this context tag. Otherwise, you may delete this property and use the standard Site Details page.

- CustomMapIconParm is required only if you intend to create a custom pin to mark this Context tag on a Site Map. Otherwise, you may delete this property and use a standard pin.

Add Connector Properties

Note: Connectors are not sites, and sites cannot be used as connectors. Connectors do not have Latitude and Longitude properties. Sites do not have other start and end sites.
To create connectors, you must have both tags configured as sites and tags configured as connectors.

Click this button to add the five properties that relate to site maps, Latitude, Longitude, CustomDetailsPage, CustomMapIconParm and SiteListDisplay. After adding the properties it is easier to assign values for the properties using the Site Display tab. *Connector-specific properties will be removed when you add site properties.*

- CustomDetailsPage is required only if you have or will create a custom page to display the child tags of this context tag. Otherwise, you may delete this property and use the standard Site Details page.

- StartSite and EndSite are required for all connectors. These will be configured with the two tags that hold site properties to mark the latitude and longitude of the start and end of the connector line.

- ConnectorColor defines the color of the line. By default the property is gray, defined using an Alpha value of FF and RGB values of E6E6E6. <FFE6E6E6> You may use any of the numeric values from the VTScada color palette instead.

- ConnectorStyle defines the line type. The drop-down list of the available styles is easier to use than assigning numeric values,

which match those provided in the list of VTScada line styles. Note that the line or pipe width cannot be configured as it will scale as the zoom-level changes.

- ShowConnectorArrows defines which of the three possible arrows should be shown in the display. This is stored as a bit-wise value where bit 0 matches the arrow at the 1/4 position on the line, bit 1 matches the arrow at the half-way position and bit 2 matches the arrow at the 3/4 position. Arrows will always point from the start site towards the end site.

- CustomMapIconParm is required only if you intend to create a custom widget to mark this Context tag on a Site Map. Otherwise, you may delete this property and use the standard Site Draw widget.

- SiteListDisplay is necessary if you wish to control whether the tag behaves as a "site" or a "folder" in the Site List page, or should not be shown at all. In the Site List, a "site" will open to a Site Details page when clicked. A "folder" will open within the sites list to show sub-sites (if any) and immediate I/O child tags.

Filter Child Types in the Sites Page

As an example, your custom type might be a city. Within that city (and as child tags of this type) there might be lift stations and other custom types that qualify as sites for the purpose of this list.

When the city tag is viewed in the sites page, you will be able to expand the list to view all subordinate sites. But, you might want to restrict that list to only the lift stations.
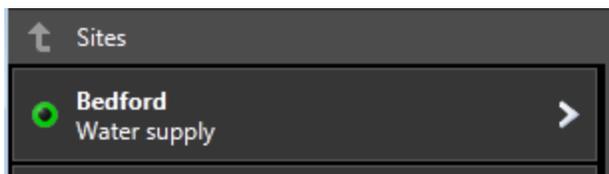
Click to view child sites

You can filter the list of child sites by adding the parameter CustomSiteListFilterType to the parent type, and setting the value of that property to the type name that should be shown in the list. You could also specify the name of a group of types, such as "Station" to filter for tags such as the MultiSmart and MPE.

The filter affects only the immediate children of the tag where it is defined.

Context Properties: Site Display Tab

Note: Before using this page, you must add either site properties or connector properties in the Settings tab.

(Site Configuration)

(Connector configuration)

There are two ways to prevent a site's location pin from being shown on a map: Leave the Latitude and Longitude values empty, or deselect the Show option for the map icon. If the site has latitude and longitude values, but the icon is not displayed, then the indicator beacon will pulse while the cursor is held over the site's coordinates on the map, even though the pin is not displayed.

Latitude and Longitude (Site configuration only)

Holds location coordinates for this tag, thereby allowing it to be represented on a Site Map page if the map icon is shown.

Start Site and End Site (Connector configuration only)

Holds links to the two site tags, between which the connector will be drawn.

Custom Details Page

If a custom details page has been created for this context tag, then that page should be selected here. If there is no custom details page, then operators will see the standard Site Details page upon clicking the pin for this tag in a Site Map.

Map Icon

You might choose to define a location for this site, but not display it on a map. This is sometimes done to add corners for connector lines or pipes. If a custom map icon widget has been defined, you can select it here. That icon will then be used instead of the standard icon when this tag is represented on a Site Map page. If there is no custom map icon, then the standard widget will be used. A custom map icon is any tag-widget that can be linked to this tag type.

Site List Display

Assuming that this tag has site properties, it will be included in the list of a Sites page. What will happen when an operator clicks on this tag in that list depends on the Site List display choice, and on whether any of this tag's children are I/O tags. Note that the following applies only to the list section of the Sites page, not to the map.

- Display as Site: A click will open the Site Details page as a pop-up.
- Display as Folder: A click will leave focus on the Sites page, but the list will now show the child tags of this site.
- Exclude from Site List: This tag should not be shown in the list section of the Sites page.
- Only Display Child Tags: This tag is used to group other sites. Those sites should be displayed, but this one should not be

displayed. Operators will not need to navigate through this site to reach the child sites.

- Automatic: If this tag contains child sites, folder display will be used. Otherwise, site display will be used.

Connector Style and Color (Connector configuration only)

Choose whether to display the connector as a pipe or as a line using any of the built-in VTScada line styles. The default color will be gray. Use the color option to select any other color.

Map Zoom Level (Site configuration only)

If Automatic is chosen (the default) then when a map is opened, showing only this site, it will zoom to level 15, which shows only the immediate surroundings. You may select any value between 18 (the closest possible level) and 2 (showing the entire globe).
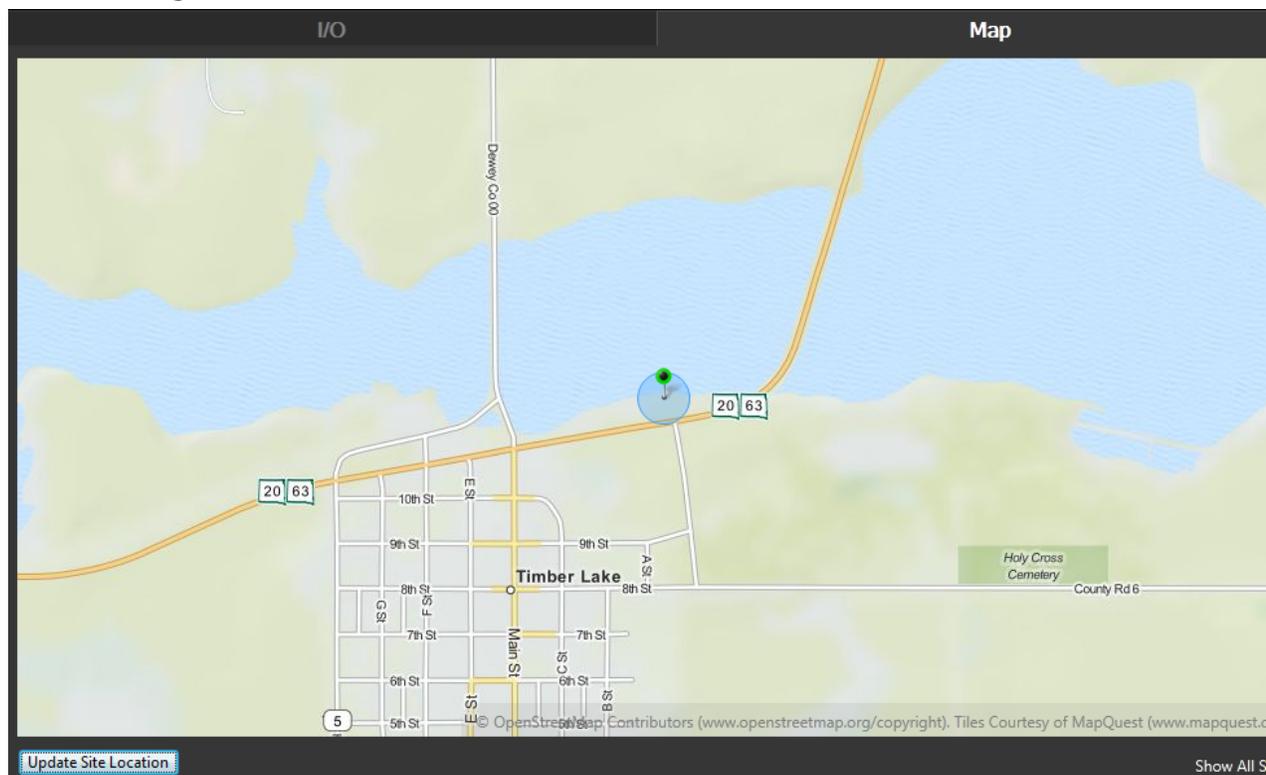
Default zoom level.

Directional Arrows (Connector configuration only)

Up to three arrows can be drawn on the connector, pointing from the first site to the second. These can be added at the 1/4, 1/2 and 3/4 positions of the visible segment of the connector.

Related Information:
Sites & Maps

Context Tag Widgets

The following widgets are available to display information about your application's Context tags. Some may not be useful unless your Context tag has been configure to have Latitude and Longitude parameters, or to be a parent to a Polling driver or station tag.

Note: There are no widgets for Connectors.

...Alarm List

...Draw HDV Widget

...Site Alarm List

...Site Details Widget

...Site Icon Widget

... Site Summary Widget

...Tag List Widget

Font Tags

Not counted towards your tag license limit.
Font tags contain style settings (including font file selection) that are used throughout your application. This is the only tag type that does not have a numeric value; instead, its value is a VTScada font type.
Several font tags are standard on every installation of VTScada, and you may create your own. The standard fonts are illustrated here:

Analog Font - 10pt Segoe UI, bold

Label Font - 9pt Segoe UI, bold

Meter Font - 12pt Microsoft Sans Serif, bold

Small Meter Font - 6pt Microsoft Sans Serif

As a general guideline, if you need fonts that differ in appearance from those that are built-in, you should create new Font Style tags rather than modifying the existing set. Built-in font style tags are used by certain VTScada user interface elements, which may be adversely affected by a significant change in font characteristics.

Font tags appear in the text format ribbon as *Saved Fonts*.

Font Style Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Font Style Properties: Settings Tab

The Settings tab shows the current property settings, and can be used to change those settings. The majority of the font characteristic setting are accessed by clicking the Change button to open the Font dialog, as illustrated. In addition to the properties set there, you can add the properties, Underline and Fixed Pitch.
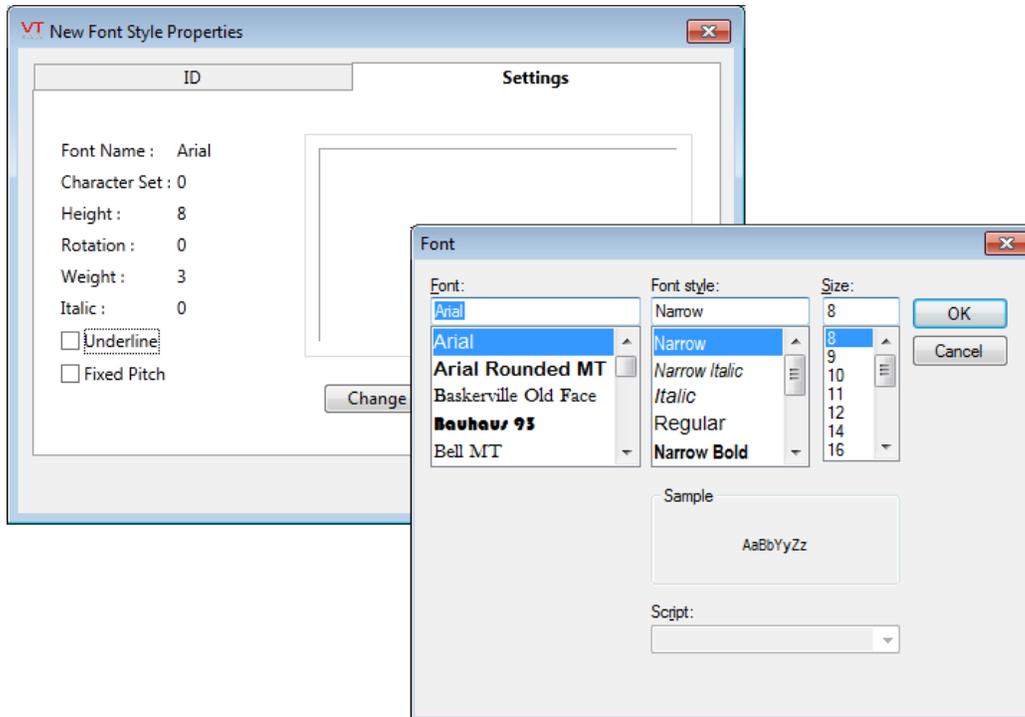
Underline

Check this option to underline all text created using this style tag.

Fixed Pitch

In the unusual event that the font you select is not available on the computer where the application is later installed, and that font comes in both fixed and non-fixed pitch versions, then checking this option will cause the search for an alternative font to be restricted to only fixed-pitch options.

Change

The Change button opens a Font dialog that enables you to configure the property settings (font, style, and size) for the selected Font Style tag.

IP Network Listener Tags

Not counted towards your tag license limit.

Add IP Network Listener tags to your application if you want to be able to accept inbound TCP or UDP connections. This tag is for use with the DNP3 and Modbus Compatible Devices, but may also be used by custom drivers. In addition to drivers, an IP Network Listener can also be used with services, such as the SNMP Agent.

The IP Network Listener will not be required in most situations. It may be useful if your TCP connections will close between polling intervals and you need to allow the RTU to connect to the driver before the next cycle, perhaps to report a fault condition. Note that most technical support calls involving the IP Network Listener tag are solved by determining that the tag was unnecessary and removing it.

While this tag will work with both protocols, it is intended primarily for TCP connections. Drivers using the UDP protocol may not need a Network Listener unless the RTU is connecting to VTScada using a different IP address each time. If using an IP Network Listener tag with a UDP con-nection, use care to ensure the port number is set correctly and is com-patible with the corresponding UDP/IP port tag configuration.

> Note: Warning: The IP Network Listener uses a socket server to accept incoming connections. Improper configuration can result in com-munications failure for UDP/IP Port tags that make use of the same local port number. Refer to UDP/IP Port Tags for more information and further warnings.

> Note: Warning: Use of an IP Network Listener Tag is generally not recommended for UDP/IP communications. The VTScada engine will already listen for UDP/IP datagram packets and place them into exist-ing UDP streams if the remote IP address matches that of the UDP stream.

Multiple Modbus and DNP3 driver tags can share the same IP Network Listener.

Related Information:

Modbus Compatible Device: Serial Tab – Modbus Compatible Device configuration

DNP3 Driver Properties: Serial Tab – DNP3 driver configuration

SNMP Agent Configuration is described in the VTScada Programmer's Guide

IP Network Listener Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

IP Network Listener Properties: Options Tab
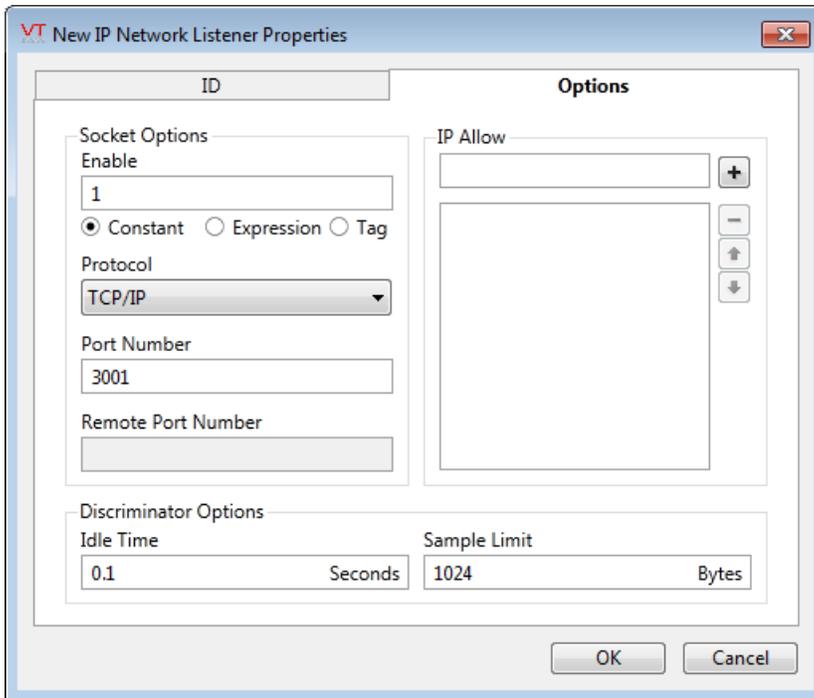
Enable

Accepts a 1 to enable or a 0 to disable inbound connections. The options Expression or Tag allow you to enable or disable connections based on user control of your choice, or based on current operating conditions. Set to 1 (online) by default for all workstations.

Protocol

Select either TCP/IP and UDP/IP, as appropriate for the device your drivers will be communicating with.

Port Number

Define the port that be used for inbound connections. Ensure that your firewall has been configured to allow connections to this port.

Remote Port Number

Enabled only for UDP/IP. UDP responses from VTScada will be sent to this port number. Leave this option Invalid in order to send responses back to the source port on the RTU.
Review the warnings in the introduction of the UDP/IP Port tag description.

IP Allow List

Optional. As a security measure, this listener may be configured to accept connections only from the IP addresses you add to this list. A range of allowed addresses may be defined by leaving portions empty.
Support is provided for CIDR[1] ranges. For example, 192.168.1.0/24 or 192.0.0.0/8

Discriminator Options - Idle Time and Sample Limit

These two values control how incoming data is sampled. Sample Limit specifies the number of bytes to be received before the stream is sampled. In the event that the stream is not changing, the passage of Idle Time seconds will trigger a sample. The default values were selected to work in most cases, but you may find that Sample Limit need only be set to 10 bytes when this tag is used by a DNP3 driver.

IP Network Listener Tag Widgets
The following widgets are available to display information about your application's IP Network Listener tags:
Comm Indicator Widget

Menu Item Tags

Not counted towards your tag license limit.

---

[1]Classless Inter-Domain Routing

Menu Item tags are used to represent page menu items and drawing tools within the various palettes. Folders and dividers within menus and palettes are also stored in Menu Item tags.

These tags do not have an ID tab. The configuration panel will vary according to the type of menu item that you are creating or editing.
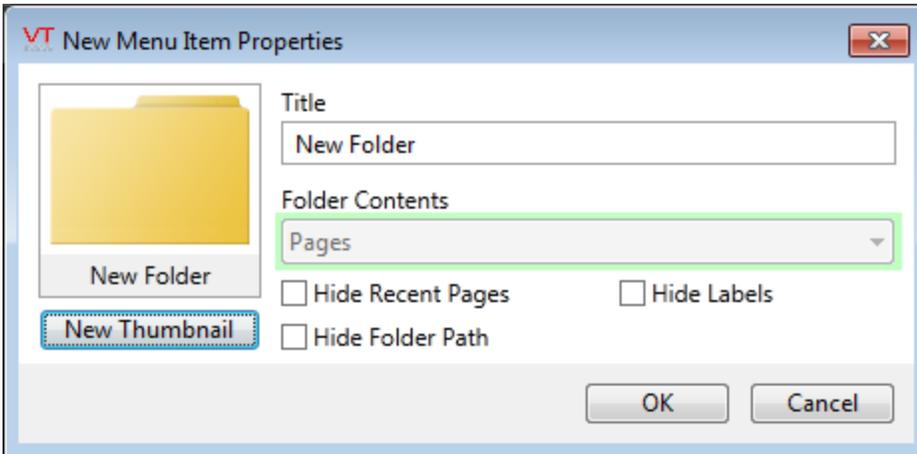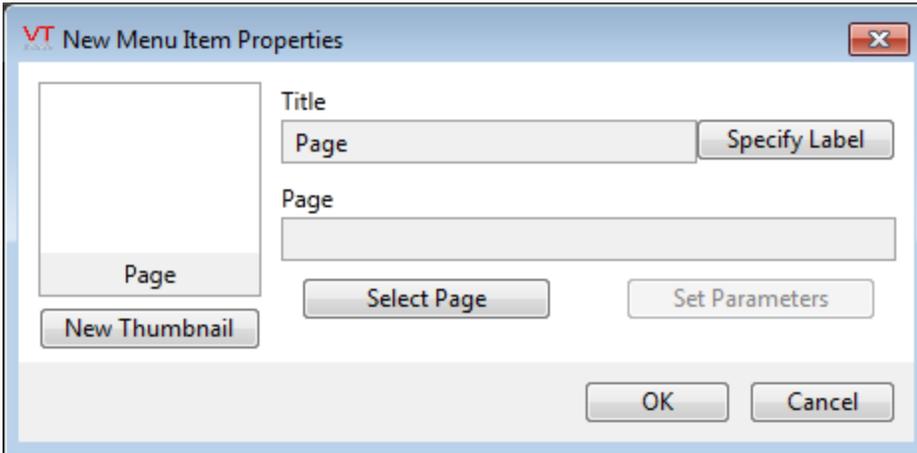
Menu Item tags can have start conditions, like any other tag. Since there is no ID panel, start conditions are added by right-clicking on the tag in the Tag Browser and selecting "Start condition" from the context menu. One reason that you might want to add a start condition is to hide menu entries from certain users based on their security role. For example, if the start condition is the following expression:

```
PickValid(\SecurityManager\SecurityCheck(\Se-
curityManager\PrivBitConfigure, 1), 1);
```

Then, only operators who have configuration privileges will be able to see this item in either the main menu or the tiled menu. This technique does not apply to page buttons or hotboxes.

Related Information:
Adding New Menu Items
Create a Start Condition

...Creating Expressions – VTScada Programmer's Guide – Reference for the VTScada scripting language.

Realm Display Setup Tags

Not counted towards your tag license limit.
The usage of the Realm Display Setup tag is described in the VTScada Internet Server chapter. It exists only to hold configuration settings that set the level of control granted to operators connecting to an application from a VTScada Internet Client. There are no widgets, no log enabled variables and no group memberships.

> Note: Realm Display Setup tags do not apply to mobile device connections.

Related Information:
Internet Realms

Realm Display Setup: ID Tab
The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:
> Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area
> The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Realm Display Setup Tag: Settings Tab

The settings tab controls the overall display options. The first drop-down selector is where you choose which realm the settings will apply to. Only realms that include the current application will be available in the list. You also have the option of making the settings apply to all realms that include this application, rather than just one at a time. To decide whether to use that option, you must choose whether you want to provide different settings to different VIC stations – which would be done via different realms. Changes to this tag take effect immediately. There is no need to restart the application after adding or changing a Realm Display Setup tag.

You can add exactly one Realm Display Setup tag for each realm that includes the application.
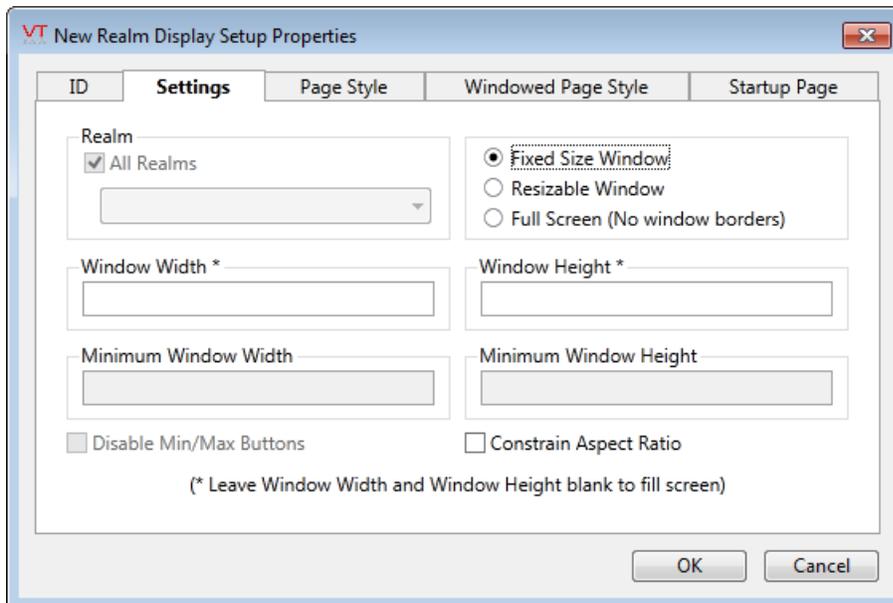
Fixed Size Window, Resizable Window or Full Screen

Running full screen implies that the window width and height will be whatever the full (maximum) screen size is.

The full screen option does not apply to Anywhere Client connections.

When a VIC is running in full screen, the Windows title bar and other window borders will not be displayed.

If Fixed Size Window is selected, you gain more control over the VIC window at the server side, while not allowing size control at the VIC. You will be able to set the VIC window size and page style options at the server with this option selected.

The operator will still be able to move the window around on his screen, and will be able to minimize and restore it, provided that the Disable Min/Max Buttons option has not been selected.

Note that if the specified Fixed Size is larger than the resolution of the computer running the client session, scroll bars will appear on the display.

By selecting Resizable Window, you provide the client with the greatest level of control. Operators will be able to re-size their screen at will, limited only by the optional minimum width and height that you may set in this dialog box.

Your choice of display option will affect which of the remaining controls on the screen are available. The description of each of those controls will tell you which window options it is relevant for.

Note that leaving the initial Window Width and Height blank (if available) will result in the VIC taking the maximum screen area as its initial size upon opening.

Initial Window Width and Height Fields

If you have selected Fixed Size Window above, these controls

set the window width and height for the client. If you have selected Resizable Window, then these controls set only the initial window size for the first time the application is opened on the client. On subsequent startups, the resizable window will remain at whatever dimensions the operator set.

If Full Screen has not been selected and these two values are also not set, the following configuration options (if set) will be used to determine a default size:

- If BrowserWidth and BrowserHeight are defined in the application properties, these values will be used.
- Otherwise, if DispMgrWidth and DispMgrHeight are set in the application properties, those values will be used.
- Otherwise, the default width and height will simply fill the screen.

These settings are maintained by the client and so (for a re-sizable application) apply only for the first time the client is opened.

Minimum Window Width and Height Fields

Sets the minimum width and height that the client is able to resize to. Defaults to 25 pixels.

If these values are not set for a resizable client, then the application will check the configuration file to see whether DispMgrMinWidth and DispMgrMinHeight have been set and if so, will use those. Otherwise, no minimum values will be set.

Disable Min/Max Buttons

This option is not available if Full Screen option is selected. Does exactly what it says to the Min/Max buttons at the top right of the client's screen.

Constrain Aspect Ratio

Available only if the Resizable Window option is selected. Forces the VIC screen to maintain a constant aspect ratio (width to height) while it is being re-sized.

Realm Display Setup Tag: Page Style Tab

The Page Style tab includes settings that apply to VIC sessions that display in Full Screen Mode (as set on the Settings Tab). This is a list of which elements of both the Title Bar and the Task Bar you wish to make visible.
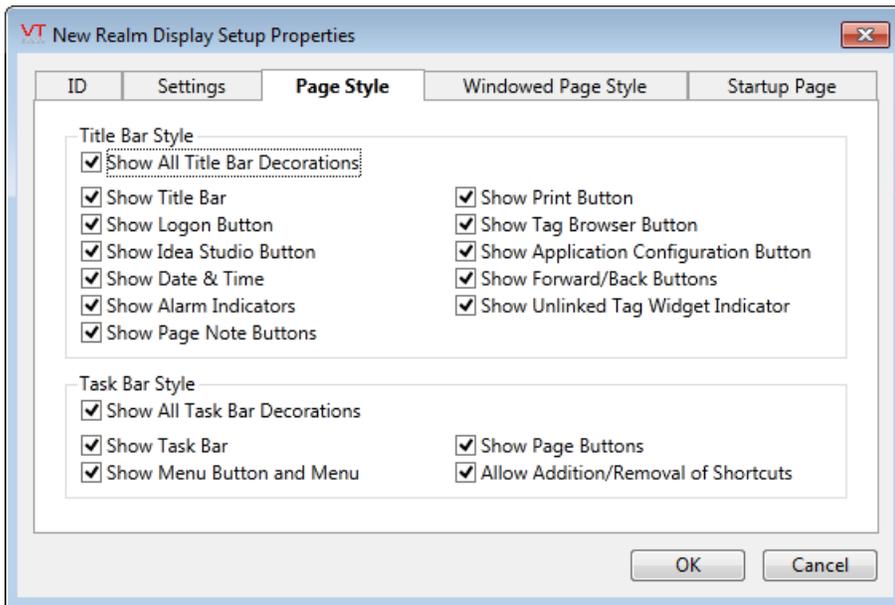
Elements of the Title Bar and Task Bar:
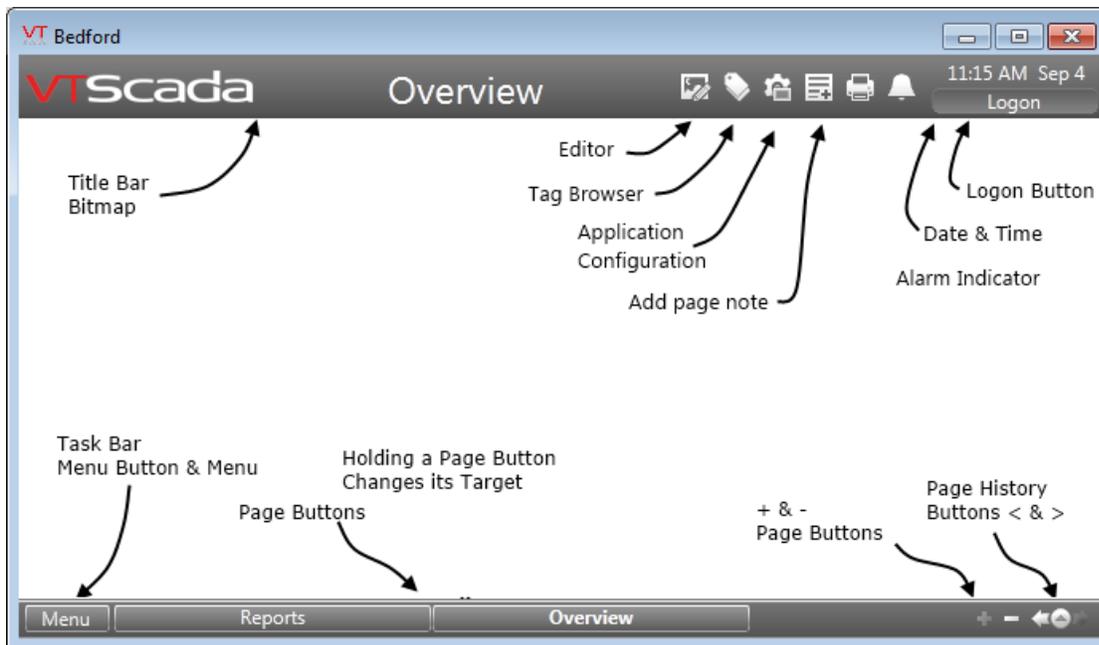
Title Bar Style:

Provides a series of yes-or-no check boxes controlling which parts of the title bar should be displayed on the VIC including whether the title bar should be displayed at all.

Task Bar Style:

Provides a series of yes-or-no check boxes controlling which parts of the task bar should be displayed on the VIC including whether the task bar should be displayed at all.

Realm Display Setup Tag: Windowed Page Style Tab

The Windowed Page Style tab includes settings that apply to pages that display as windows – whether Fixed Size or Resizable, as selected in the Settings Tab.
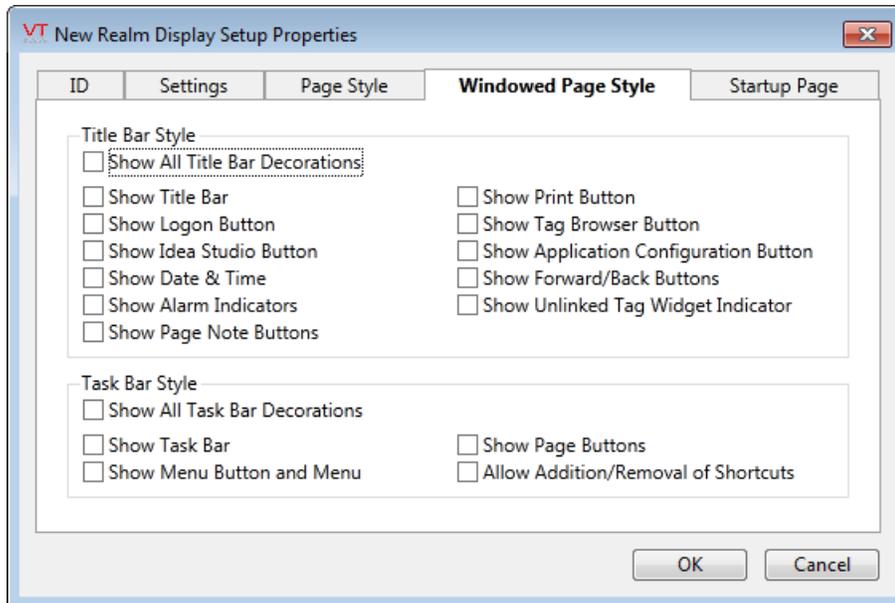
Title Bar Style:

Provides a series of yes-or-no check boxes controlling which parts of the title bar should be displayed on the VIC including whether the title bar should be displayed at all. An image to refresh your memory of the various parts of the screen is included with the previous topic, Page Style Tab.

Task Bar Style:

Provides a series of yes-or-no check boxes controlling which parts of the task bar should be displayed on the VIC including whether the task bar should be displayed at all.

Realm Display Setup Tag: Startup Page Tab

First page at startup

Provides you with control over which page will open first for users connecting to the realm associated with this tag. You may select the application's default startup page (as

configured in the Display tab of the Application Configuration dialog), the last page viewed (for that user), or you may select any page within the application to use as the start page.

Available Widgets

Realm Display Setup tags work without being drawn.

Style Settings Tags

Not counted towards your tag license limit.

Style Settings tags allow you to link visual indicators to tag types, helping to ensure that all indicators for those tags follow a consistent visual style. For example, your industry may specify that a digital value in state 0 (off) should be shown in red, and state 1 (green) should be shown in green. In other industries, the opposite is true. For widgets that obey the Style Settings value, you need only set your preferences once, and all displays will use a consistent format.

Note: Not all widgets recognize or use the Style Settings tag.

One default Style Settings tag, "System Style", is included in every new VTScada application. It will be used by every style-aware tag that has not been configured to use an alternative selection, including no Style Settings tag.

Many widgets and most tags can be configured to use a Style Settings tag. In the event that both a widget and the tag linked to it are configured with different Style Settings tags, the following hierarchy will be used. Note that the configuration of the widget takes precedence over the configuration of the linked tag.
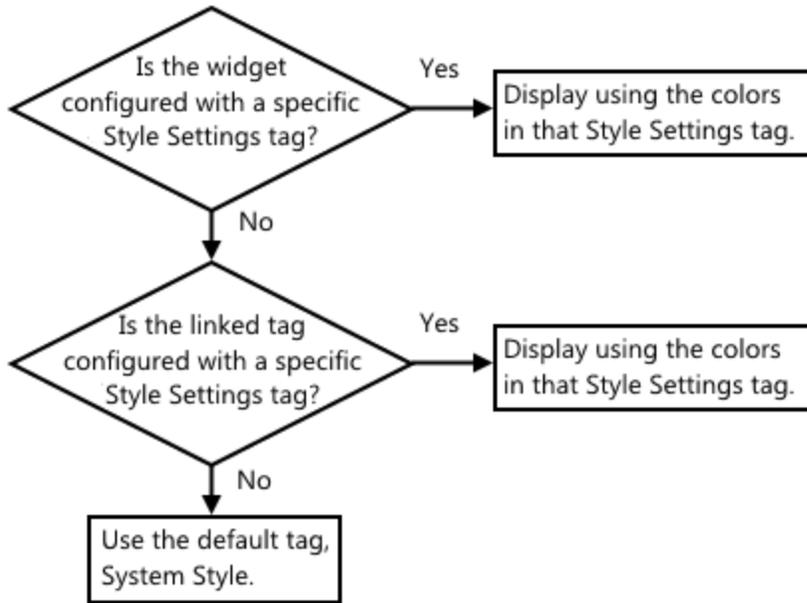
Troubleshooting:

- This object that I've drawn doesn't change with a style setting.
- I don't see how to link this object to a style setting.
  Style Settings are used only by certain widgets. They cannot be used with text, pipes, images, shapes or with widgets that aren't designed to use a Style Setting. It is possible to animate any property of any object, using an expression or a linked tag's value, but that is not the same as using a Style Setting tag.
  Every widget's description states whether it can be used with a Style Setting. Also, widgets that can use a Style Setting are marked with a star in the table of contents.

Style Settings Properties: ID Tab

The ID tab of every tag includes the same four elements. Name, Area, Description and Help ID.

Name:

Uniquely identifies each tag in the application. If the tag is a child of another, the parent names will be displayed in a separate area before the name field. You may right-click on the tag's name to add or remove a conditional start expression.

Area

The area field is used to group similar tags together. By defining an area, you make it possible to:

- Filter for particular tag groups when searching in the tag browser
- Link dial-out alarm rosters to Alarm tags having a particular area
- Limit the number of tags loaded upon startup.
- Filter the alarm display to show only certain areas.
- Filter tag selection by area when building reports

When working with Parent-Child tag structures, the area property of all child tags will automatically match the configured area of a parent. Naturally, you can change any tag's area as required. In the case of a child tag, the field background will turn orange to indicate that you have applied an override.

To use the area field effectively, you might consider setting the same Area for each I/O driver and its related I/O tags in order to group all the tags representing the equipment processes installed at each I/O device. You might also consider naming the Area property for the physical location of the tag (i.e. a station or name of a landmark near the location of the I/O device). For serial port or Roster tags, you might configure the Area property according to the purpose of each tag, such as System or Communications.

The area name may be up to 256 characters in length. You may define as many areas as you wish and you may leave the area blank for some tags (note that for Modem tags that are to be used with the Alarm Notification System, it is actually required that the area field be left blank).

To define a new area, type the name in the field. It will immediately be added. To use an existing area, use the drop-down list feature. Re-typing an existing area name is not recommended since a typo or misspelling will result in a second area being created.

There is no tool to remove an area name from VTScada since such a tool is unnecessary. An area definition will exist as long as any tag uses it and will stop existing when no tag uses it (following the next re-start).

Description

Tag names tend to be brief. The description field provides a way to give each tag a human-friendly note describing its purpose. While not mandatory, the description is highly recommended.

Tag descriptions are displayed in the tag browser, in the list of tags to be selected for a report and also on-screen when the operator holds the pointer over the tag's widget. For installations that use the Alarm Notification System, the description will be spoken when identifying the tag that caused the alarm.

The description field will store up to 65,500 characters, but this will exceed the practical limits of what can be displayed on-screen.

Style Settings Properties: Analogs Tab

Fields in the Analogs tab of this tag are used to set colors and labels for LCD widgets.
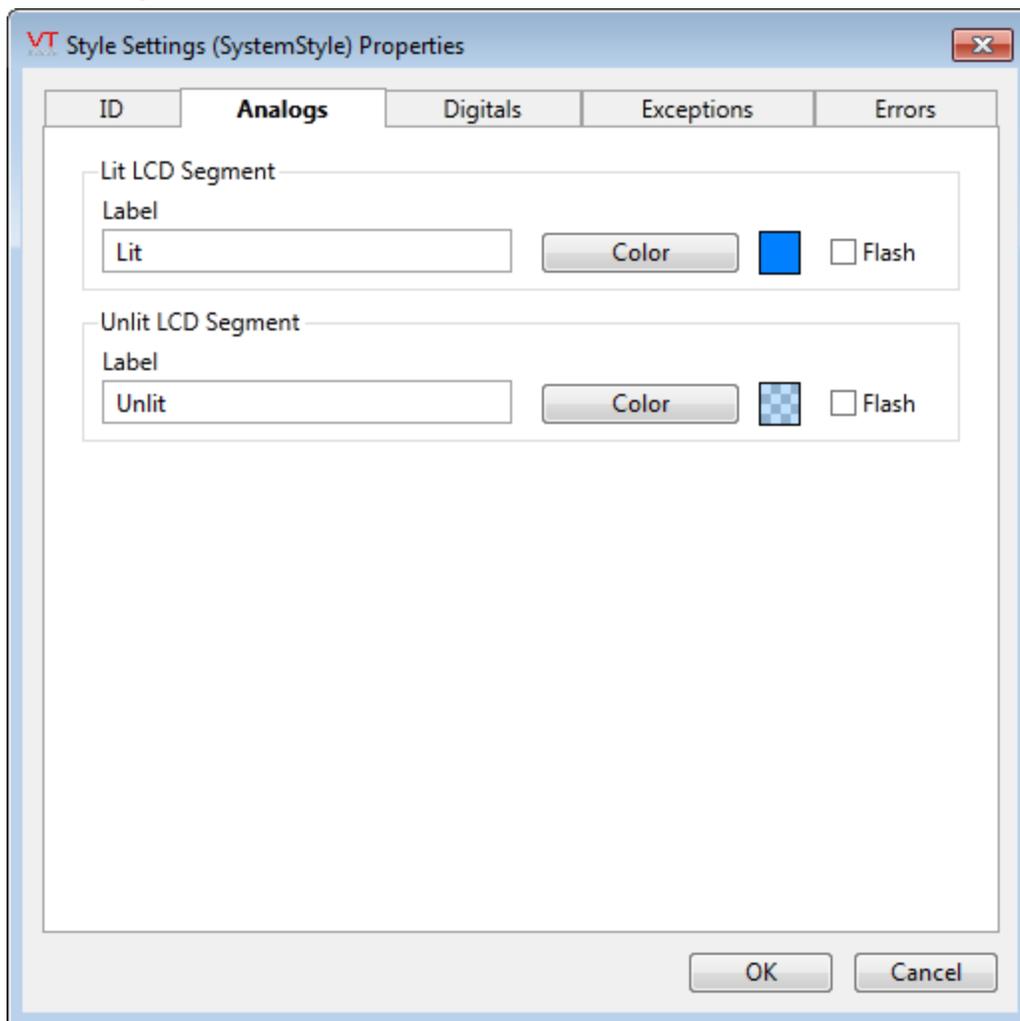
Lit LCD Segment

When an LED indicator widget is used for a numeric display, the lit portion will have this color.

Unlit LCD Segment

When an LED indicator widget is used for a numeric display, the unlit portion will have this color, typically using an opacity value less than one.

Style Settings Properties: Digitals Tab

Fields in the Digitals tab of this tag are used to set color and text for equipment/status widgets, indicator-lights and other style-aware widgets for digital tags.
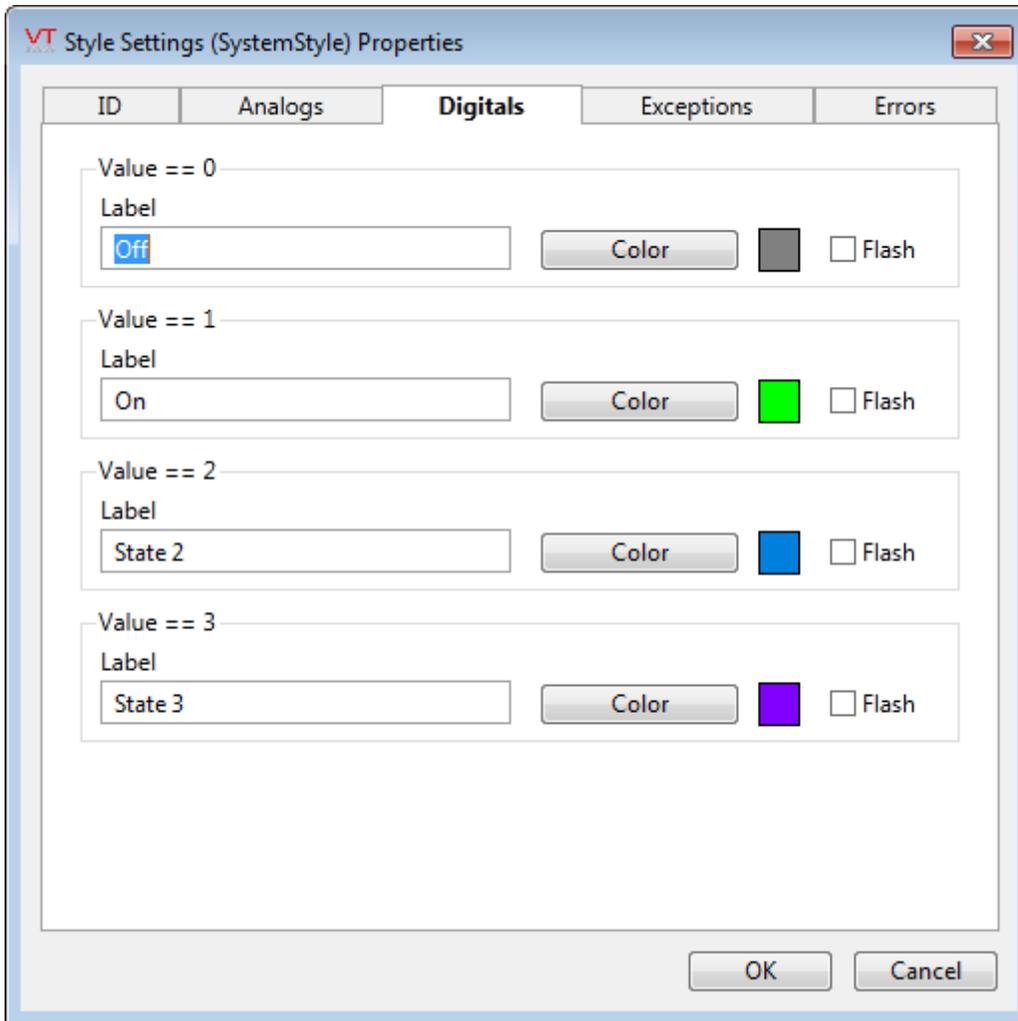
Value == 0

Set the text and color of the text, displayed when the associated tag is in the zero state.

Value == 1

Set the text and color of the text, displayed when the associated tag is in state one.

Value == 2 and Value == 3

For digital tags that use two address fields, and therefore can

display up to four states, this label and color will be displayed when the associated tag is in state two.

Style Settings Properties: Exceptions Tab
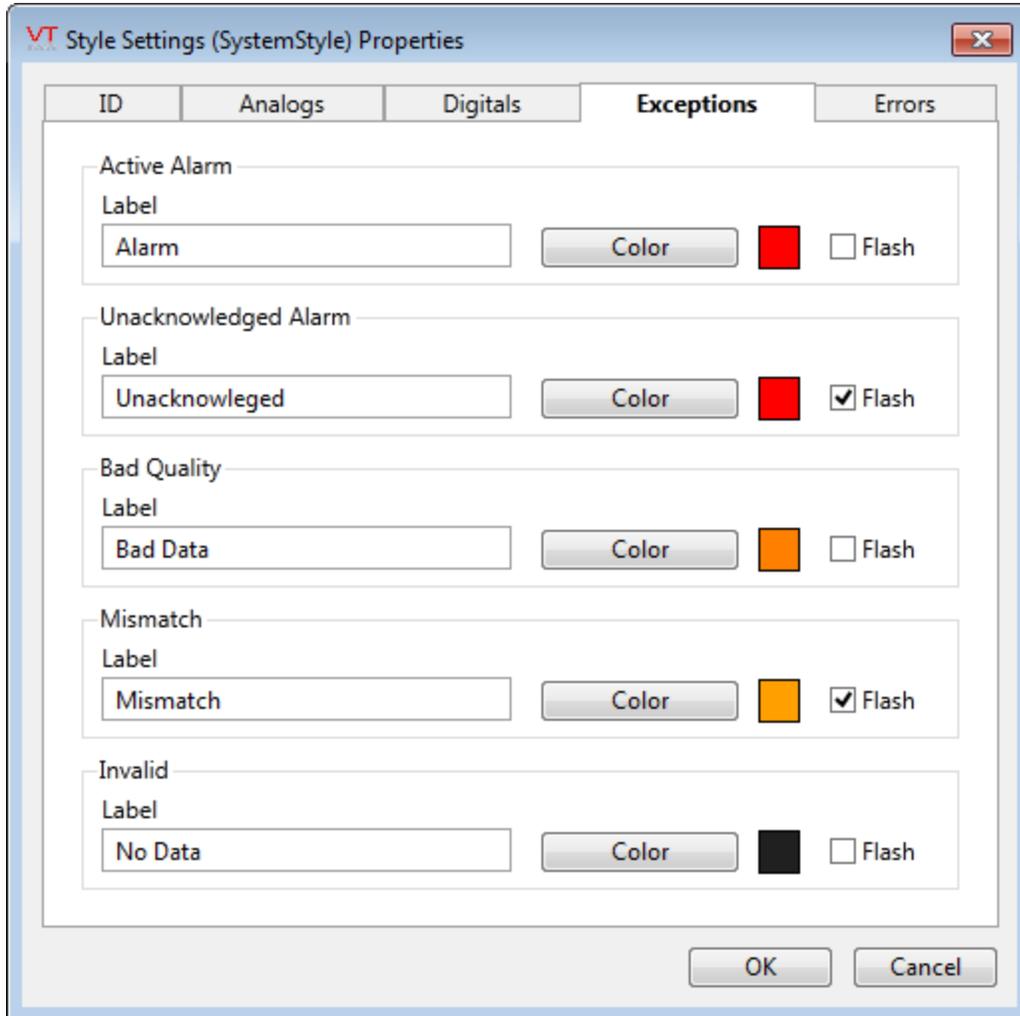
Active Alarm

Set the text and color of the text, displayed when the asso-ciated tag has an active alarm.

Unacknowledged Alarm

Set the text and color of the text, displayed when the asso-ciated tag has an unacknowledged alarm.

Bad Quality

Custom programming, using the VTScada scripting language, must be done in order to make use of the Quality field of a tag. If the associated tag is using the Quality field, the associated widget will use the label and color set in this field to indicate a bad quality value.

Mismatch

For output tags, if the feedback value for the associated tag does not match its current value, or value last written, this text and color will be displayed on the associated widget.

Invalid

Set the text and color of the text, displayed when the associated tag is in an invalid state. Invalid states usually indicate a break in communications, whether due to faulty configuration or communication.

Style Settings Properties: Errors Tab

The value property of tags in the ports group, drivers group, the Modem tag and the SMS Appliance tag, will indicate whether or not it is in an error state. When drawn using a Status Color Indicator or Indicator Light widget, the tag's value, which indicates the error state, will be shown using the colors and labels configured here rather than the colors defined in the digitals tab.
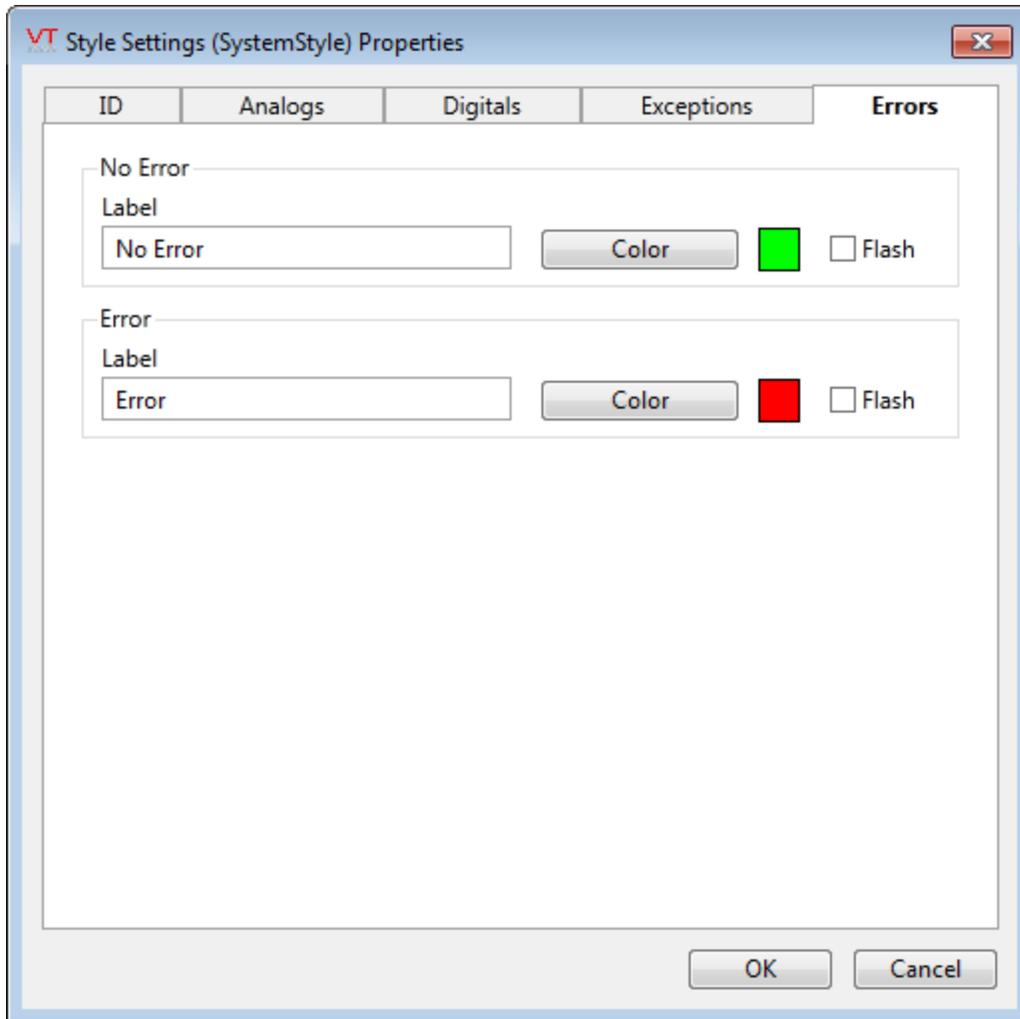
No Error

Color to be used when the linked tag is not in an error condition.

Error

Color to be used when the linked tag is in an error condition.

Related Information:

...Merit Tab and Quality Tab – Description of the Quality attribute of tags.

...Equipment / Status Color Indicator – Widgets that use the Style Settings tag.

...Illuminated Toggle Switch Widget

...Indicator Light Widget

...LCD 5x7 Matrix Widget

...LCD 7 Segment Widget

...LCD Arc Widget

...LCD Bar Widget

...LCD Meter Widget

...LCD Ring Widget

Modem Configuration and Use

Modems may be used for I/O communication between VTScada and a remote telemetry unit. They are also used for voice and pager calls by the Alarm Notification System.
All modems are under the control of the Modem Manager, a VTScada service that enables modems distributed across a number of different workstations to be managed as a common pool. The Modem Manager Server and clients cooperate to provide modem pool management and to set up data paths between modem users and the allocated modems. The modem pool is managed using a time priority queue.
Outgoing calls are queued and are then executed at the requested time. You can receive calls on any modems, even on those on backup servers or clients. Call queue information is distributed to all copies of Modem Manager on a networked system so that queue integrity is maintained, even after switching to a backup server. If a call has not yet been established, and server fail-over has occurred, the call will be completed by the backup server that will handle the Modem Manager's preserved queue.
There must be one Modem tag for each modem in the system. For the Alarm Notification System, the area field of the Modem tag must be left blank. If using a data modem for I/O, you can force data calls to use that

modem by configuring the area property to match that of its serial port and adding the application property, UseSerialAreaInModemCall, set to TRUE (1).

All modem access uses TAPI – the Microsoft telephony management layer that enables programs to share and pass calls. TAPI enables telephony resources to be shared between multiple applications. It interfaces to a modem driver so VTScada doesn't have to be configured for the specifics of the modem.

Both a modem audio driver and a TAPI Service Provider (TSP) are provided by Trihedral. These have been designed to avoid several of the problems associated with the older Unimodem V driver, including its limitation in detecting and switching between voice vs. data calls.

The Unimodem V Driver is still available, but will not display in the list of drivers unless you set the application property, UseUnimodem to 1.

Note: if your system includes more than one modem, and those modems are attached to servers located in regions with different area codes, then use the **canonical address format**[1] for each phone number and configure the server's location within the Windows™ Phone and Modem Settings dialog.

Some internal modems appear to enforce a "serial port" baud rate of 115200, even though they don't use a real serial port, they just appear to be a COM port from a Windows software point of view. Since raw 16-bit linear audio at 8000Hz requires 160000 baud, the decoder on the modem is starved for data, and data accumulates in the ModemTSP queue. For this reason, you should avoid using the 16-bit linear format. Both µ-law and A-law are available on most modems and give excellent audio quality. With a data rate of ~80000 baud, they are well under 115200 baud.

---

[1]A canonical phone address is a text string with the following structure:
+ CountryCode SPACE [(AreaCode) SPACE] SubscriberNumber | Subaddress ^ Name CRLF …

Related Tasks:

...Verify That Your Modem has Voice Support – How to know if your modem can be used by the Alarm Notification System.

...Install a Voice Modem – Steps for proper configuration of drivers.

... Speech Lexicon Configuration – (spoken and dialed alarms) Adjust the voice's pronunciation.

...Configure Modem Logging – Record a history of dialer activity.

...Remove a Voice Modem – Proper removal of drivers to ensure that a new modem will not encounter conflicts.

...Designate Modem Manager Server(s) for a Remote Application – Assigning modems in a multi-server installation.

Related Information:

...Modem Tags – There must be one modem tag for each modem.

...Alarm Notifications (Notification Options) – (Alarm Notification System) Controls which alarms are dialed out, the delay, the greeting and the initial menu.

...Modem Manager Operation – Technical reference.

...Modem Status and Statistics – Collect and display information about modem operation.

Verify That Your Modem has Voice Support

Manufacturer's claims aside, you can easily verify that a modem does in fact provide the voice support required. To do so, follow these steps:

1. Open the Windows® operating system control panel.
2. Open Phone and Modem options.
3. Select your modem.
4. Click on Properties
5. Select the Diagnostics tab
6. Click on Query Modem

The response will include a line that begins with: AT+F class = ?

➤If this line includes a ,8 at the end, then you have voice modem support.

     (e.g. AT+F class = ? 0,1,8)

➤If not, then your modem does not have the required voice support.

Voice Modem Installation and Removal

These procedures are intended for VTScada users who intend to use the voice dial-out feature of the VTScada Alarm Notification System. Follow the procedures precisely to ensure proper modem installation.
The actions required to access the various Windows® operating system user interfaces described here may differ somewhat between versions.

> Note: Caution: Proper modem installation and configuration can be difficult and time consuming. If you experience difficulties during modem installation, driver installation or VTScada Alarm Notification System configuration, contact Trihedral® Technical Support (1-800-463-2783) for assistance.

Related Tasks:

...Install a Voice Modem – Ensures that all drivers are installed and functioning correctly.

...Installing the Trihedral TSP and Modem Audio Driver – Required only if not automatically installed.

...Configure Modem Logging – Enable logging at the Windows™ device level.

...Remove a Voice Modem - Ensures that drivers are correctly removed so that a new modem may be installed.

Install a Voice Modem

If your VTScada application is running, stop the application and shut down the VTScada Application Manager before beginning this procedure. If the VTScada program is not already installed, it may be installed either before or after the modem installation.

Step 1: Install the voice modem

Your modem will come with specific instructions to guide you through proper modem installation. This will include proper installation of the modem within the computer chassis, as well as proper installation of the modem drivers. *Follow the manufacturer's instructions exactly as written.* If you have Internet connectivity, it is recommended that you download and install the most updated device drivers.

Step 2: Verify modem driver has been installed correctly

Verify your modem driver installation to ensure that the voice feature of the modem will work correctly.

Verify the voice feature as follows:

1. Access your the Control Panel of the Windows® operating system.

2. Open, Phone and Modem Options.

3. If this is the first time that you have configured a modem in this computer, you will be prompted for location information. If so, complete this information and select OK to continue. The Phone and Modem dialog box will open to the Dialing Rules tab. *Caution: Neglecting to complete the location information will cause your modem to function improperly.*

4. Select the Modems tab.

5. Choose the modem you have just installed and click on Properties. The properties dialog box for your modem will open.

6. Select the Diagnostics tab and click on the Query Modem button. When your modem responds, you will see a set of modem commands and responses as shown in the following images. A modem that is correctly configured for voice will include a comma followed by the number eight (i.e. ,8) within the responses from either the AT+FCLASS=? command or the AT#CLS=? command.

Note: If your modem query fails, then your modem driver has not been installed properly. Uninstall the modem and reinstall as per the manufacturer's instructions.

If neither of the responses include a ',8', then the modem will not sup-

port the Alarm Notification System. Purchase a modem that provides voice support.
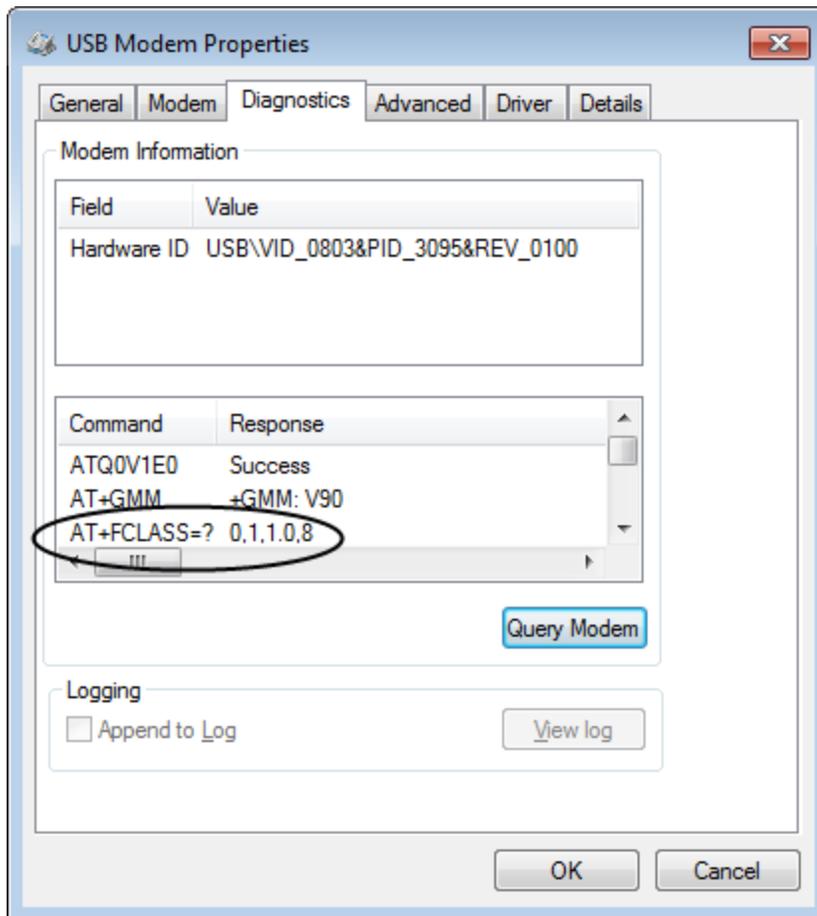
    Correct: The Response includes ',8'

7. Close Phone and Modem Options.

Step 3: Install the VTScada modem driver and service provider, if not already installed.

In most cases, the wave bus enumerator, voice modem wave device and modem TSP (Telephony Service Provider) will be installed automatically and silently. If not, you may be prompted with a "Found new Hardware" pop-up the next time that VTScada starts.

If this dialog appears, do the following:

1. Select, Install Automatically, when prompted.

2. Click, Next.

3. Select, Not This Time when prompted to connect to Windows Update.

4. Click, Next.
    As part of this installation, you will see a pop-up titled, Software Installation.

5. Click, Continue Anyway to allow installation.
    Caution: Do not click Stop Installation, or the VTScada modem drivers will not be installed.

Step 4: Verify correct installation of the VTScada voice modem TSP

Verify your VTScada voice modem TSP installation to ensure the voice feature of the modem will work correctly with the VTScada Alarm Notification System.

Verify the installation as follows:

1. Access your Windows Control Panel.

2. Select Phone and Modem Options. The Phone and Modem dialog box will open to the Dialing Rules tab.

3. Select the Advanced tab.

4. Verify that the Trihedral® Voice Modem Service Provider is listed as shown here. Note: If not, your VTScada modem drivers have not been installed correctly. Contact Trihedral Technical Support (1-800-463-2783) for assistance.
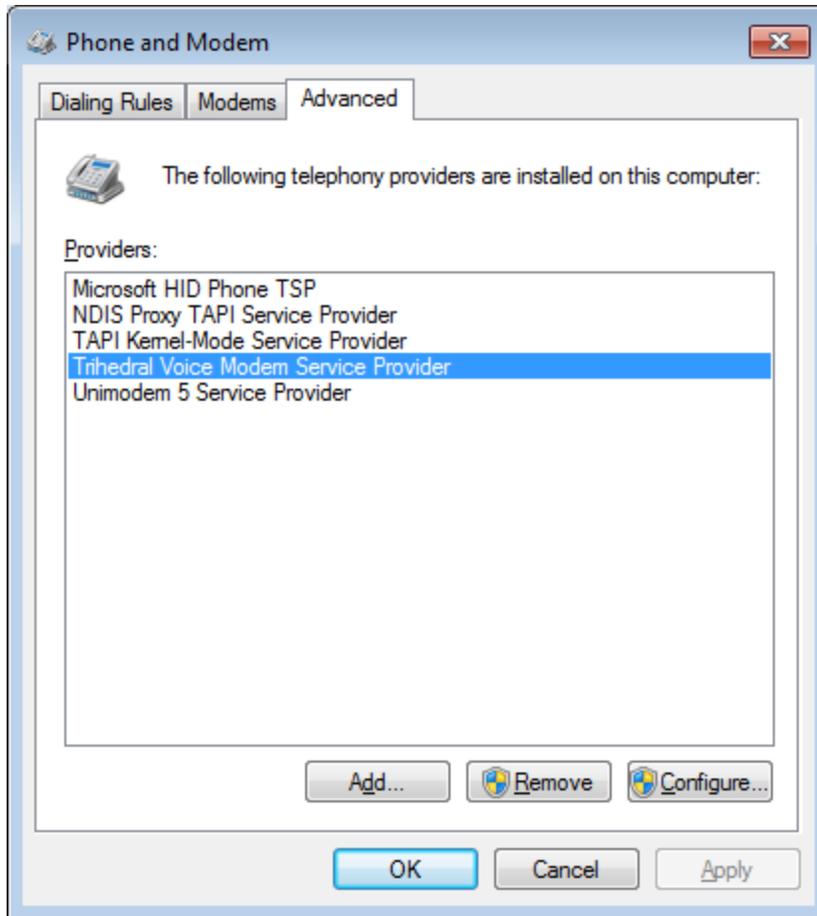
5. Select, Trihedral Voice Modem Service Provider

6. Click on the Configure button.
   A Trihedral Modems Configuration pop-up will show the modems associated with the Service Provider.

7. Ensure your new modem is listed.
   If not, your VTScada modem drivers have not been installed correctly.
   Contact Trihedral Technical Support (1-800-463-2783) for assistance.
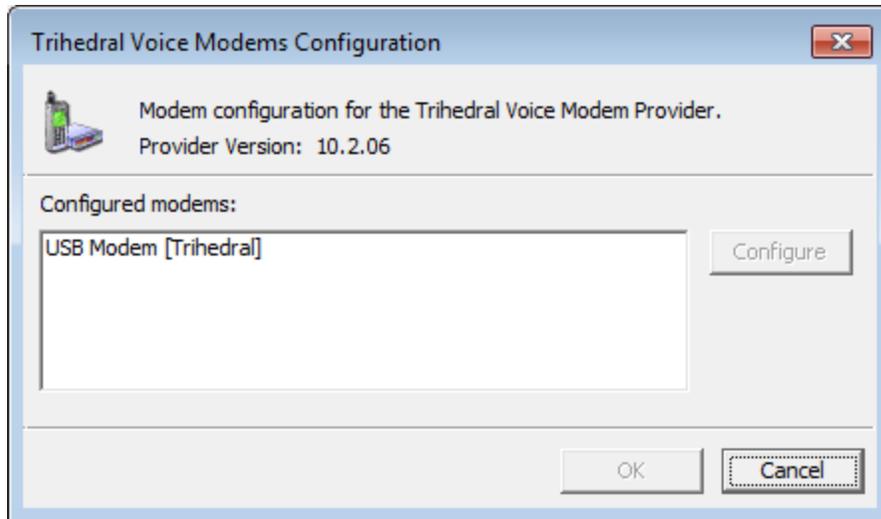
8. Close Phone and Modem Options.

Step 5: Verify correct installation of the voice modem wave bus enumerator

Verify your VTScada voice modem wave bus enumerator to ensure the voice feature of the modem will work correctly with the VTScada Alarm Notification System. This is done as follows:

1. Access your Windows Control Panel.

2. Open, System.

3. Click, Device Manager.

4. Expand the entry for Sound, Video and Game Controllers.

5. Verify that there is an entry for Voice Modem Wave Device.

6. Right-click on this device and click, Properties.

7. Verify there is a System device named Voice Modem Wave Device as shown.

8. Verify that the driver version matches your VTScada version.
   If not, your VTScada modem drivers have not been installed correctly.
   Contact Trihedral Technical Support (1-800-463-2783) for assistance.
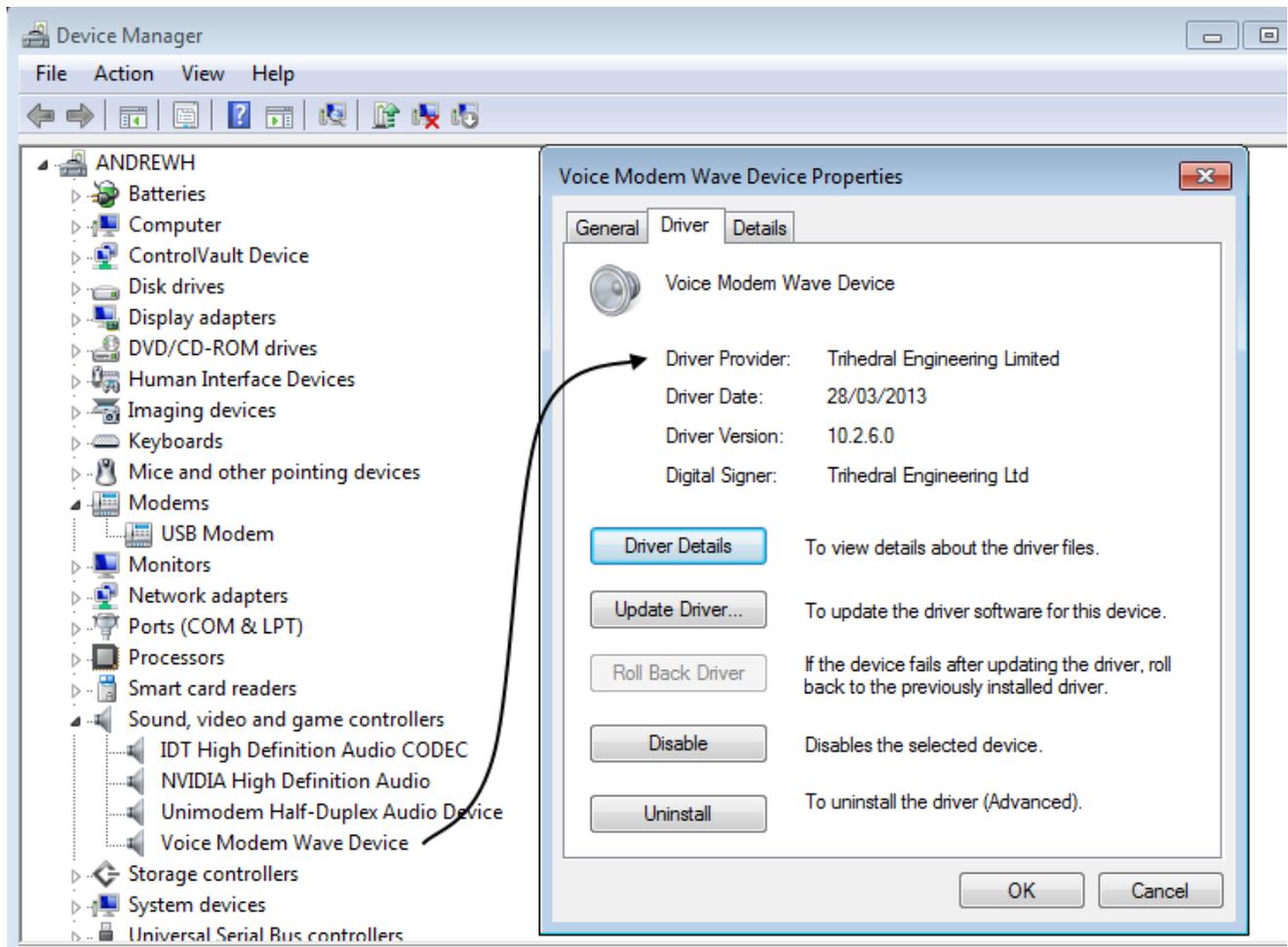
9. Close Device Manager

Installing the Trihedral TSP and Modem Audio Driver

The Trihedral drivers will be stored on your system when VTScada is installed. The TSP driver installs silently on all versions of the Windows® operating system, but you may be required to confirm that you want to install the modem audio driver.
A set of dialogs, similar to the following will be shown. You should select "Install" for both.
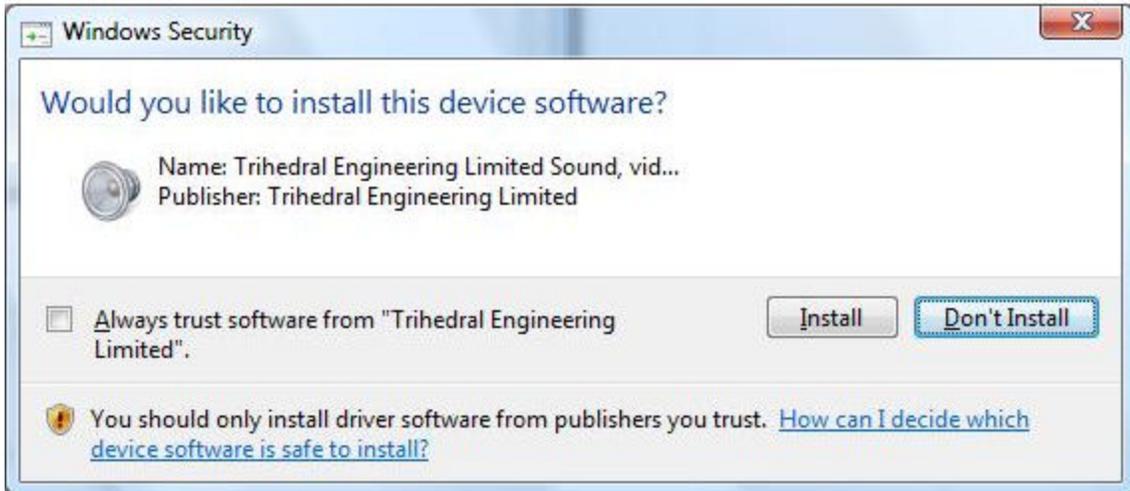
To verify that the modem audio driver has been installed, you may view the Windows Device Manager dialog. Each modem has one audio driver installed, shown as a "Voice Modem Wave Device". The following image shows how this would appear in Windows Vista where three modems have been installed:
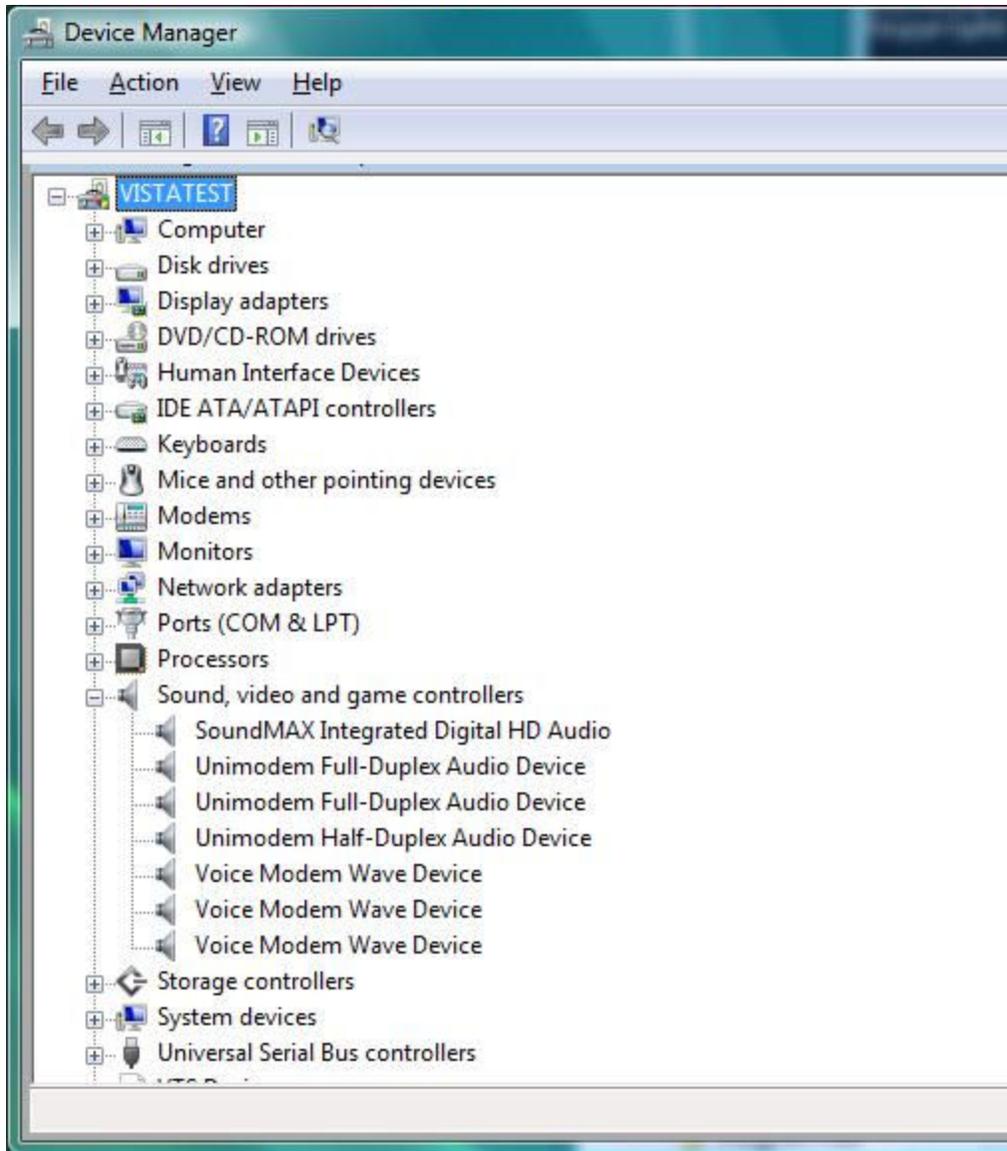
Related Information:

...Verify That Your Modem has Voice Support – The Alarm Notification System will work only with modems that have voice support.

Configure Modem Logging

The Trihedral® Voice Modem Service Provider can be used to log modem traffic in one of two ways:

- Standard logging, which is designed to mimic the Unimodem log file format
- Advanced Logging, which provides control over where the file is stored, and what information is logged. The files created by this option are stored in .csv

format, meaning that they can be opened with Microsoft Excel, then filtered and sorted.

To enable modem logging:

1. From the control panel of the Windows® operating system, open Phone and Modem options.
2. Open the Advanced tab.
3. Select the Trihedral Voice Modem Service Provider
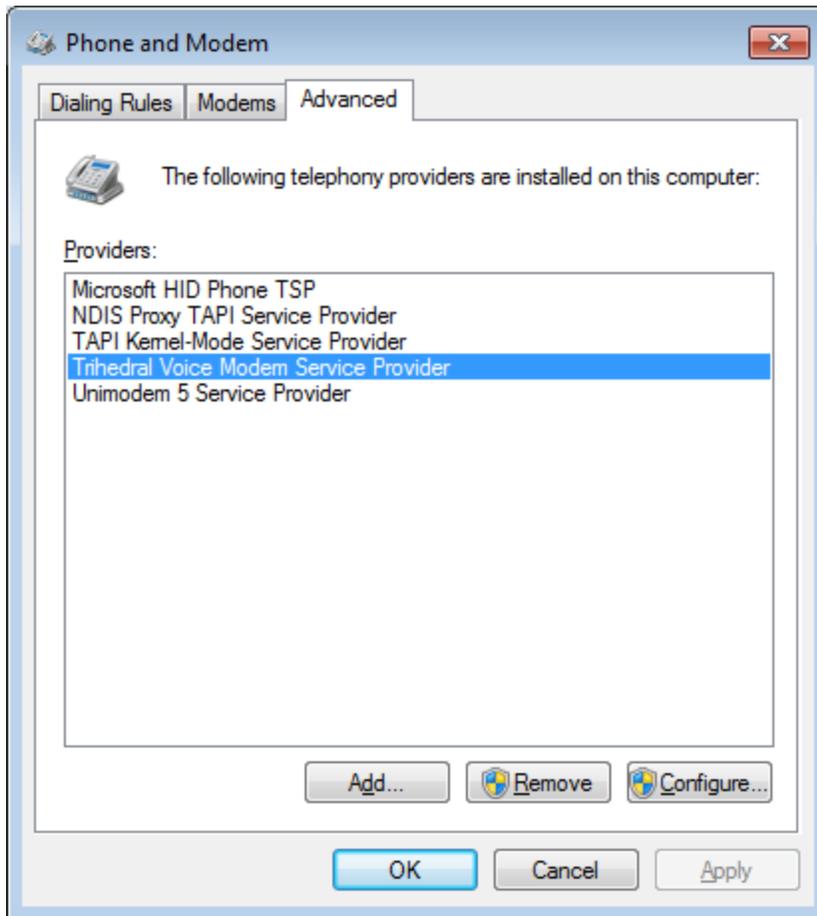
4. Click, "Configure".
5. Select your modem.
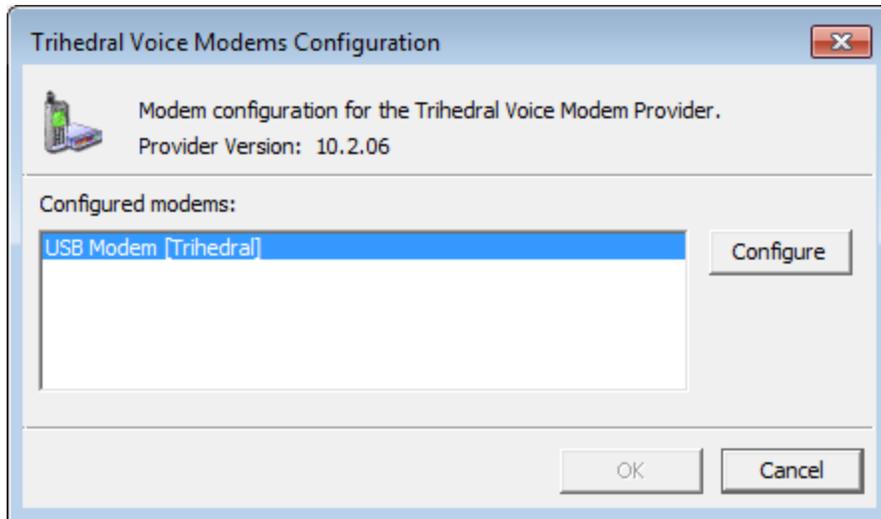
6. Click on its Configure button.
   The Modem Properties dialog will open.

7. Open the Logging tab.

8. Check the "Enable" box for either standard or advanced logging. See the following notes to help you decide which.
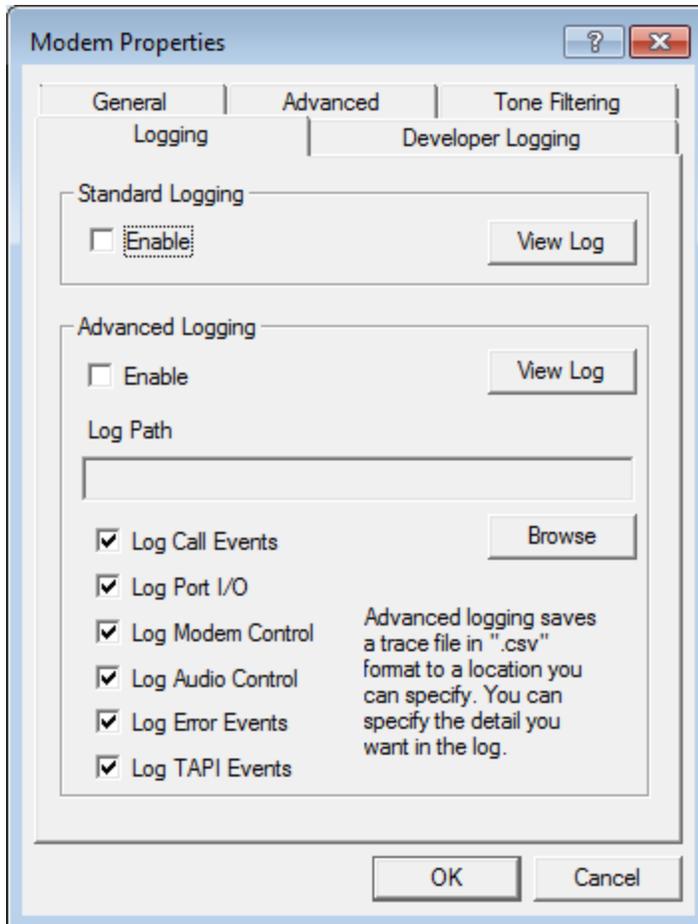
If selecting Advanced Logging, take care to select only the options that are necessary. The Audio Control, in particular, will result in the creation of very large log files. Instructions follow. It is expected that in most cases, standard logging will prove to be sufficient.

Note: Warning: Options in the Developer Logging tab should not be selected. These are intended for use only for short time periods and only when working with Trihedral Technical Support to resolve modem problems. While the Developer Logging options are enabled, overall performance of communications through the modem will be impaired.

For Advanced Logging:

1. Choose the folder path where the logs will be stored by clicking the "Browse" button.
   The folder selection dialog opens. The path chosen here will be displayed in the Log Path field.

2. Make the Log Path accessible to the "NETWORK SERVICE" user, which is a virtual user account used by the operating system to run the Trihedral Voice Modem Service. This is done using the tools provided by Microsoft Windows as follows:

    a. Navigate to the selected folder using Windows Explorer and right-click on the folder, selecting Properties from the droplist that opens.

    b. In Properties, click the Security tab.

    c. In the Security tab, click Edit.
        The Permissions dialog opens.

    d. Click, Add...
        The user selection dialog opens. This dialog has a single data entry field.

    e. Type "NETWORK SERVICE" (all caps, include the space but not the quotes) into the data entry field, then click OK.
        The User Selection dialog closes. The Permissions dialog remains open.

    f. Ensure that "NETWORK SERVICE" is selected in the box in the top half of the Permissions dialog.
        At the bottom of the Permissions dialog is a list of permissions including Full control, Modify, etc.

    g. Ensure that there are no selected check boxes in the Deny column and that the check boxes in the Allow column are selected for at least the following permissions:
        - List folder contents
        - Read
        - Write

    h. Click OK on the Permissions dialog and the Properties dialog.

Remove a Voice Modem

There are various reasons why a modem might need to be replaced. When replacement is necessary, it is critical that the correct modem and driver removal is followed before a new modem can be installed. Follow this procedure precisely to ensure proper modem and driver removal.

Step 1: Remove the manufacturer's modem driver

1. Access the control panel of the Windows® operating system.

2. Open Phone & Modem Options.

3. Open the Modems tab.

4. Select the Modem to be removed.

5. Click on the Remove button.
   The modem should now be removed from the list.

6. Close Phone & Modem Options.
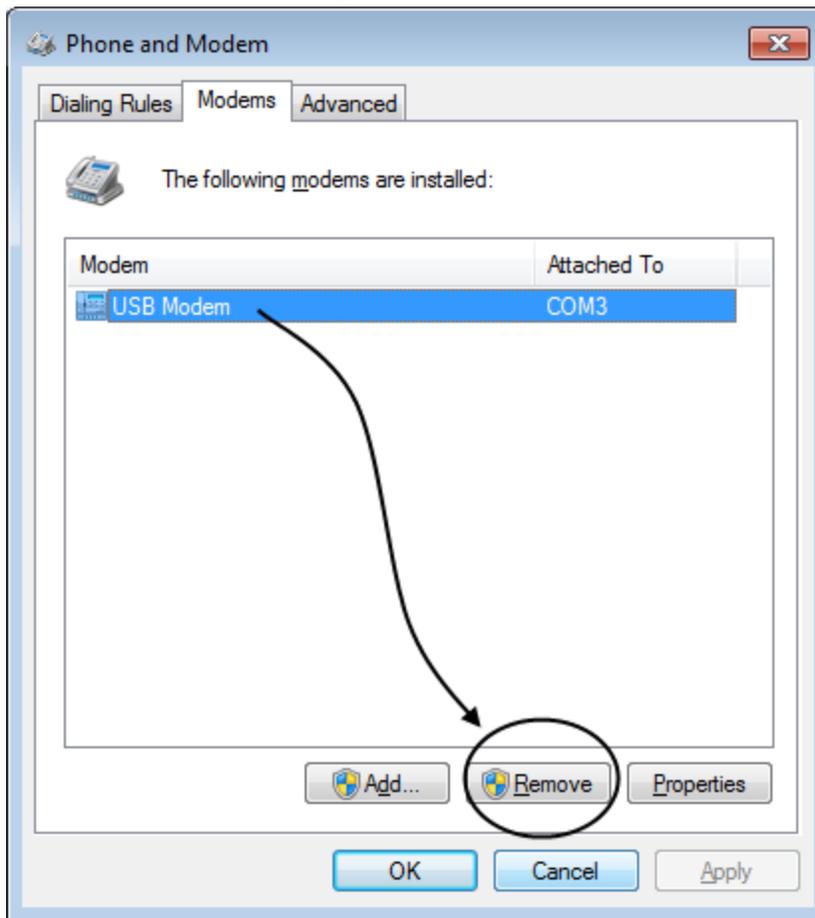
Step 2: Remove the physical modem from the computer.

1. Shut off your computer and physically remove the modem from the computer.

2. Restart your computer.

Step 3: Verify that the voice modem driver entries in the registry have been removed.

Note: Manually making changes to the Windows Registry may cause your Windows Operating System to become unstable and should only be attempted by advanced users. If you are not familiar with accessing and working within the Windows registry, contact Trihedral® Technical Support (1-800-463-2783) for assistance.

1. Create a system restore point
   Refer to your Windows Help system for guidance on creating and using system restore points.

2. Open your registry editor.

3. Navigate to the registry folder shown here. (Refer also to following image.)

```
Computer
  HKEY_LOCAL_MACHINE
    SYSTEM
      CurrentControlSet
        Control
          Class
            {4D36E96D-E325-11CE-BFC1-08002BE10318}
```

Any sub-folders of this that are numbered 1000 or higher, refer to the VTScada modem TSP. As such, ONLY delete folders numbered 1000 or higher. All keys within these folders will be deleted when the folders are removed.

Caution: If there are any existing folders with labels less than 1000, contact Trihedral Technical Support (1-800-463-2783) for assistance.
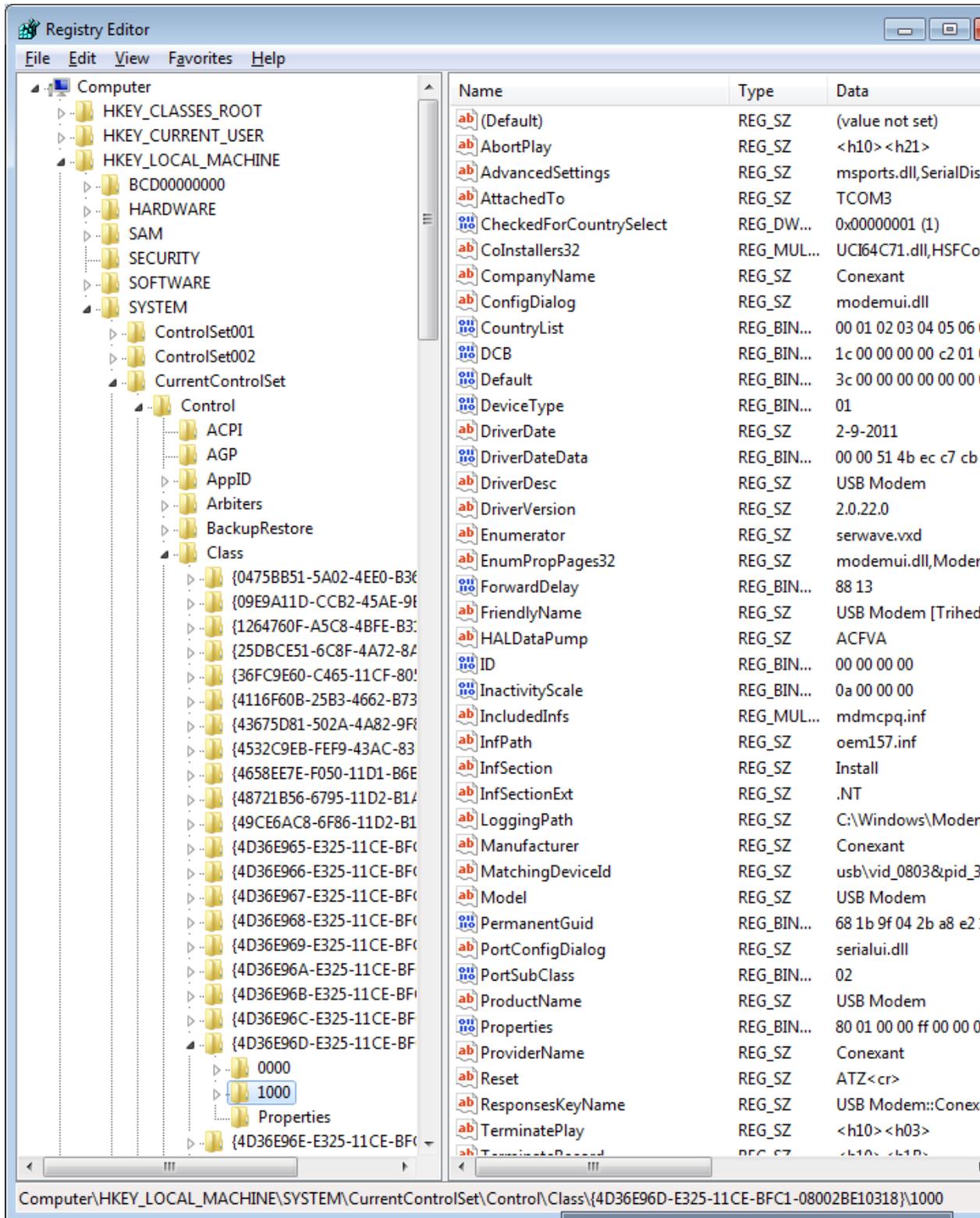
The removal of the modem and its associated drivers should now be complete. You may install a new modem safely.

Modem Manager Operation

The Modem Manager runs as a server on one system and as a client on others. The server and client Modem Managers cooperate to provide the modem pool management and to set up data paths between modem users and the allocated modem.

Incoming and outgoing calls are referred to as "tasks". When an outgoing call (or task) is requested by a workstation on the network, the Modem Manager will determine if a modem is free, select an unused modem, and dial the call. If the Modem Manager determines that all modems are in use, the call will be queued until a modem becomes free.

The queue operates on a time priority basis. If an outgoing call is attempted and fails, the call is not sent to the tail of the queue, but is slipped into the queue according to the parameters configured within its configuration file settings. For example, if a call is tried and fails, and has been configured to be redialed in one minute, then the call will be added to the queue as needed.

The modem selected by the Modem Manager to perform the outgoing call may be on a different workstation from the task requesting the call. The Modem Manager sets up transparent data paths between the modem and the requester. In the event that the Modem Manager fails to complete the connection, it can retry the attempt using alternative modems (if available).

When an incoming call is received, the Modem Manager passes the initial data to tasks that have previously registered an interest in incoming calls. If a task accepts the offered call, then the Modem Manager will set up transparent data paths between the modem and the recipient. If no one accepts the call, then the Modem Manager will return the call to the operating system to allow other applications the opportunity to take the call.

Related Information:

... See: "Modem Manager Properties" in the VTScada Admin Guide.

...Factors to Consider for the Configuration of Incoming Calls – Requires a custom-coded driver.

...Factors to Consider for the Configuration of Outgoing Calls – System settings to control call handling.

...Allocating Modems in a Central Managed Pool for Outgoing Calls – Use of the Area property in modem tags.

...Selecting Which Modem to use in a Network Application – Cycle through modems to ensure that all are working.

Factors to Consider for the Configuration of Incoming Calls

Incoming calls are handled by a discriminator, which will control where to send the call. In most applications, the only audio discriminator is the Alarm Notification System, which appears to the Modem Manager as a driver.
If you wish to create your own audio discriminator, or if you need to manage incoming data calls from any device, write a custom driver using the Modem Manager programming interface.
Consider the following factors for the configuration of incoming calls that are to be handled by the Modem Manager.

- Will incoming calls be answered?
- Set up modems (on a per modem basis) to answer on a specific number of rings.
- Decide how many modems should be kept free to accept incoming calls.
- Specify noise filtering conditions to ensure that good connections are detected correctly.
- Register the driver (provide a module).
- Decide whether to hand-off unaccepted calls.
- Specify to the Modem Manager where to route the data.

Related Information:

... See: "Modem Manager Properties" in the VTScada Admin Guide.

...Modem Manager Service – Reference and API for VTScada programmers.

Factors to Consider for the Configuration of Outgoing Calls

Restrictions of call retries are handled using the CallLength, CycleDelay, and CallInterval1 through CallInterval10 application properties. Local telecommunications authorities may have regulations regarding the frequency at which call attempts are made to a particular number. The CallLength, CycleDelay, and CallInterval1 through CallInterval10 parameters allow a redial cycle to be defined. CycleLength defines the number of steps in the cycle (with a maximum of 10), CallInterval1 through to CallInterval10 define the delay (in seconds) at each step transition, and CycleDelay defines the final delay (in seconds) before the cycle restarts.

The DialResetTime application property enables you to configure the minimum amount of time in seconds between a hang-up and the next use of the modem. The default value is 10 seconds. During this time, TAPI sends initialization strings to reset the modem. If not granted a long-enough time interval, the modem will not reset properly.

> Note: if your system includes more than one modem, and those modems are attached to servers located in regions with different area codes, then use the canonical address format[1] for each phone number and configure the server's location within the Windows™ Phone and Modem Settings dialog.

Related Information:

... See: "Modem Manager Properties" in the VTScada Admin Guide.

Allocating Modems in a Central Managed Pool for Outgoing Calls

There are situations that require a caller to indicate that only modems with a specific property should be used when making a call, even though those modems are part of the central managed pool. For example, where

---

[1]A canonical phone address is a text string with the following structure: + CountryCode SPACE [(AreaCode) SPACE] SubscriberNumber | Subaddress ^ Name CRLF …

virtual modems have been set up with special initialization strings, such as for cellular connection.

The Area parameter for Modem tags can be used for this purpose. If a Modem tag has an Area defined, then it will only be selected for an outgoing call if the call originator has specified the same area name. Additionally, when allocating local modems on a particular system, the allocator will take account of any Area name associated with the Modem tag, and use it to map the physical modem. The actual capabilities of a modem (Data, Voice, Fax, etc.) are handled by matching the physical media mode of the modem.

Related Information:

...Modem Tags

Selecting Which Modem to use in a Network Application

If you have a networked application with modems available on more than one server, the standard VTScada behavior prior to version 10 was to use only the modem on the primary modem server, so long as it is available. Should the modem on the primary server not be available, another modem will be selected for use based on which was least recently used. There is a potential drawback to this system in that the modems on backup servers are not used (and therefore not verified to be working) until they are needed.

This modem selection behavior is controlled by the application property, MMCycleTime. When not set to any value, modems are selected as just described.

When MMCycleTime is set to zero (0), then for each call, VTScada will select the least recently used of all modems. This ensures that modems are used equally.

You may retain the original behavior, but if so, you should ensure that other modems are used and thereby tested on a regular basis by setting MMCycleTime to a numeric value greater than 0. For example, if MMCycleTime is set to 86400 (the default value, which is the number of seconds in a day) then any modem that has not been used for that length

of time will be selected for the next alarm. If more than one modem has not been used in a day, then the least recently used will be selected. This system enables you continue using the modem on the primary server while it is available, but once a day (depending on the need for alarms to be dialed out) modems on backup servers will be used.

> Note: if your system includes more than one modem, and those modems are attached to servers located in regions with different area codes, then use the **canonical address format**[1] for each phone number and configure the server's location within the Windows™ Phone and Modem Settings dialog.

Related Information:

... See: "Modem Manager Properties" in the VTScada Admin Guide.

Modem Status and Statistics

The Modem Tools folder of the WSidgets palette has tools that enable you to configure displays related to the modems in your application. A common practice is to create a page showing the tools from the Modem folder, plus a Modem Indicator widget for each Modem tag in your system.
Taken together, these tools allow you to monitor current and recent modem activity, read modem log files and view collected statistics. Modem Indicators should be drawn from each Modem tag instance in the Tag Browser.

Related Information:

...Modem Indicator Widget – Shows current call activity for a specific modem tag.

...Event Log List – View information saved with the Save Log button.

...Save Log Button – Save information for the Event Log list.

---

[1]A canonical phone address is a text string with the following structure:
+ CountryCode SPACE [(AreaCode) SPACE] SubscriberNumber | Subaddress ^ Name CRLF …

...Modem Statistics – Call statistics for all modems.

...Modem Status – Extensive information for all modems, displayed in tabular format.

Event Log List

With the Event Log tool you can draw a list that displays the last 256 entries from the Modem Manager's internal log buffer, where the most recent entries will be at the bottom of the list.

> Note: VTScada also makes it possible to save the data being displayed in the Event Log tool's list to a text file by using the Save Log Button tool.

An example of the Event Log dialog. The only user-configurable option is the title that will be displayed at the top of the log.
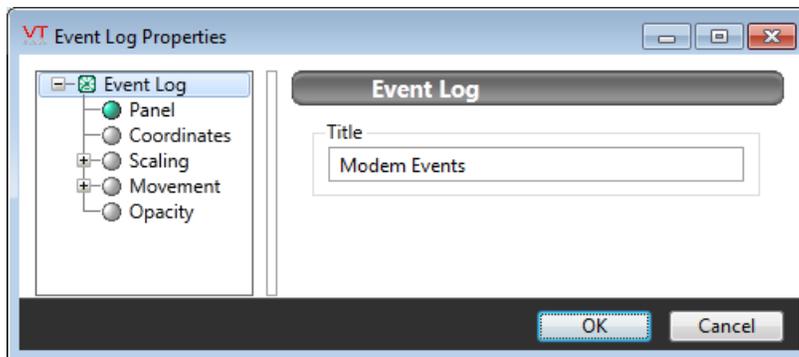
Modem Event	Description
Modem Manager Started	The Modem Manager has successfully been started.
Server	The server is physically connected to the modem.
Line Name: Phone Line Started	The Modem Manager server has obtained a connection to a modem.
Modem Name: Modem Started on Workstation Name	The identified modem on the identified workstation has been successfully initialized.
Modem Name: Answering on Ring X	The modem has successfully answered the incoming call on the ring indicated.
Workstation Name: Queue	The amount of time in seconds prior to the outgoing call

Call to Serial Port Name: Phone Number in X Seconds	being made. If the number of seconds displayed is 0, then there will be no delay in the call-out time. In some cases, the Modem Manager may have queued other calls ahead of this outgoing call. In such cases, the number of seconds will reflect the amount of time before a call out can be made.
Modem Name: Dialing to Serial Port Name on Phone Number	The modem is attempting to connect to a serial port using a configured phone number.
Modem Name: Connected To: Serial Port Name	The modem has successfully connected to the serial port indicated.
Modem Name: Call Ended To: Workstation Name: On Phone Number	Indicates that having successfully connected to the serial port, the modem can now end the communication.
Modem Name: Ending Call	The modem has completed the call and is now idle.

The number of entries that you wish to have stored in the Modem Manager's internal log buffer, as well as the format for each entry can be customized using application properties.

Related Information:

...Save Log Button – Save information to be viewed in this list.

... See: "Modem Manager Properties" in the VTScada Admin Guide.

Save Log Button

Create a button that the operator can click in order to save the Modem Manager's internal log buffer to a file for later analysis. This file may be viewed with the Event Log List.
The properties dialog for the Save Log Button dialog:
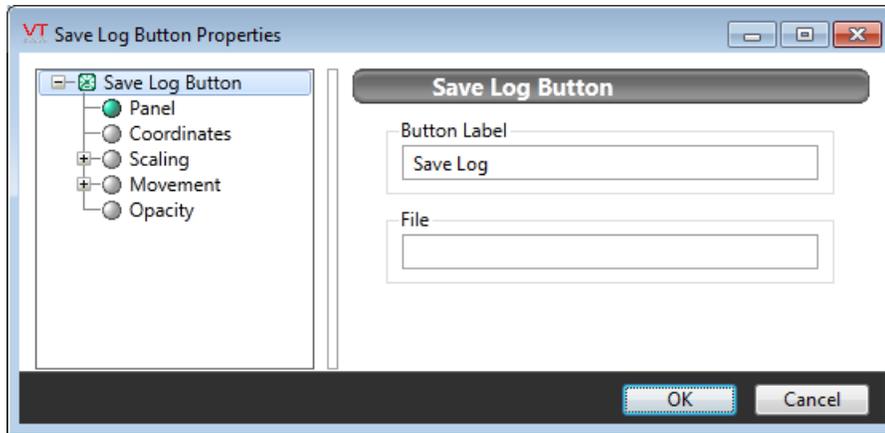
You can create your own custom label for the button with the Button Label field. If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear. The file that the data will be saved to is set with the File field. The default name of "modem.log" should be used in most cases.

VTScada will save the modem log file in your application folder, but you may save it to a sub-folder by specifying the folder name. For example: "\Modem Log Files\Modem.log".

Note: VTScada will not create sub-folders automatically. If you specify a sub-folder as the file location, it is up to you to create that folder before running your application.

Related Information:

...Event Log List – View information saved with the Save Log button.

Modem Statistics

Use the Statistics tool in the Modem Tools folder of the Widgets palette to draw a list of modem call statistics for analysis.

This differs from the Event Log List, which is used to view the modem entries in the Modem Manager's internal log buffer.

Modem Manager Statistic	Description
Lines Available	Identifies the total number of available modems.
Lines Used	Identifies the total number of modems that are in use.
Total Outgoing Calls	Displays the total number of outgoing calls made since the application was started.
Failed Outgoing Calls	Displays the total number of outgoing calls that have failed since the application was started.
Total Call Delay	Displays the amount of time in seconds from the request for the modem to the start of communications.

An example of the Statistics properties dialog. There are no user-configurable fields.
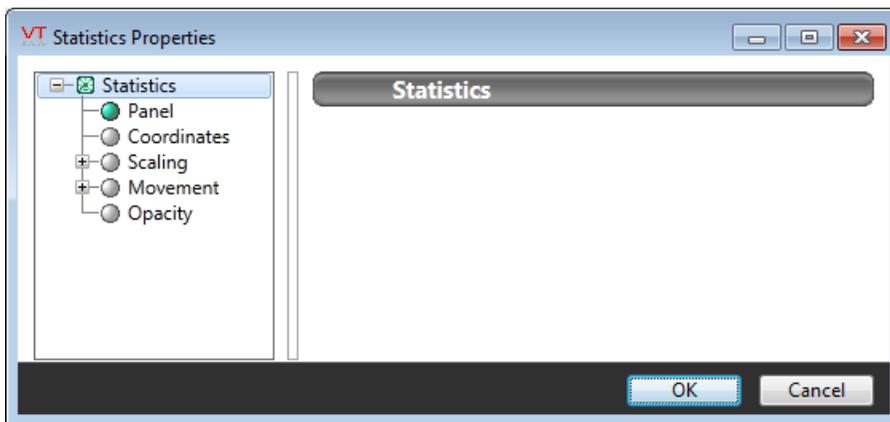
Note that some of the statistics collected will depend upon the settings configured for the Modem Manager in the application.

Related Information:

...Event Log List- Display the last 256 entries from the Modem Manager's internal log buffer.

... See: "Modem Manager Properties" in the VTScada Admin Guide.

Modem Status

Use the Status tool in the Modem Tools folder of the Widgets palette to draw a list of modem status information for monitoring. The information that is available for each modem can be seen in the properties dialog for the tool. Use the buttons in the center column to select the status monitoring values that you wish to display.
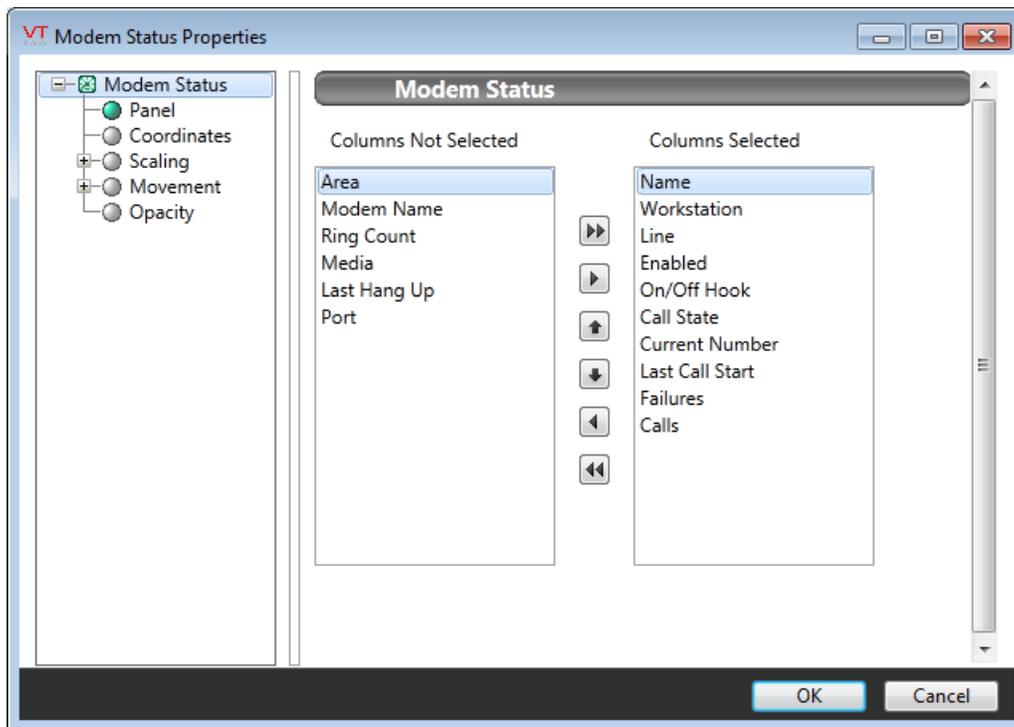
The status display is initially 1064 pixels wide. Once placed on a page, you may adjust this as required for the best display of the selected columns. In this example, all columns have been selected for display. The image has been broken onto two lines, the better to fit this space.

Modem Status								
Name	Workstation	Line	Enabled	On/Off Hook	Call State	Current Number	Last Call Start	Failures
VoiceLink	ANDREWH	HARVIE	Yes	ON	Idle		Thu Apr 18, 2013 13:22:14.146	0

Calls	Area	Modem Name	Ring Count	Media	Last Hang Up	Port
2		USB Modem [Trihedral]	0	Voice + Data	Thu Apr 18, 2013 13:23:58.882	COM3

Designate Modem Manager Server(s) for a Remote Application

As with many of VTScada's services, it is possible to designate a workstation other than the primary configuration server to manage the Modem Manager Service.

In a remote application, the Modem Manager is configured to run on one workstation that has been designated as its server. This server can be a workstation other than the primary configuration server as described in the topic, Working with Remote Applications. It is prudent to spread services out over several different workstations to distribute the load on a larger application.

The Modem Manager runs as a server on one workstation and as a client on the other workstations in the system.

It is recommended that at least two servers be arranged: one as the primary Modem Manager server; the other as a backup server in the case that the primary Modem Manager server fails. There should be a one-to-one correspondence between modems and servers. Using a network device such that the same modem is shared between two servers may have unintended side effects.

Related Tasks:

OPC Configuration in VTScada

OPC[1] is a communication interface based on DCOM[2] and web services[3] technology, which enables automation and control applications running on your PC to communicate with control equipment. In short, the OPC specification defines a set of interfaces for easy-to-use objects, and includes the functions and properties to enable the manipulation of these objects.

The basic transport layer for OPC is DCOM (Distributed Component Object Model). Because it is distributed, a human-machine interface (HMI) or supervisory control and data acquisition (SCADA) software package like VTScada can process and collect data from OPC Servers that are running on different computers on a network.

VTScada provides both an OPC Client driver and an OPC Server as tag types. Access to the server tag depends on whether your VTScada license key includes this optional feature. Please see your Trihedral account representative for details.

OPC Client

The OPC Client driver enables VTScada to act as a client to an OPC server, enabling the exchange of data between a VTScada application and the server. The advantage to using OPC is that it enables VTScada to communicate with new control devices without requiring a custom-written driver. The OPC server takes care of communications with the control

---

[1]Open Platform Communications. A communication interface based on DCOM and Web Services technology.

[2]Distributed Component Object Model

[3]An application programming interface (API) that can be accessed using HTML.

device, and enables VTScada to access its data via an OPC automation DLL.

> Note: If you install third-party OPC software in some cases the VTScada OPC Client or Server may stop functioning correctly. Sometimes this can be solved by re-installing the OPC Core Components. Contact technical support for assistance.

OPC Server

The OPC Server (OPC Driver) enables VTScada to act as a server to an OPC client. This enables other programs (including another VTScada application with an OPC client driver) to exchange data with VTScada.

An OPC Server enables clients to retrieve information about objects.

There are 3 basic levels of objects:

- The OPC Server,
- The Group, and
- The Items.

The OPC Server object maintains information about the server, and acts as a container for OPC group objects. It handles data requests from clients.

The OPC Group object is a collection container for OPC Items. The OPC Group maintains information about itself and provides the mechanism both for containing and logically organizing OPC items.

The OPC items represent connections to data sources within the OPC Server. In the case where VTScada is providing the server, these items are tags. Each item provides the client with data and properties. Data presented to the client by each item is specified to contain one value, one time stamp and a set of quality flags. The data will be either numeric or text. The OPC server object enables clients to browse within the server namespace to find what items are available to add to groups. The namespace is organized as a hierarchy and contains not only tag value identifiers, but property value identifiers that may be associated with a tag.

Related Information:

...Compatibility with OPC Tunneller Software – Avoiding problems.

... OPC Addressing for VTScada – Reference.

...Properties of Tag OPC Items – Table of standard item properties, available to VTScada OPC clients.

...Tags that support OPC – Reference

...Item Value/Quality/Timestamp – Three values, updated whenever the OPC tag's value changes.

...Securing an OPC Server – General concepts and tasks.

...OPC Client Driver I/O Addressing – Includes an example of the namespace structure and how it is possible to browse through it.

...Trace Viewer Application – You can monitor OPC communications using the Trace Viewer. Note that communications will be slowed while tracing is enabled. Instructions and reference provided in the VTScada Programmer's Guide

Related Tasks:

...Configure VTScada as an OPC Server

...Configure VTScada as an OPC Client

...Register or Unregister a VTScada OPC Server

Securing an OPC Server

Security for OPC servers is provided by the Microsoft Windows™ DCOM security tools. For a comprehensive discussion of how to secure an OPC server, please refer to the OPC Foundation's website: http://www.opcfoundation.org
Additionally, the Software Toolbox® (not affiliated with Trihedral Engineering Ltd.) provides very good reference information and tutorials for DCOM security configuration. These materials can be found at:
http://support.softwaretoolbox.com/app/answers/detail/a_id/1554
In general terms, the following steps are required to secure the server:

A. Run the program, DCOMCNFG.exe on the workstation to which users will be connecting.

In Windows™ XP, this is also available through the menu: Control Panel >> Administrative Tools >> Component Services. Under Windows Vista, you will need to run the program directly since as there is no link to it from the Control Panel.

B. Find the entry "Trihedral VTScada OPC Server" as shown in the following image:
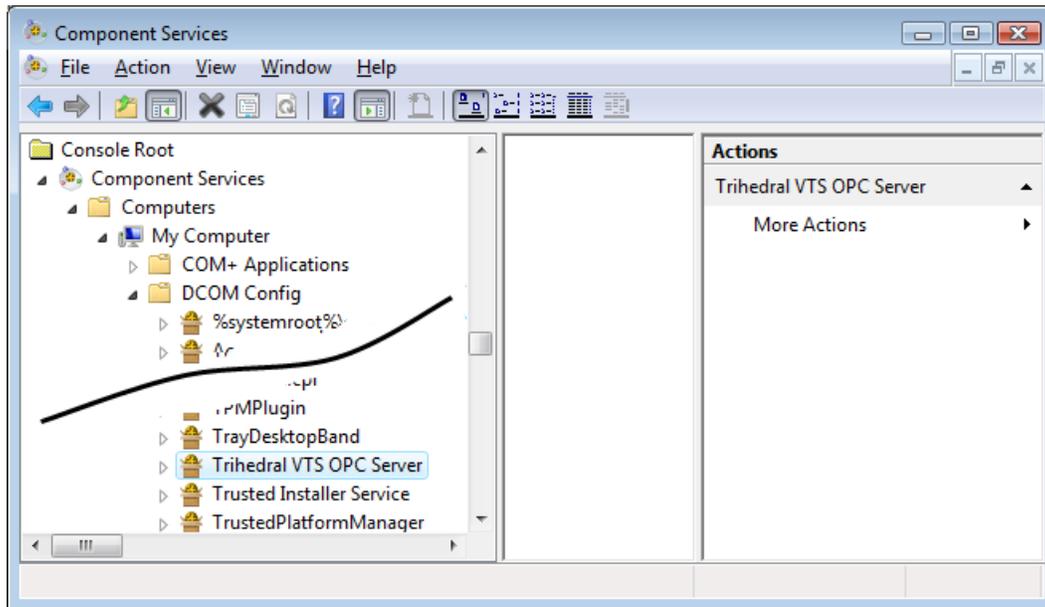
C. Right-click on the Trihedral VTScada OPC Server entry and select Properties. The following dialog should open (shown with the Security tab selected):
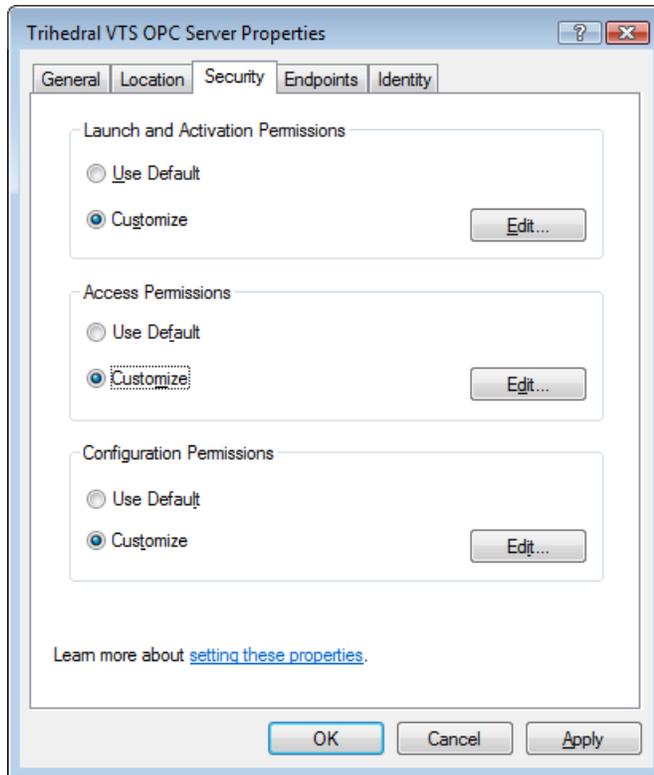
Use this to select the groups and users who will have permissions for each of the three categories.

Note: It is very important to deny Launch and Activation privileges to all users. If a client application attempts to launch VTScada, it will do so from its own directory, not the application folder. This will not work.

Notes for Windows Workgroups

If your computers are networked in a workgroup there are additional security settings that may be required.

The default installation forces remote users to authenticate as Guest. This means that DCOM clients cannot connect to a server unless the Guest account is enabled and has enough rights to launch the server.

You can adjust these settings from the Control Panel:

A. Open the Control Panel.
B. Click to open Administrative Tools.
C. Open the Local Security Policy dialog.

D. Click, "Local Policies".

E. Click, "Security Options".

F. Click, "Network access: Sharing and Security model for local accounts".
Change this setting to the following:

"Classic – users authenticate as themselves".
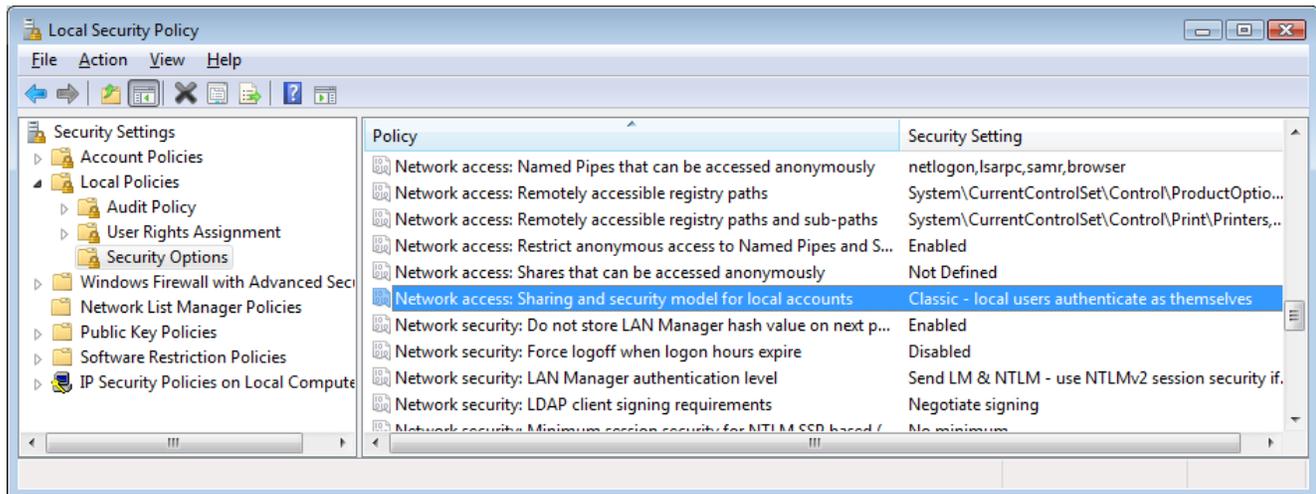
Compatibility with OPC Tunneller Software

Certain third-party OPC tunneller software may attempt to launch a non-interactive VTS.exe as the OPC server instead of connecting to the already-running VTS.exe. The following actions can help you avoid problems:

- Configure the VTScada OPC Server (in DCOMCNFG) to disallow "Launch for all users" and to run VTScada as the interactive user.

- After installing the third-party OPC tunneller software, install the latest OPC Core Components from OPCFoundation.org.

- All configuration tools for the tunneller software should be run as Administrator.

- Ensure that all necessary services related to the tunneller software and OPCEnum are running.

- Ensure that the server's firewall explicitly allows access to the port that the tunneller software uses.

Configure VTScada as an OPC Client

To configure VTScada as an OPC client, you will need:

- A program that provides an OPC server to which you wish to connect
- The OPC server's program identifier.
- If the OPC server is not on the same computer, then you will need the name of the workstation where the server is located.

To configure an OPC client driver where another VTScada application is the server:

Given a VTScada application (App1) that provides an OPC Server whose server namespace is BedfordSim and that includes an Analog Input tag named Tank1Level, whose values you wish to read in another application. (All steps to be done in the second application)

1. Using the tag browser, add an OPC Client Driver tag to App2
2. Provide a name for the OPC client driver tag.
3. The Server Prog ID will be "Trihedral.VTSOPC.DA"
   Note: this name is case-sensitive.
4. Provide the name of the workstation that App1 is running on.
5. Click OK to finish configuring the client driver.
6. Add a new Analog Status or Analog Input tag.
7. Set the I/O device field of this tag to use the OPC client driver you added in step 1.
8. Set the address using the following template: OPC_Server_Name\tags\all\AI_tag_name.
   For this example, the address would be: BedfordSim\tags\all\Tank1Level.
9. Draw the tag.

Related Information:

...OPC Client Driver Tags – How to configure and use an OPC client tag in VTScada

Configure VTScada as an OPC Server

To configure VTScada to work as an OPC Server you will need:

- A license key that includes the OPC Server option. Contact your account representative.
- A configured OPC Server Setup tag. (steps follow)
- Security.

Steps:

1. From the tag browser, add an OPC Server Setup tag
2. Give it a namespace – the name of the application is recommended

Within VTScada, no further action is required in order for the client to read the values of tags in the current VTScada application, but you may wish to secure your application. This is done outside VTScada using Windows® operating system user and group configuration tools.

Values are updated on change. The client attached to the VTScada OPC server need not (and should not) use polling. The fastest update rate supported by the server is 20ms.

Adding an OPC Server tag to a networked application enables the OPC server on every workstation running the application. Clients must specify which workstation they are connecting to.

If your application uses an OEM layer, having its own custom tag types, you will need to add code to those tag definitions in order for the OPC client to see them.

If your goal is to share live data between separate VTScada applications, using an OPC Server and Client combination will provide the most efficient means of transferring that data.

Related Information:

...Securing an OPC Server – Your connection may not be secure without these procedures.

...Make a Custom Tag Visible to OPC Clients – Reference for VTScada programmers.

OPC Addressing for VTScada

The name of OPC server program id is "Trihedral.VTSOPC.DA". All clients connecting to a VTScada OPC server will need to connect to this Prog ID.

Each VTScada application must have its own namespace. This is defined in the OPC Server Setup tag. The naming hierarchy can be visualized with the help of the following diagram:
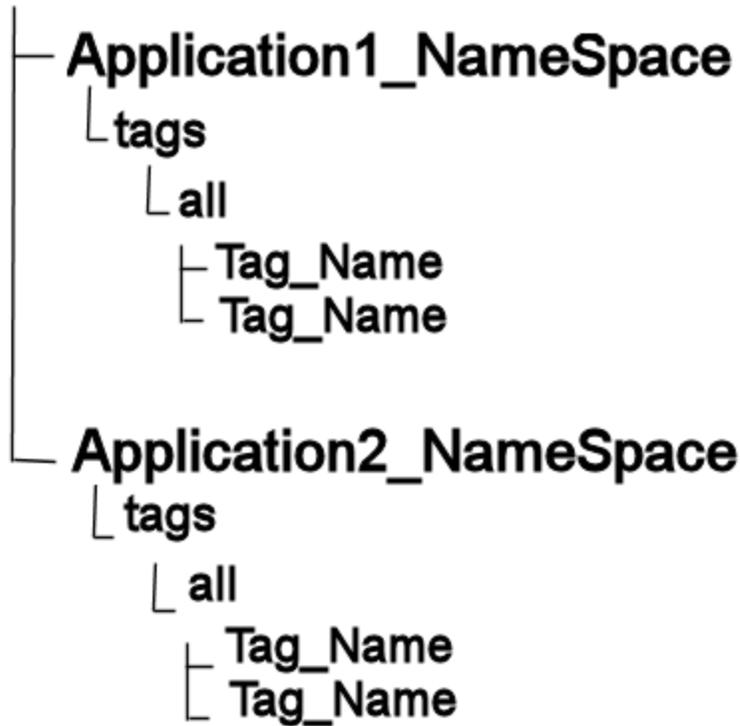
To access a tag's value, use the syntax:

```
NameSpace_name\tags\all\TagName
```

Related Information:

...Properties of Tag OPC Items

Properties of Tag OPC Items

The OPC Item tags, which represent VTScada tags, may reveal the following standard properties. Items that do not have a property name are not available to the VTScada OPC client.

Property ID	Description	Property name
1	Data Type	N/A
2	Item Value	N/A

3	Item Quality	N/A
4	Item Timestamp	N/A
5	Item access rights (e.g. R/W)	N/A
6	Server scan rate (ms)	N/A
100	EU Units	EUUnits
101	Description	ItemDescription
102	High EU	HighEU
103	Low EU	LowEU
104	High Instrument Range	HighInstrumentRange
105	Low Instrument Range	LowInstrumentRange
106	Contact Close Label	ContactCloseLabel
107	Contact Open Label	ContactOpenLabel
108	Item Timezone	ItemTimezone

The following custom OPC properties have been added to the VTScada tag OPC Items as needed:

ID	Description	Property Name
5000	Area	Area
5001	Device Tag	DeviceTag
5002	Address	Address
5003	Raw (unscaled) Value	RawValue
5004	Low Alarm Limit	LowAlarmLimit
5005	High Alarm Limit	HighAlarmLimit

To access a particular OPC Property value, use the following syntax:

```
AppOneOPCServer\tags\all\TagName\PropertyName
```

To access a tag's value, use the syntax:

```
AppOneOPCServer\tags\all\TagName
```

Not every tag has all of the properties that are listed in the above tables. The following will help you find whether a property is available for the tag you are interested in.

Property	Available in...
ItemDescription	All tags
EUUnits	All tags with a Units variable
HighEU, LowEU, HighInstrumentRange, LowInstrumentRange	All scalable tags (AnalogStatus, AnalogInput, AnalogControl, AnalogOutput)
ContactCloseLabel, ContactOpenLabel	DigitalStatus
DeviceTag, Address	All I/O tags
RawValue	AnalogStatus and AnalogInput
LowAlarmLimit, HighAlarmLimit	AnalogStatus

Related Information:

... OPC Addressing for VTScada

Tags that support OPC

All tag instances of the following VTScada types will show up as OPC items. The writable tags make it possible for an OPC client to use an output tag to write to a PLC value.

Read:

- Calculation
- Function
- Network Status
- Pump
- Counter Tag
- Totalizer Tag

Read and write:

- Analog Input
- Analog Output
- Analog Status
- Analog Control
- Digital Input

- Digital Status
- SelectorSwitch

Write:

- Digital Output
- Digital Control

Item Value/Quality/Timestamp

An OPC Item representing a tag has three values that get updated whenever the tag value changes:

- Value
- Quality
- Timestamp

The timestamp is usually the time that the tag value changed, except in the case of input tags that retrieve timestamps along with values from I/O devices. In the latter case, it is the timestamp that corresponds to the current tag value.

VTScada OPC Server items support the following qualities:

Quality value	Name	Description
0x00	Bad	The tag's value is Invalid or the tag's Quality value indicates Bad quality.
0x04	Config Error	The tag has been deleted or the server application has been stopped.
0x40	Uncertain	The tag is marked as Questionable.
0xC0	Good	Good quality
0xD8	Good (Local Override)	The tag has Manual Data specified.

Register or Unregister a VTScada OPC Server

When VTScada is installed, the VTScada OPC Server DCOM component is automatically registered with the operating system. This is required for a

client to be able to connect to the VTScada OPC Server (for any VTScada application).

If (after unregistering), you wish to register a VTScada OPC Server, you should run the following command:

```
vts/RegOPCServer
```

If you wish to unregister VTScada as an OPC server, you should run the following command at a command prompt (Note that, in Windows Vista® and later operating systems, this must be run from an administrator-privileged command prompt):

```
vts/UnRegOPCServer
```

This is a VTS-wide setting. It does not need to be repeated for each application.

OPC Troubleshooting

There have been reports of the OPC core components being damaged during the installation of third-party OPC server. The usual symptom is slow connectivity.

The following two suggestions may help to alleviate the problem.

- If slow connectivity is experienced while running the 64-bit installation of VTScada, try reinstalling with the 32-bit option selected. In testing, this has been found to substantially reduce connection time.

- Fast response times should be available with the 64-bit version of VTScada. The problem may be caused by OPC-related registry settings and components. To restore the settings:

1. Stop VTScada

2. From an Administrator command prompt, execute the following files to remove the current installations of the OPC Core components:

```
OPC_Core_Components_Redistributable 3.00.101.0 (x64).msi
OPC_Core_Components_Redistributable 3.00.101.0 (x86).msi
```

3. From the same command prompt, use the same MSIs to reinstall the OPC Core components.

4. Restart VTScada and test your application.

Build the Display

All the user-interface pages of your application will be built using the Idea Studio. This opens in its own window, allowing you to create and edit pages while simultaneously viewing them live in the background. Your changes are applied immediately as you make them.

The Idea Studio is an extremely powerful development environment, placing many tools at your disposal.

Related Information:

...Customized Alarm Display – Add an interactive alarm list to any page.

...Build the User Interface with the Idea Studio

...Create a Page (or Widget) – Create pages for the user interface.

...Define Navigation Between Pages – Build menus or add navigational links between pages.

...Draw Widgets – Display tag values or allow operator control.

...VTScada Graphics Library, Draw Shapes, Decorations – Provide context for the display.

...Select and Arrange, Formatting Tools– Arrange and format the contents of the display.

Build the User Interface with the Idea Studio

The Idea Studio is where you will do all the work of building your application. Open it by clicking the button shown circled.

Note: Just getting started with VTScada? If you have pressed F1 in response to the message on the initial Overview page, then here's what to do next:

You can clear the welcome message from the Overview page by pressing Ctrl + A (select all) then Delete. Then you will have a blank page, on which to begin development.

Information about the development process can be found in the topic: Install VTScada and Build an Application.

When open, the Idea Studio window will always be in front of the running application. Changes made here are instantly reflected in the matching page of the running application. (Immediate deployment to other workstations on your network is optional.)

Note: The Multi Image Widgets panel may be configured to open automatically whenever a multi-image widget is selected. This may be inconvenient at times. You can disable this behavior within the Multi Image Widgets panel.
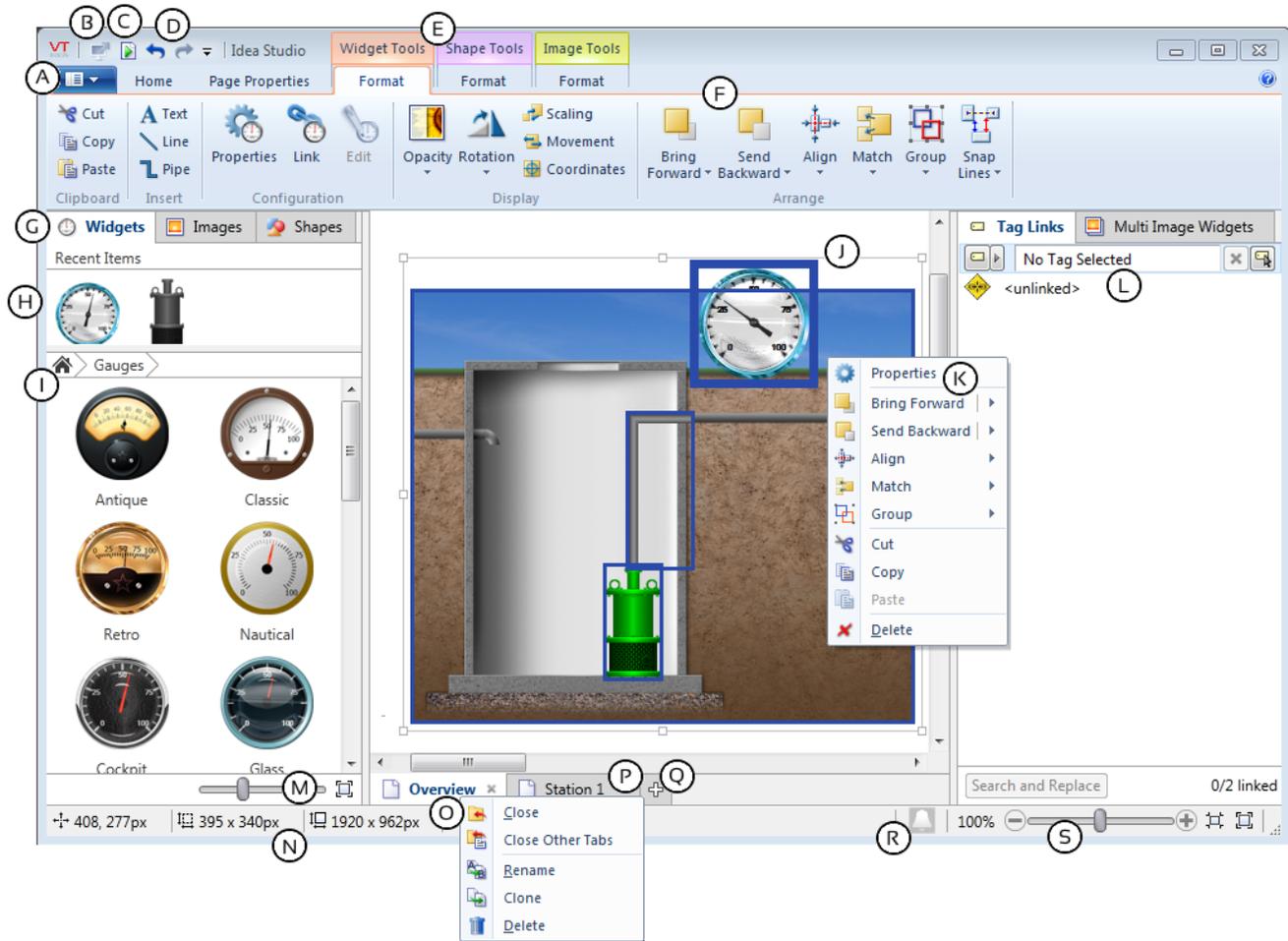
Idea Studio Components:

A	The File Menu	Open / close / create and other commands related to both pages and widgets. The File Menu
B	Deploy changes	Enabled only when automatic deployment is off. The Configuration Deployment Process
C	Operator view	Closes the Idea Studio and displays the page that was being edited in the Idea Studio, allowing you to test operator controls. (Enabled only for pages, not when widgets are being edited.) Quick Access Toolbar

D	Undo / Redo	Make editing mistakes go away. Quick Access Toolbar
E	Ribbon Selection	Property ribbons for all currently selected objects.
F	Tools within the current ribbon	Tools vary by object type. Ribbons
G	Palette selection	Switch from one palette to another as you work.
H	Recently drawn objects	Convenient for drawing more of the same object.
I	Palettes	Drag widgets, images and shapes from the palettes to the page to represent your system and provide operator controls. Palettes
J	Selected Graphics	A gray box encloses all selected objects. Faint blue box for each object within the selection and dark blue box for the "anchor" object. Select and Group
K	Object selection – right-click menu	Properties and commands relevant to the object you right-clicked on.
L	Tag Links Panel – – – – – – – – Multi Image Widgets Panel	Use the Tag Links panel to view and change the linkages between selected widgets and tags. – – – – – – – Multi Image Widgets Panel
M	Scale palette display	Show more or fewer items in the palette.
N	Coordinates	Current cursor location, size of selected objects and area of all objects.
O	Current page.	The current page or widget is highlighted. Right-click for a menu of commands. Create a Page (or Widget)

P	List of currently open pages & widgets	All open pages and widgets are listed, but only one is displayed for editing at a time.
Q	Add new pages & widgets	Create pages as necessary for sub-systems, regions, controls, and other parts of your application. Create a Page (or Widget)
R	Alarm notification	Don't miss alarms while working in the Idea Studio. See: Working with Alarms in the Operator's Guide
S	Scale editor display	Scale the overall view of what you are working on. Right-click on scroll bars for a pop-up menu of options controlling how far each click is to scroll.

Related Tasks:

Pages...

...Create a Page (or Widget)

...Open a Page or User-Defined Widget to Edit

...Formatting Tools

...Define Navigation Between Pages

...Adding New Menu Items

Widgets & Tags...

...Editing the Palette MenuItem Tags – Palette editing.

...Draw Widgets – and link them to tags.

...Style Settings Tags – Set display standards rather than configuring every widget.

...Design Your Own Widgets – Expand the feature set with widgets for your own industry.

Graphics...

...VTScada Graphics Library – Add context to the display.

...Import Images

...Draw Shapes

...Draw Text

...Snap Lines – Control tracking and alignment tools.

Related Information:

...Draw Widgets

...Tag Concepts and Features

...Sample Pages

The File Menu

The File Menu of the Idea Studio will be labeled with the word File, or with only an icon, depending on the version of the Windows® operating system that is installed on your computer.
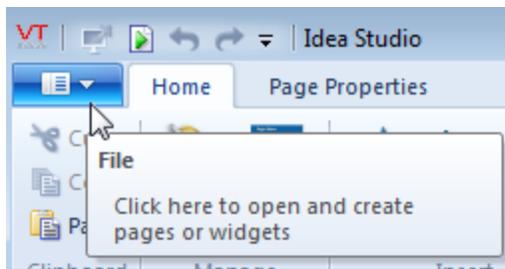
Tools in this menu allow you to:

- Create new pages and widgets.
- Open existing pages and widgets.
- Close the open page or widget.
- Clone the current page or widget, creating an identical copy.
- Import a page or widget by selecting its source code file.
- Delete a page or widget.
- Open the Application Configuration dialog
- Open the Help documents.
- Exit from the Idea Studio.

Quick Access Toolbar

The icons across the top of the Idea Studio are part of the Quick Access toolbar. You can move this to below the main ribbon if you choose.
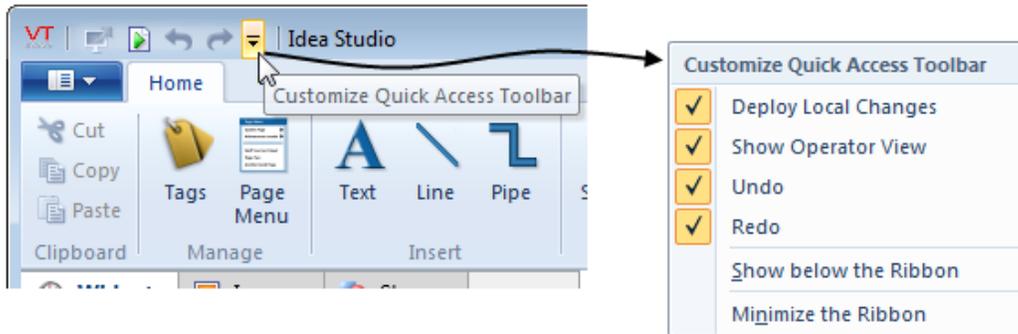
Standard tools in the Quick Access toolbar include:

- **Deploy Local Changes** – Enabled only when automatic deployment is switched off.
- **Show Operator View** – Close the Idea Studio and open the current page in the main window.
- **Undo / Redo** – Back up from design errors.
- **Customization Menu** – As shown in the image. Hide tools, change the location of the toolbar, minimize the main ribbon.

You can add any often-used ribbon tool to the Quick Access toolbar. To do so, right-click on any tool in any ribbon and select, Add to Quick Access Toolbar.
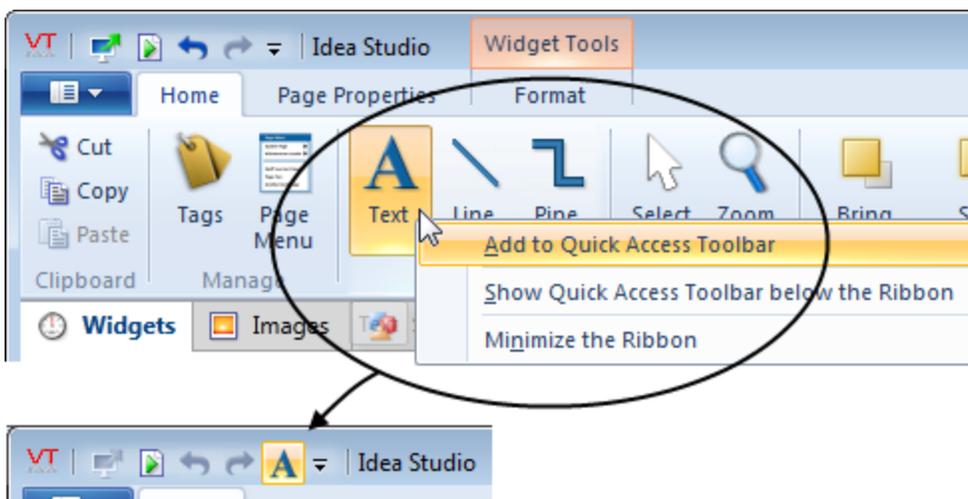
To remove a tool, right-click on its icon in the Quick Access toolbar and select "Remove from Quick Access Toolbar".

Ribbons

Many of the object creation, formatting and alignment tools, available to the Idea Studio, are stored in ribbons across the top of the page. Several ribbons exist, but only the Home and the Page Properties ribbons will be visible at all times. Other ribbons contain formatting tools and will be visible only you have selected objects of a type that the ribbon was designed for.

Home Ribbon
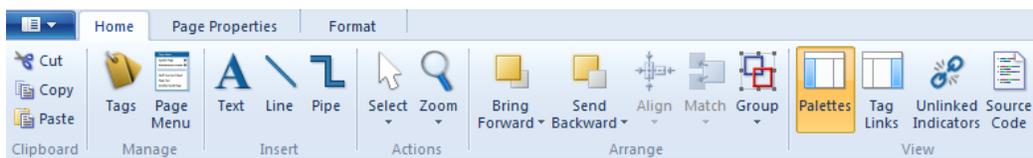
- Clipboard tools: Cut Copy and Paste are standard tools for duplicating objects.
- Manage Tools: Both Tags and Page Menu open the Tag Browser. The difference lies only in the branch of the tag tree initially selected: MenuItems or other.
- Insert: Contains the three elements most commonly used when building a user interface: Text, Line and Pipe.
- Actions: Select contains a menu of object selection tools, including the ability to select all objects of any given type.
  Zoom contains a series of preset display scales, and a link to a slider where you can adjust the display scale by any percentage.
- Arrange: Tools for changing the display order and alignment of selected objects. Also include in this section are tools for matching properties across all elements in a selection set, and to group objects together.
- View: Open or close the Palettes window. Open or close the Tag Links window. Enable or disable the unlinked indicator display, so that you can track which widgets are not yet linked to any tag.
  Open the source code file for the current page. Changes made to any page must be imported via the VAM (VTScada Application Manager) before they will be used by the application.

Page Properties Ribbon
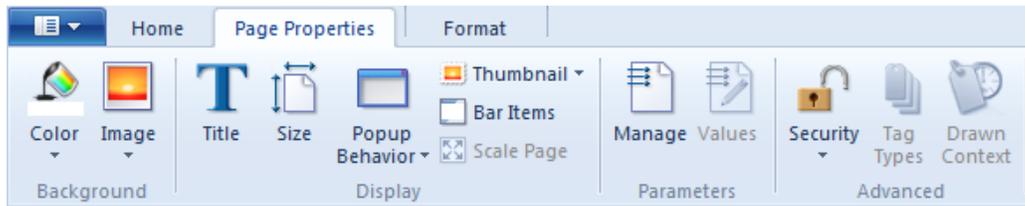
- Background: Add color or an image to the page background

- Display: Tools to adjust all visual aspects of the page besides the back-ground.

- Parameters: Create a parameterized page (or widget), then adjust the values used for the parameters.

- Advanced: Prevent unauthorized access to a page by adding an application privilege. This tool also provides access to the Security system, where you can create application privileges and grant those privileges to accounts. If editing a page, you can use Tag Types and Drawn Context to modify the types that are or can be linked to the widget.

Widget Properties Ribbon

Used when editing a user-defined widget.
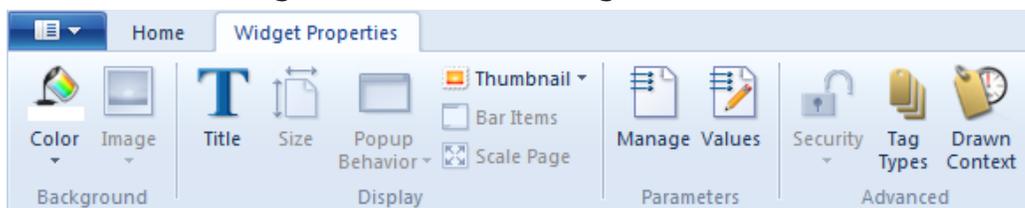
- Background: Set a background color to be shown in the Idea Studio and in the palette. Not included when the widget is drawn on a page. Background images are not available for widgets.

- Display: Change the name of the widget. Select an image that will represent the widget in the palette.

- Parameters: Add and control parameters, if any are to be associated with the widget.

- Advanced: Edit the list of tag types that can be linked to this widget. Select a particular tag to use for the instance shown in the editor.

Widget Format Ribbon

Used to format a widget as drawn on a page.
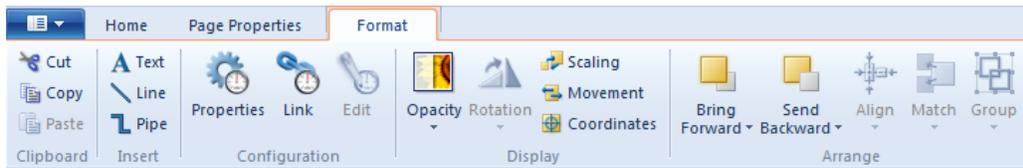
- Clipboard tools: Cut Copy and Paste are standard tools for duplicating objects.
- Insert: Contains the three elements most commonly used when building a user interface: Text, Line and Pipe.
- Configuration: Open the Properties dialog for the selected widget, or link it to a tag. Edit applies only to widgets of your own creation, opening the configuration page for that widget.
- Display: Adjust the opacity of the widget, or open the Properties dialog to one of the menu items, Scaling, Movement or Coordinates.
- Arrange: Tools for changing the display order and alignment of selected objects. Also include in this section are tools for matching properties across all elements in a selection set, and to group objects together.

Image Format Ribbon
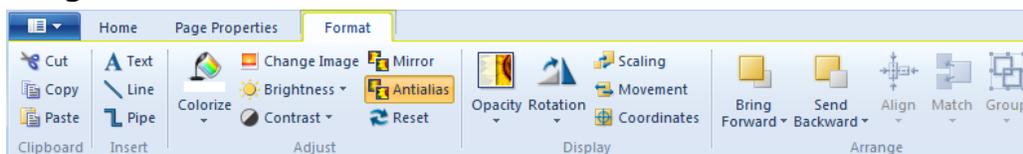
- Clipboard tools: Cut Copy and Paste are standard tools for duplicating objects.
- Insert: Contains the three elements most commonly used when building a user interface: Text, Line and Pipe.
- Adjust: Mirror and Anti-alias are toggle buttons that can be switched on or off. Other tools provide a range of control over all aspects of the image's color, and brightness.
- Display: Adjust the opacity of the widget, or open the Properties dialog to one of the menu items, Scaling, Movement or Coordinates.
- Arrange: Tools for changing the display order and alignment of selected objects. Also include in this section are tools for matching properties across all elements in a selection set, and to group objects together.

Shape Format Ribbon
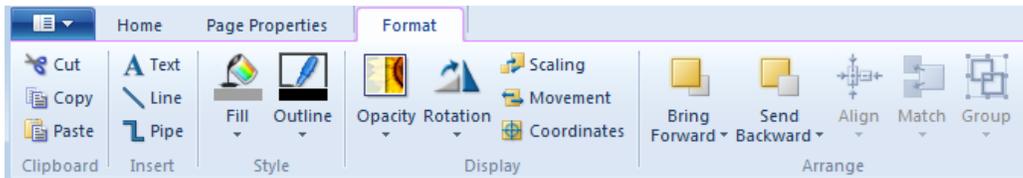
- Clipboard tools: Cut Copy and Paste are standard tools for duplicating objects.
- Insert: Contains the three elements most commonly used when building a user interface: Text, Line and Pipe.
- Style: Adjust the display properties the shape's fill or outline.
- Display: Adjust the opacity of the widget, or open the Properties dialog to one of the menu items, Scaling, Movement or Coordinates.
- Arrange: Tools for changing the display order and alignment of selected objects. Also include in this section are tools for matching properties across all elements in a selection set, and to group objects together.

Text Format Ribbon
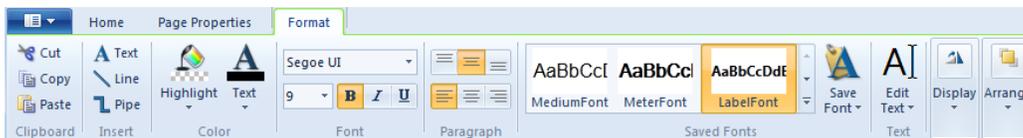
- Clipboard tools: Cut Copy and Paste are standard tools for duplicating objects.
- Insert: Contains the three elements most commonly used when building a user interface: Text, Line and Pipe.
- Color: Adjust the color and opacity of the text's foreground or background.
- Font: Select the font characteristics that should be used for the selected text.
- Paragraph: Adjust the vertical and horizontal alignment of the text within its bounding area.
- Saved Fonts: Choose an existing font (as stored in a Font tag) or save the current configuration as a new, or updated existing Font tag.
- Edit Text: Opens the edit text dialog for the selected line of text.
- Display: Adjust the opacity of the widget, or open the Properties dialog to one of the menu items, Scaling, Movement or Coordinates.

- Arrange: Tools for changing the display order and alignment of selected objects. Also include in this section are tools for matching properties across all elements in a selection set, and to group objects together.

Palettes

There are three palettes in the Idea Studio: Widgets, Images and Shapes. These hold the majority of the objects that can be added to a page. Palette organization is controlled by the Menus branch of the Tag Browser. You cannot create new palettes(*), but you are free to reorganize the contents of any of the three standard ones. (* New folders can be added at the same level as Widgets, Images and Shapes, but will be ignored by VTScada.)
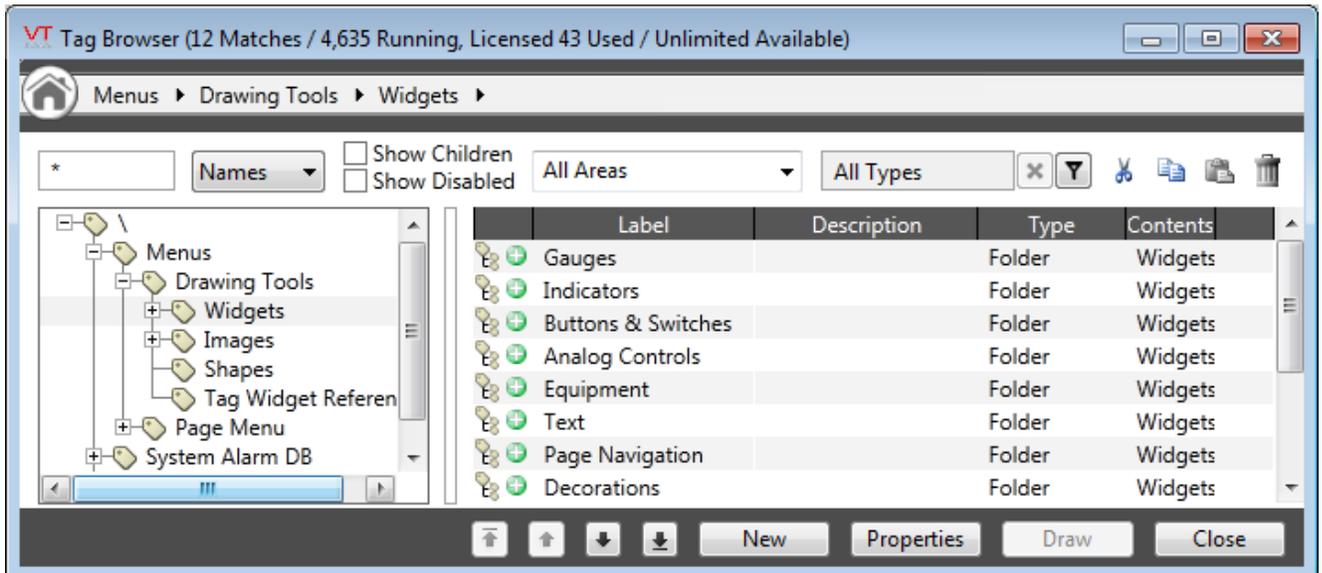
The palette display can be expanded or collapsed using the bar that divides it from the main page editing area. The bar may be dragged to the left or right to adjust the area of each. A double-click on the bar will collapse (or re-open) the palette. You can also collapse or re-open the palette using the Palette tool in the Home ribbon.
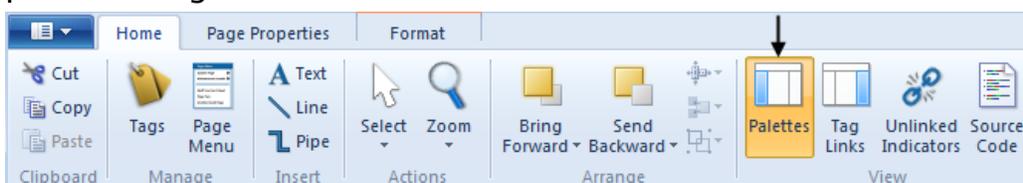
Recently-used objects are shown in a window above each palette. Dividing the main palette from the list of recently-used objects is an address bar that helps you to keep track of which folder you have navigated to within the palette. You can click on entries in the address bar, using it to navigate up or down the folder list.
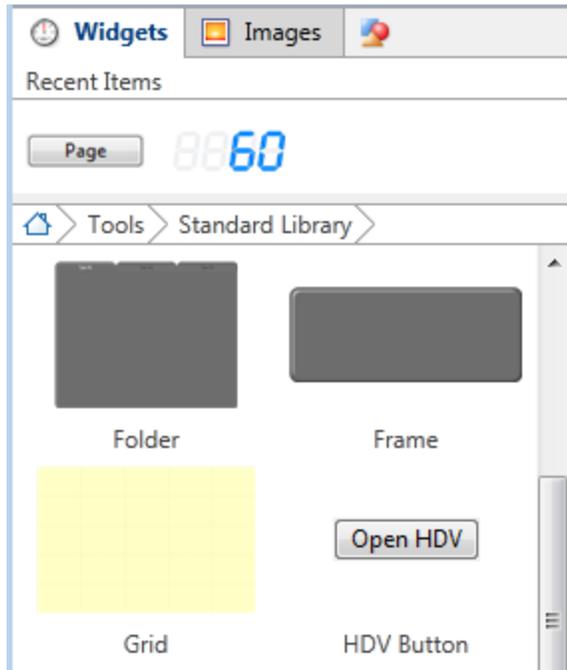
You can edit the contents of each palette, either with the Page Menu button in the Home ribbon (which opens the Menu Items section of the Tag Browser) or by right-clicking on any item in any palette.

Editing the Palette MenuItem Tags

Palettes can be edited either within the Menus section of the Tag Browser, or by right-clicking directly within a palette.

The right-click menu will vary slightly, according to whether you click on a menu item, a folder, or empty space within the palette. Features within the context menu will be enabled or not based on the context, and labels will vary slightly. The functionality of each command, when enabled, does not change from one version of the right-click menu to the next. Remember that each item in the palette, including folders, is a Menu Item tag in the Tag Browser. The following commands are shortcuts to work with Menu Item tags.
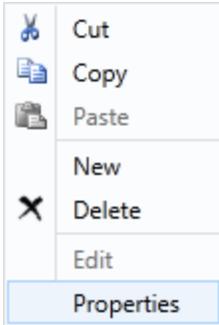
Cut, Copy and Paste	These are used together to create new Menu Item entries based on existing entries.
New / New Child	If you right-click on a folder, the option will be "New Child". Otherwise, it will be "New". In both cases, the result is that the appropriate version of the New Menu Item Properties dialog will open.
Delete	Removes the Menu Item tag, and therefore the entry in the palette.
Edit	Enabled only for user-defined items in the Widgets palette. Opens the widget for editing in the Idea Studio.
Properties	Opens the Menu Item tag's properties folder for editing. The specifics of what can be edited depends on the type of Menu Item selected.

Related Information:

...Draw Widgets - Reference for the Widgets palette.

...VTScada Graphics Library – Reference for the Images palette.

...Draw Shapes - Reference for the Shapes palette.

...Adding New Menu Items - Adding new folders and other menu items. Reference to the various forms of the Properties dialog.

Panels

The panels section of the Idea Studio has two tabs: Tag Links and Multi-Image Widgets. Use the Tag Links panel to edit the links between widgets and tags in a column format. Use the Multi-Image Widgets panel to control which images you want to edit within a widget that has more than one image.

The panels are not visible by default in the Idea Studio, but will open automatically to the Multi-Image Widgets view whenever a multi-image

widget is selected. Click the Panels button in the Home toolbar to open or close.
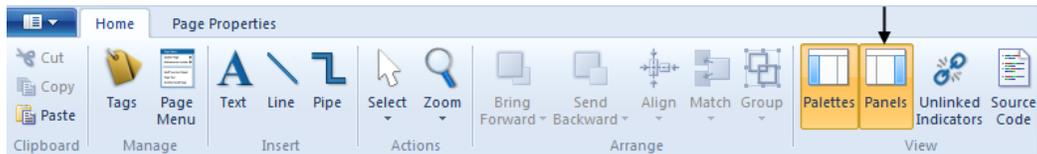
Tag Links Panel

Use the Tag Links panel to view and change the linkages between selected widgets and tags.

The list is sorted alphabetically by linked tag name, re-sorting as each link is made or changed. Unlinked widgets are always at the top of the list.

You can choose to have the panel display only selected widgets, or to display all widgets in the open page or user-created widget. The option to make this choice can be found at the bottom of the panel.
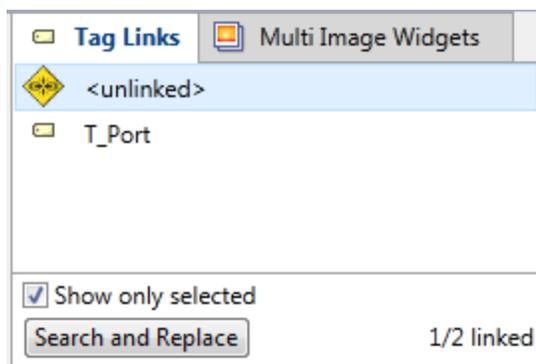
The same region of the panel also shows a count of widgets and the button to open the Search and Replace dialog, which is useful when you want to change several links in one step.

You can edit the links with exactly the same tools that are available in the properties dialog for a widget. Click on a link to see a tag browser button at the right, and a data source selection at the left. On the page, the matching object will be marked with a bold selection window, becoming the anchor object in the selection set.
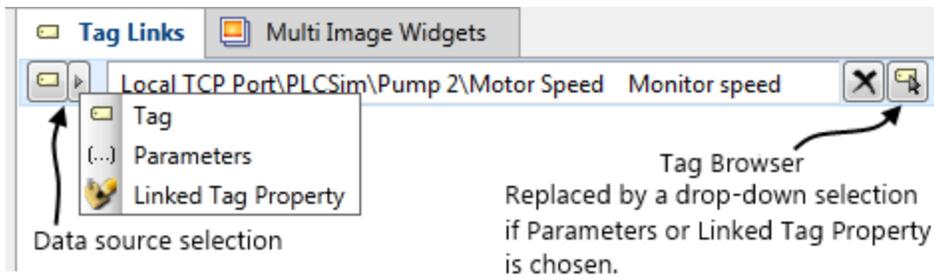
Tag Links | Multi Image Widgets

Local TCP Port\PLCSim\Pump 2\Motor Speed   Monitor speed

Tag
Parameters
Linked Tag Property

Data source selection

Tag Browser
Replaced by a drop-down selection
if Parameters or Linked Tag Property
is chosen.

Note: The Multi Image Widgets panel may be configured to open auto-matically whenever a multi-image widget is selected. This may be inconvenient at times. You can disable this behavior within the Multi Image Widgets panel.

Search and Replace Tag Links

Given a set of selected widgets, you can replace several tag links in one step by using the Search and Replace button of the Tag Links panel. For example, you may have drawn a set of widgets related to Pump 1 on a page. You also have a similar set of tags for Pump 2 and for Pump 3, each of which must be drawn on its own page.

Note: The preview of the search and replace operation is updated as you change each field. The links are not changed until you click the Replace button.

Preliminary Steps:

- You have created I/O tags for Pump 1, Pump 2 and possibly others.
- You have created a page for Pump 1 and linked those widgets to the tags for Pump 1.
- You have created a page for Pump 2 and have copied the contents Pump 1's page to Pump 2's page. All the copied widgets are still linked to Pump 1.
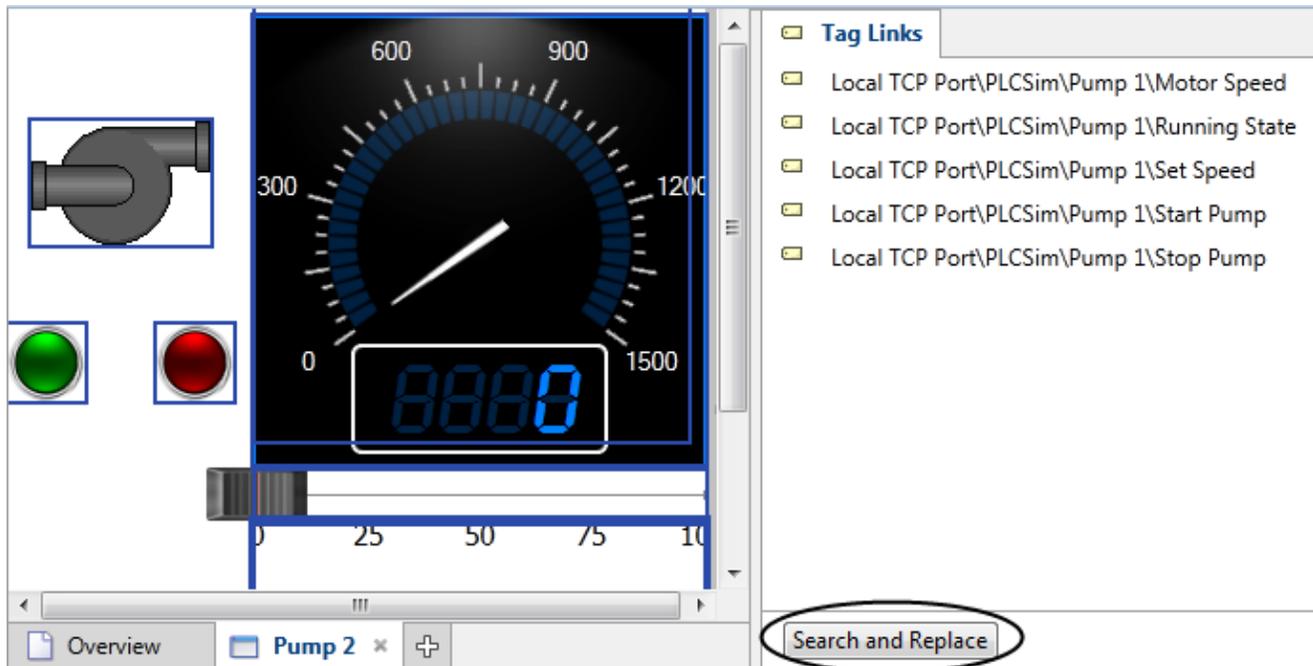
1. Open the Tag Links panel.

2. Click the Search and Replace button.

   The Search and Replace dialog opens.
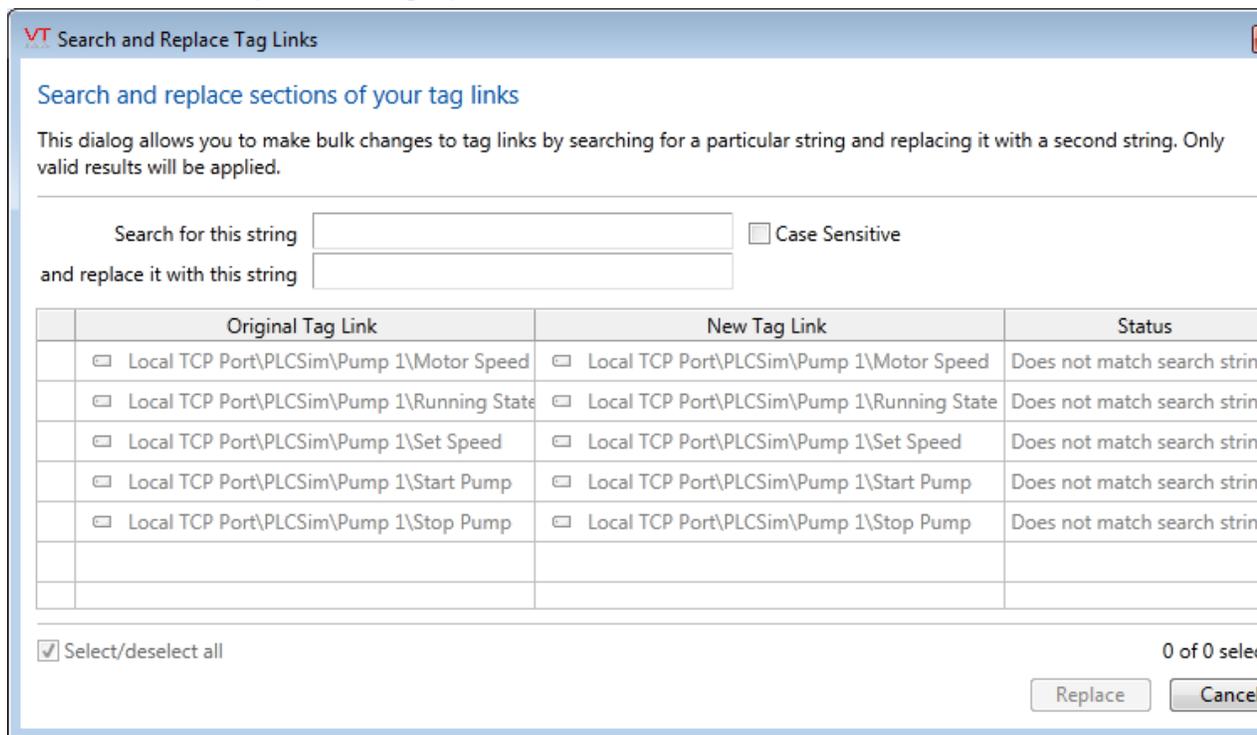
3. Type "Pump 1" in the field, Search for this string.

   All links that have "Pump 1" anywhere in the link are selected. The preview

operates immediately, replacing "Pump 1" with an empty string in every link. The status will change to "Bad tag reference." Nothing will happen to the widgets until you click Replace at the bottom right of the dialog.

4. Type "Pump 2" in the second field, then press tab or enter.
   The links will be updated to use Pump 2 in every case that previously used Pump 1.
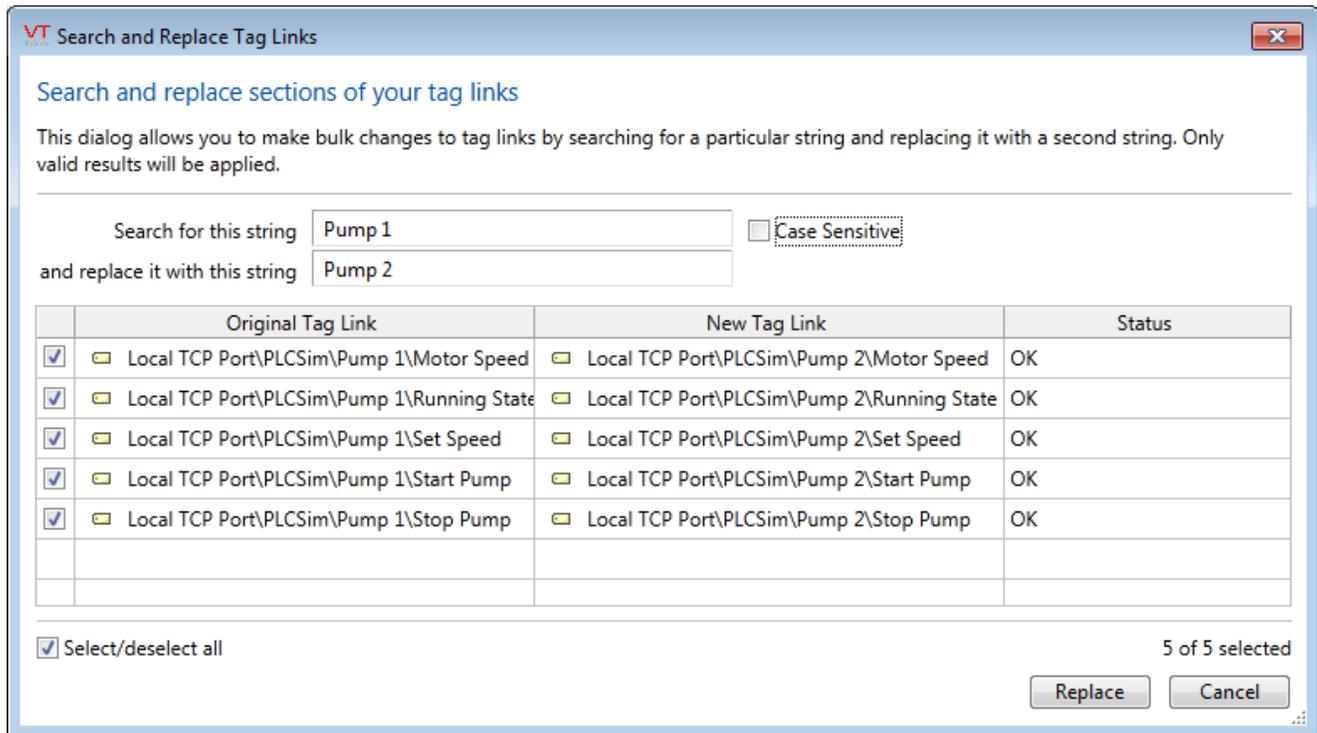
5. Click, Replace.

Note that you have the option of deselecting any links that matched the search, but should not be changed.

Related Information:

...Selecting a Data Source

Multi Image Widgets Panel

Opens automatically when any widget composed of multiple images is selected, including any that have just been drawn. The panel can also be opened or closed using a tool in the Home ribbon.

Note: The automatic-open feature can be disabled using a check box at the bottom of the panel.

When widgets containing component images are selected for editing, both the Widget ribbon and the Image ribbon are enabled. You are able to change the image used by the widget, and change all of the formatting characteristics of that image.

If the widget has multiple images, as is the case with the toggle switch, indicator lights, selectors and others, then you can choose which of the component images should be affected by certain formatting changes.

Note: Changes from the Adjust section of the formatting ribbon will apply to the selected component images. Changes from the Display section apply to the overall widget. For example, note that the Display section has a Rotation command, which will be applied to the full widget, while the Adjust section has an Image Rotation command, which will apply only to the selected component images.

All selected images will be added to the list, with their component images separated by relevant action or state.
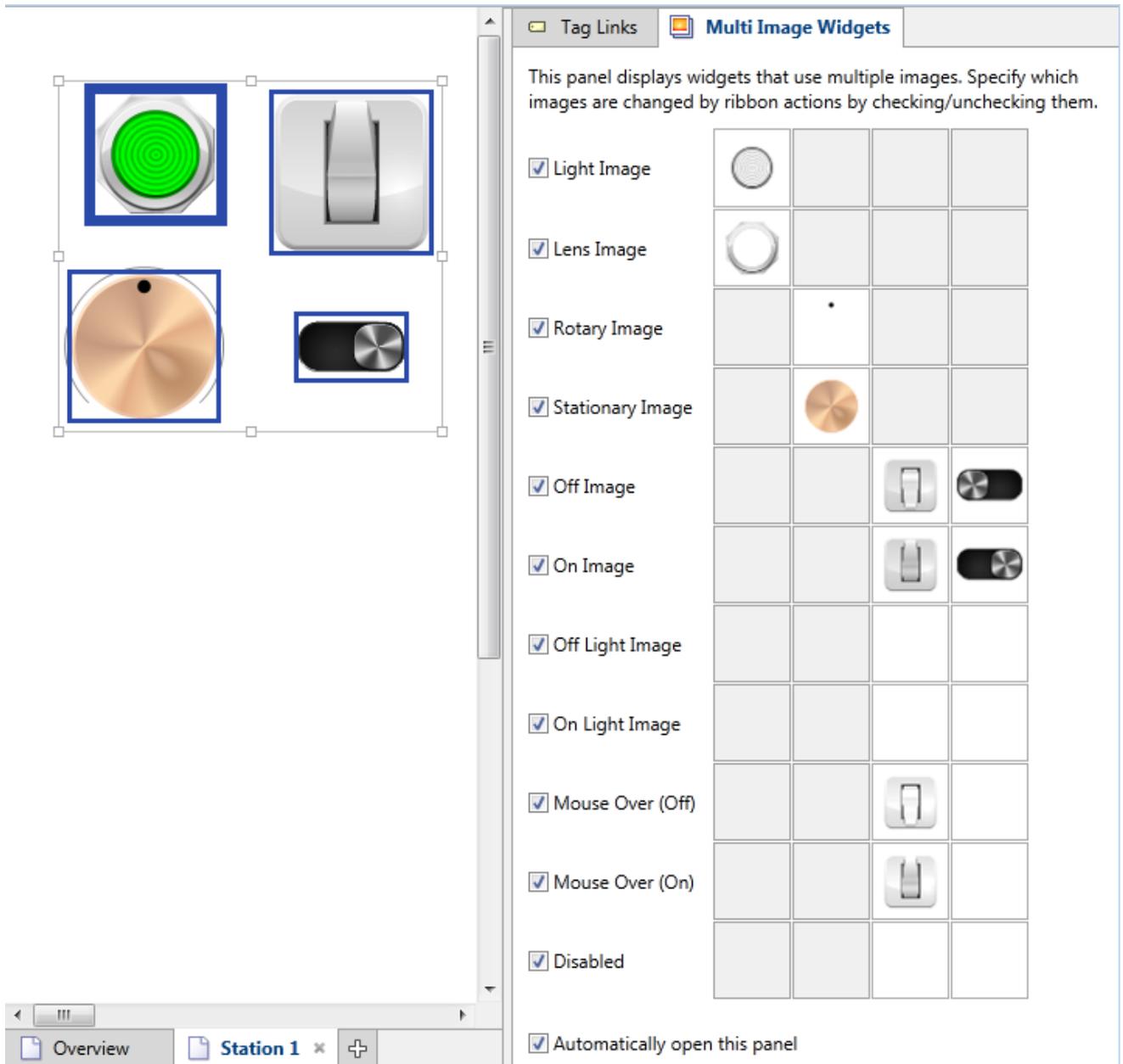
This feature gives you a fine level of control over image editing within widgets, and is also useful for making global updates to widgets. For example, you might need to change a large number of pump widgets from the old model to a new version. You can also use this feature to update a set of Image Change widgets to use a new color scheme for each state. (The states of an Image Change cannot be linked to a Style Settings tag, therefore the color selection for each state must be set directly.)
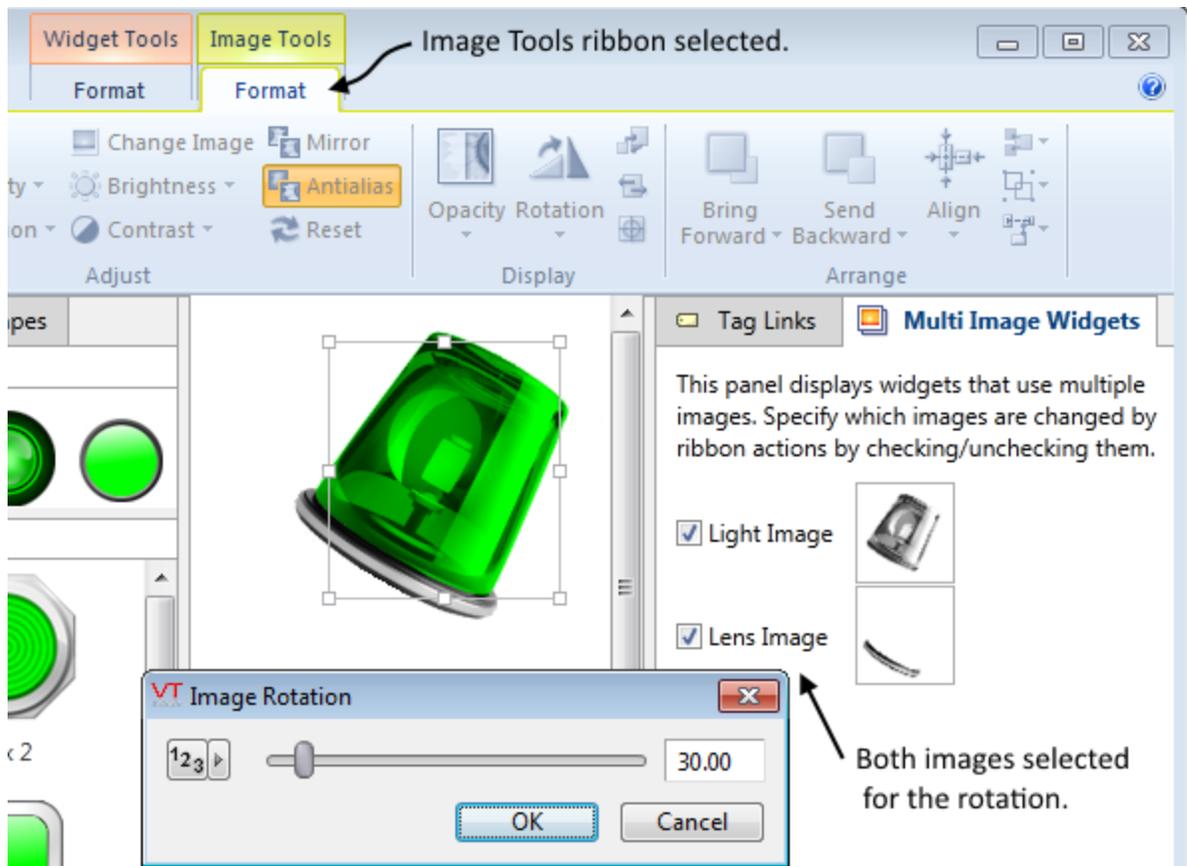
Rotating both images in a widget.
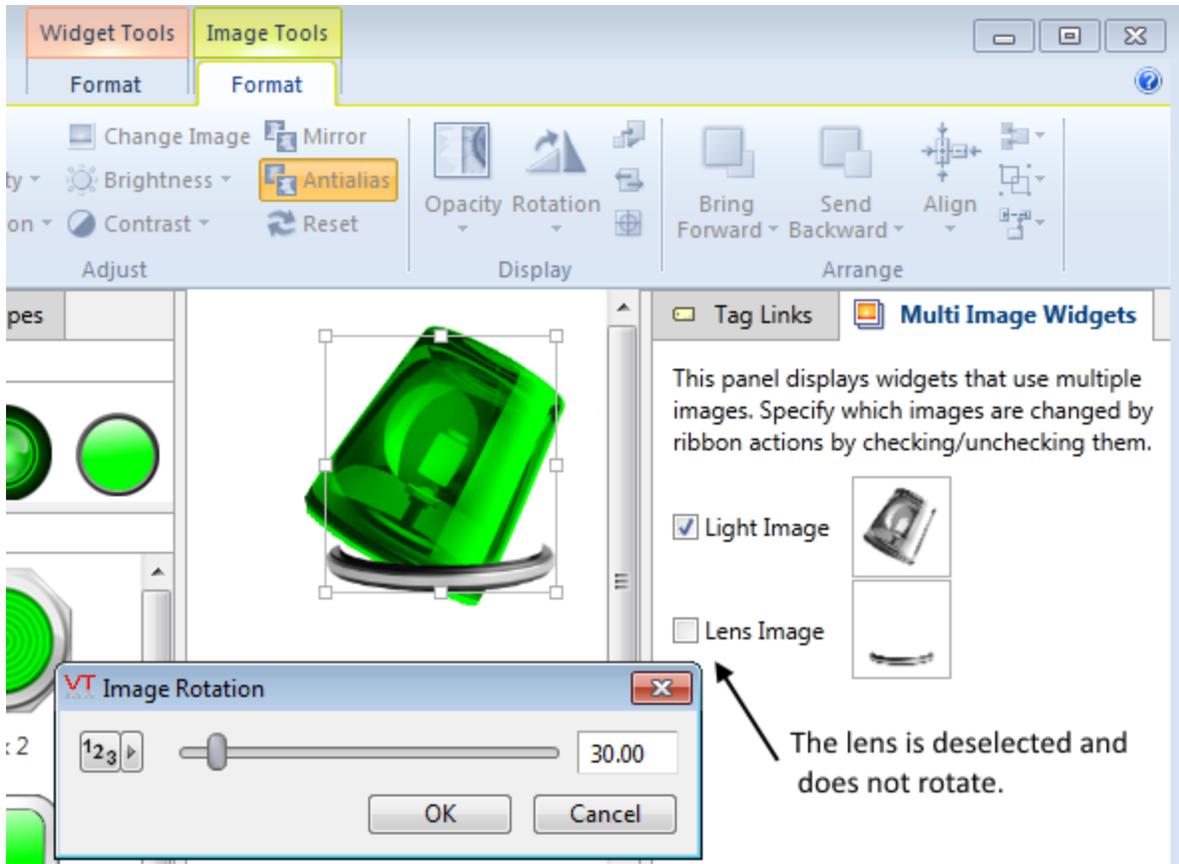
Editing only one image of a multi-image widget

The auto-open behavior of the Multi-Image Widgets panel is under your control. You will find the option to disable it at the bottom of that panel:
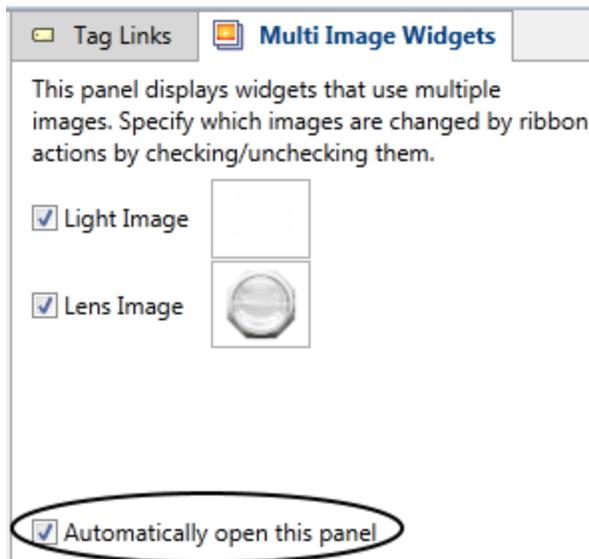
Create a Page (or Widget)

There are several ways to create a page.

> Note: The steps are identical to those for creating a new widget, except that Ctrl-N will create only pages.

The end result will be the same, whichever method you choose. You can also import existing pages from another application, but be aware that you may also need to import images or widgets that exist only in the other application, and that linked tags will not be imported with the page.

When creating a page, you will be asked for a title, and (with some methods,) whether you want a standard page or a pop-up. Both can be changed later using the page properties ribbon.

Preparation:

- The Idea Studio must be open before you can create a page.
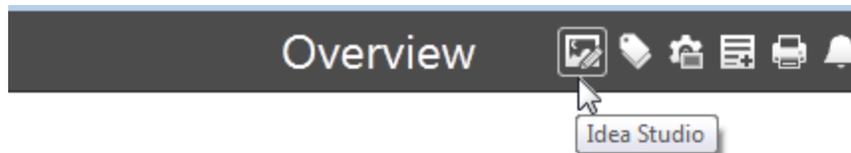
- If security has been enabled, your account must have configuration privileges and the Page Add privilege.

New Page Title / New Widget Title

When creating either a page or a widget, you will always be prompted for a title. A default value is provided, but you are advised to create a meaningful title for the page. It can be changed later, if required.
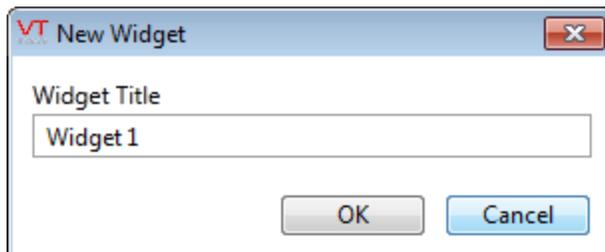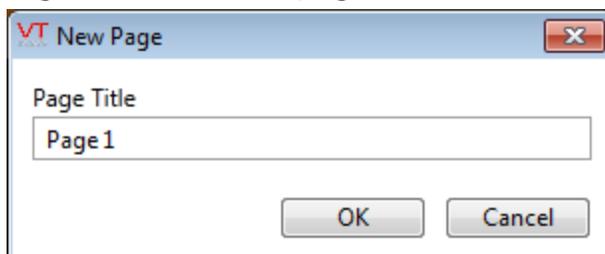
Prompting for a title – this dialog will be seen for every page and widget creation method.

Steps (Option 1):

1. In the Idea Studio, press the key combination Ctrl + N on the keyboard.
2. Provide a title for the page in the dialog that opens.
3. Click OK to save the title and close the dialog.

Steps (Option 2):

1. In the Idea Studio window, click the "File" menu to open it.
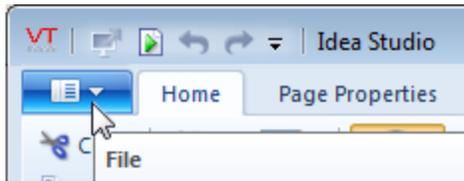
2. Click on or hover over the option, "New" expanding the option list.
3. Click either Page or Pop-up Page, depending on the type you wish to create.
4. Provide a title for the page in the dialog that opens then click OK.
5. Choose whether to include the page in the menu.
   You can edit the menu later if you change your mind.
6. Click OK to save the title and close the dialog.

Steps (Option 3):

1. In the Idea Studio, click the plus sign at the bottom of the editing window.
   The New Tab dialog opens.
2. Choose create a new standard or pop-up page.
3. Provide a title for the page in the dialog that opens.
4. Click OK to save the title and close the dialog.

The end result will always be the creation of the page or widget and the addition of a new tab to the list of pages and widgets open.

Click the + button
to open the New Tab dialog.
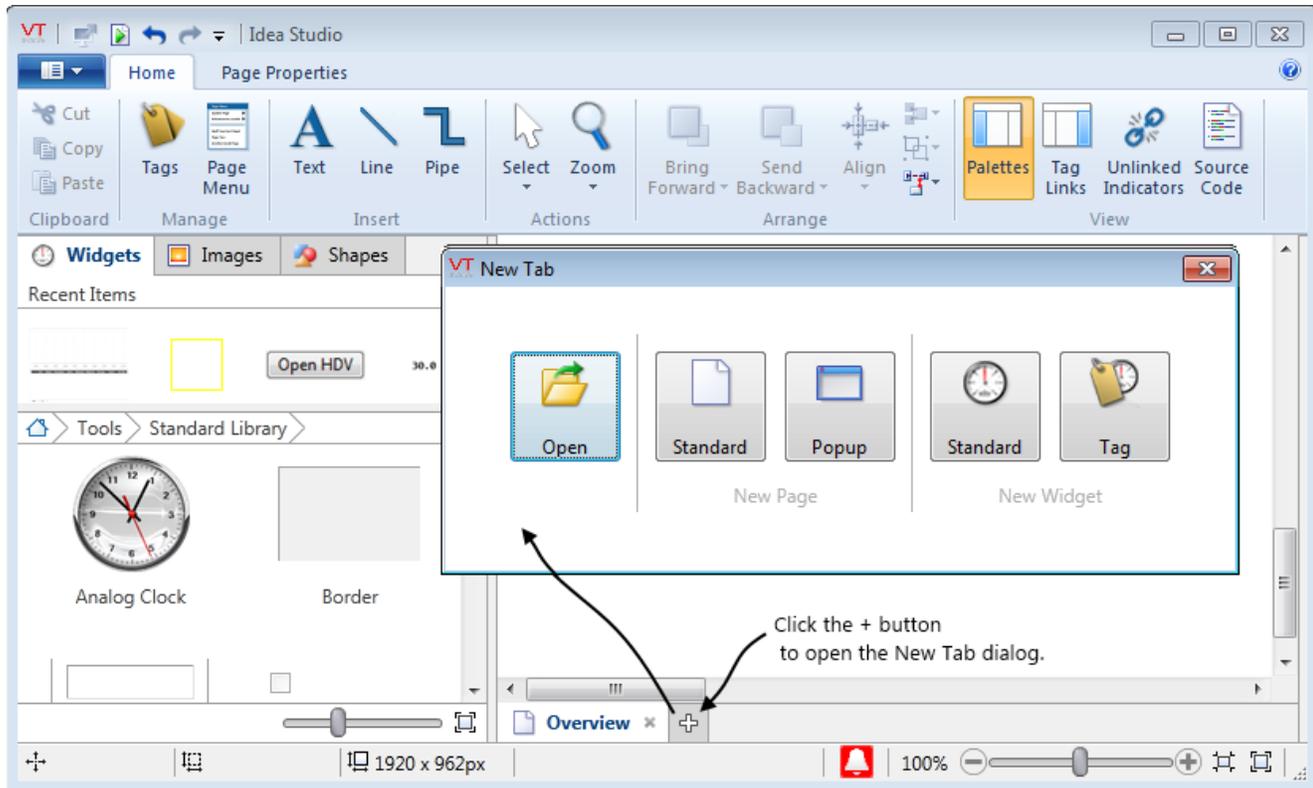
Steps (Import an existing page):

1. In the Idea Studio, click the File menu to open it
2. Click Import.
3. Click Page.
4. Use the Browse button to find and select the .SRC file of the page you want to import.
5. Provide a title for the page in the dialog that opens.
6. Click OK to save the title and close the dialog.

Having created a page, you can configure how it opens (full screen or dialog window), its appearance, size, parameters and more. All configuration is done using the Page Properties ribbon.

Steps (Clone an existing page):

An application may require several very similar pages or widgets. Rather than create each from start to finish, it can be easier to copy or clone an existing instance and make a small number of changes.

To clone a page or widget:

1. Open the page or widget in the Idea Studio and ensure that it is the selected item.
2. Open the file menu.
3. Click the command, Clone.
4. Type a name for the new instance, then click OK.
   The new instance will be created and opened for editing.

Alternate steps:

1. Open the page or widget in the Idea Studio.
2. Right-click on its tab at the bottom of the Idea Studio.
3. Click the command, Clone.
4. Type a name for the new instance, then click OK.
   The new instance will be created and opened for editing.

General Display Properties

You can control how the display appears to operators, including the color theme, the format of the time and date clock in the title bar, whether the page is full screen or windowed, and more.
The most commonly changed properties are available in the Display tab of the Application Configuration dialog. More can be accessed through the Advanced Mode of that dialog.

Related Information:

...Page Properties – Reference guide for the Page Properties toolbar. Refer to the VTScada Admin Guide for the following:

...Display Tab of the Edit Properties Page

...Display Manager Properties

Next Steps:

...Define Navigation Between Pages – Create menus, navigational hot-boxes, and buttons.

...Build the User Interface with the Idea Studio

...Restrict Access to a Page – Prevent unauthorized viewing of pages.

...Delete a Page – Remove pages (and widgets) that are no longer needed.

Limit Pop-up Pages

If your application uses pop-up pages for controls or other features, you may wish to limit the number of such pages that can be open at any one time. By default, VTScada will allow four pop-up pages, to be opened at once before a warning is displayed to the operator. More pages may be opened, but the operator must chose to do so by acknowledging the warning.
Trend windows (HDV graphs) that were opened by clicking a tag do not count to ward the pop-up page limit.
A related option is that you might want to close all pop-up pages when the operator changes the page in the main display. This is sometimes done on the basis that the pop-up pages are often controls related to the current display. When the main display changes, the control pages should close.
Use the following application properties to limit or control pop-up page behavior.

PopupPageLimit
– Sets the maximum number of pop-up pages that may be open at any one time.
If the operator attempts to open more pages than set by this limit, then the action will depend on the next property:

PopupLimitAction
Use this property to set one of the five possible actions:

0 or not set	No action. The limit will be ignored.
1	Warn. Display a warning, but allow the operator to open the pop-up page.
2	Prevent. Display an error dialog, and prevent the opening of the pop-up.
3	Close Oldest. Whichever pop-up has been open the longest during the current session will be closed so that this one can open and remain within the limit.

| 4 | Close Newest. Close the second most recently opened pop-up page what that the current page can be opened without exceeding the limit. |

PopupCloseOnPageClose

Close the pop-up page automatically if the page from which it was opened closes. (There must be a hotbox or hot-button on that page.) This action can be overridden on a page-by-page basis by adding the variable, PopupCloseOnPageClose, to the page. If this variable exists and is set to 0 (FALSE), then the page will remain open while others close. Variables can be added only by editing the page's source code. Remember that it is necessary to import file changes after editing a source code file.

There is no method for adding this variable to a system page.

Related Information:

...Refer to the VTScada Admin Guide for the following property definitions:

...PopupPageLimit

...PopupLimitAction

...PopupCloseOnPageClose

Open a Page or User-Defined Widget to Edit

When you open the Idea Studio, the current page in the application will be the one that the Idea Studio displays first.

If the page had been open for editing earlier, it will be accessible using the tabs across the top of the Idea Studio screen.

Click to choose which previously opened page you want to edit.
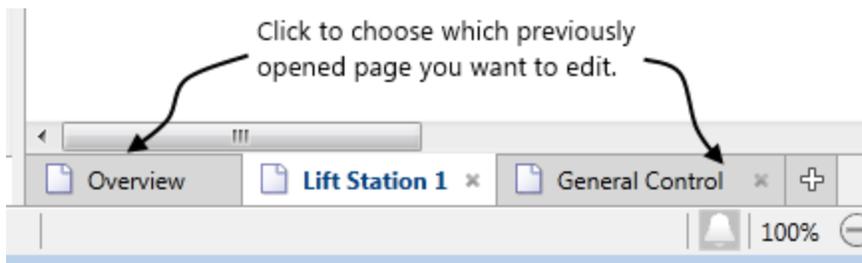
If the page has not yet been opened in the Idea Studio:

1. In Idea Studio, click File
2. Click Open
   Alternate methods: Press the keyboard combination, Ctrl-O or click the plus button in the list of open pages.
3. Select the page or widget that you want to edit.
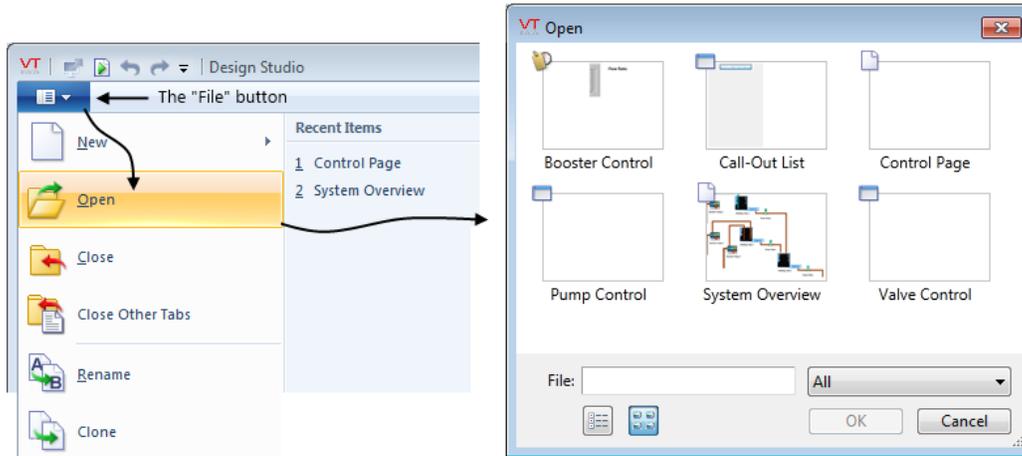
Symbols used in the "Open" dialog:

- This is a dialog-box page.
- This is a full-screen page.
- This a widget.

The "Open" dialog can display pages and widgets in a list format instead of the default preview format. This can be useful if you want to see the name of the file that each item is stored with.
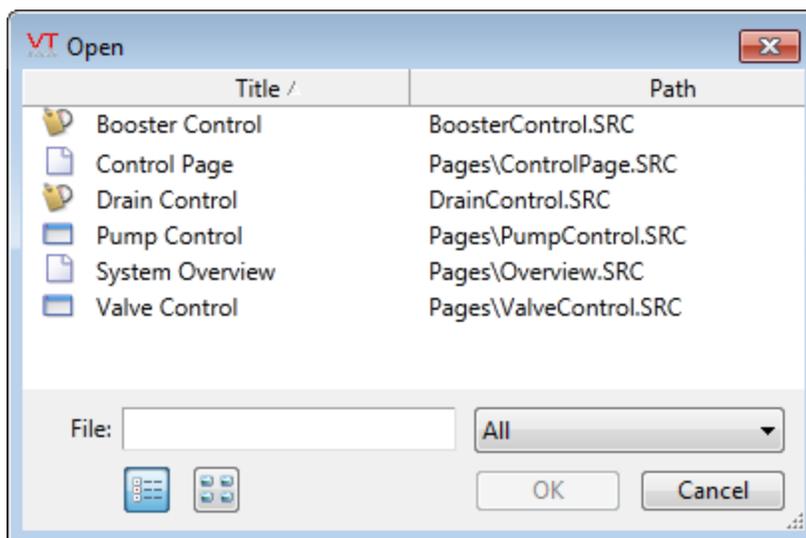
Delete a Page

Pages that are built into VTScada, such as the Alarm page and the Reports page, cannot be deleted.

Pages that you created may be removed using either of the following procedures.

Method 1:

1. In the Idea Studio, open the page for editing.
2. Open the File menu.
3. Click the Delete entry.
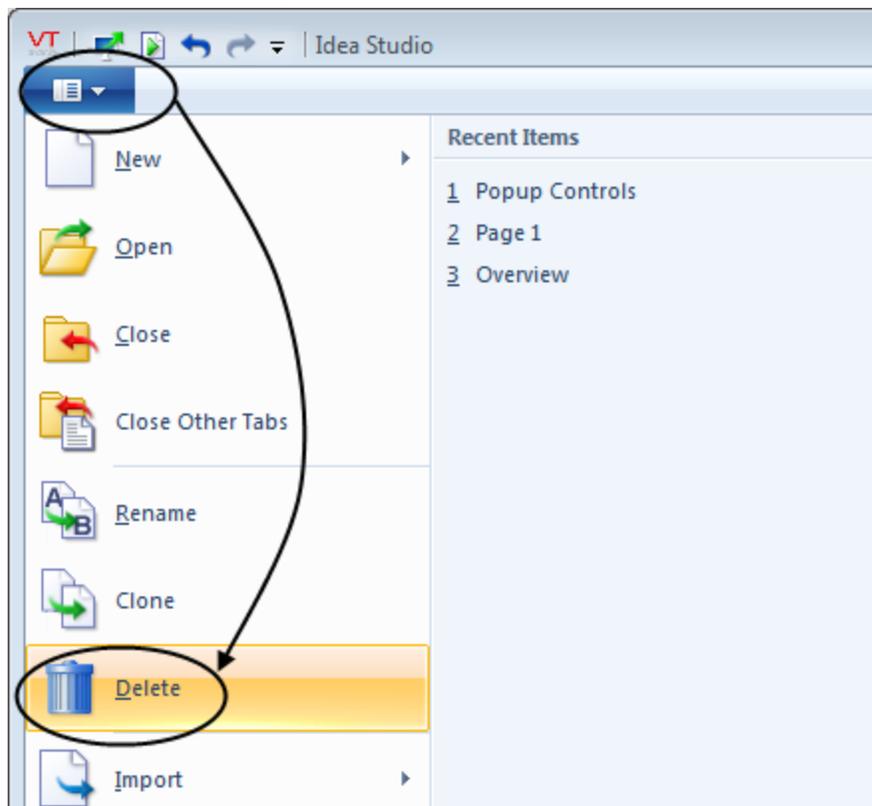
Method 2:

1. In the Idea Studio, open the page for editing.
2. Right-click on the page tab.
3. Click the Delete entry.
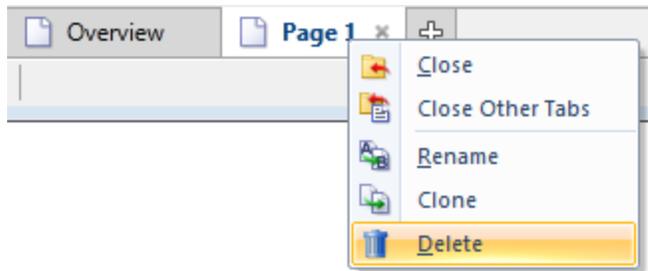
Page Properties

The Page Properties ribbon of the Idea Studio provides full control over the appearance and behavior of each **application page**[1]. The selected page in the editor will be the page modified by the toolbar.

Page management commands such as New, Open, Import and Delete are found in the File menu, rather than the ribbon.
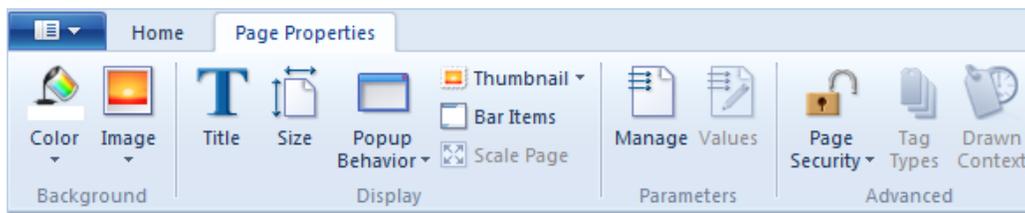
Color
Sets the background color of the page, using the color selector.

Image
Use any image on your computer for the background of the page, using the Image Selector.

---

[1]A display screen that you create and have control over.
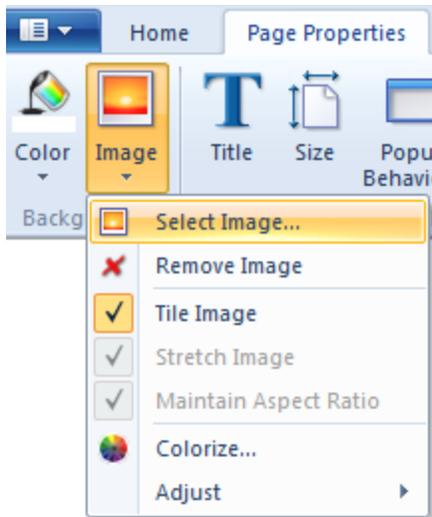
If a background image has been applied, you can use this same tool to remove the image.

You can force an image to fill the page using either of two options: Tile will cause the image to repeat in an array until the entire page has been filled. Stretch will re-size the image until it fills the screen. If stretching the image, you can choose whether or not the aspect ratio is to be maintained.

The Colorize option opens the Select Color dialogue, with which you can add a color to the image.

The Adjust menu contains six further controls:

Opacity	Adjust the visibility of the background by reducing its opacity, making it semi-transparent.
Brightness	Adjust the brightness of the colors in the background.
Contrast	Adjust the contrast between bright and dark shades in the background.
Mirror	Flip the image left-to-right.
Tile Scaling	Applies only if the Tiled option has been selected. Adjusts the size of each tile. You may select one of the pre-set scale options, or click "Custom" to open the Tile Scaling dialog.
Margins	While a page has a background image (and only if it has a background image), you can adjust the margins around that image to control its location.
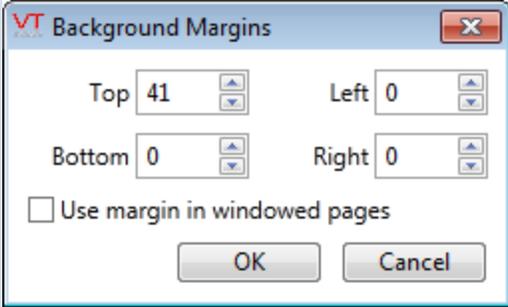
The top margin defaults to 41 pixels, allowing space for the VTScada title bar. While you can adjust all four sides, you will generally find it to be most effective to adjust only two edges, setting a margin for either the top or the bottom, either the left or the right.
As indicated by the check box, the values do not apply to pop-up windows unless you direct them to do so.

Title
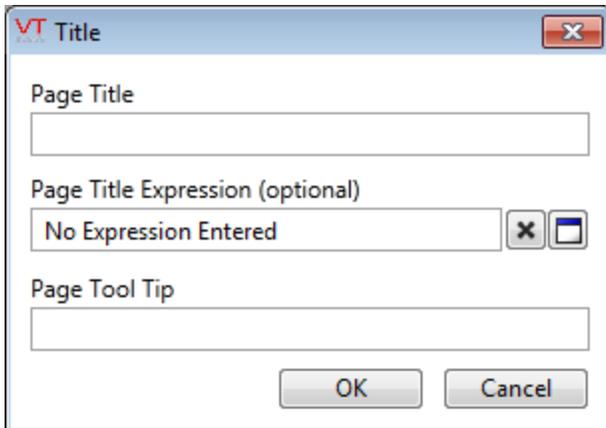
Note: The same dialog is also used when renaming a page or widget.

The page title is the displayed name, used in menus, the navigation bar, and at the top of the screen while the page is open.
You have the option of providing an expression, which will override the displayed name of the page, but not the name shown in the menu or the navigation bar. Title expressions are typically used with parametrized pages, where one page can display different tag values depending on the parameter set used when it is opened. In these cases, the expression is designed to provide some indication of the parameters being displayed by each instance of the page.

The page tool tip is seen only when the operator hovers the pointer over the pinned page name in the navigation bar. The contents of the tool tip field in this configuration dialog are displayed as a sub title to the page name.

Size
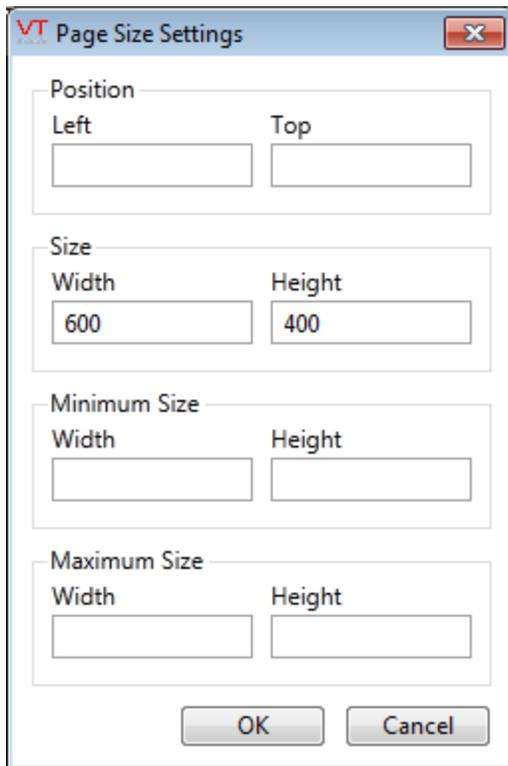
Size settings apply to pop-up pages. Using horizontal and vertical coordinates measured in pixels, you can set initial, minimum and max-imum sizes and the location on the screen, measured from the top left corner of the screen to the top left corner of the page.
Size settings have no effect on full-screen (standard) pages, but they can be extremely useful in limiting the apparent page area in the Idea Studio. By setting a page size that matches resolution of your monitor (or the smallest monitor that the application will be displayed upon) the Design Studio's working space will help you stay within this area.

Pop-up Behavior
Choose between No Restrictions, Always Pop-up and Never Pop-up.

A page marked as "No Restrictions" will fill the application area when the operator left-clicks to open it, but will open as a pop-up window if the operator right-clicks to open it.

Thumbnail

Optionally select or remove the thumbnail image, used for reference when working with the page. The thumbnail image is not shown as part of the page, like a background.

Bar Items

Choose to include or exclude standard items in the title bar and task bar of the page.
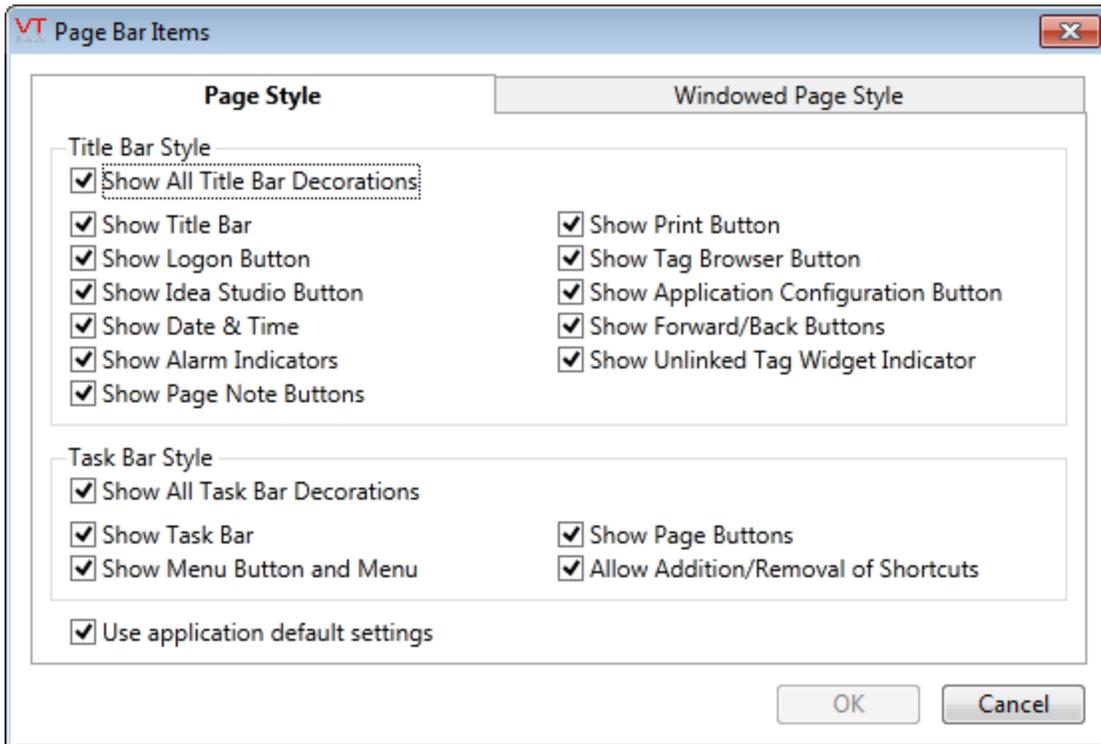
A title bar with all decorations shown:
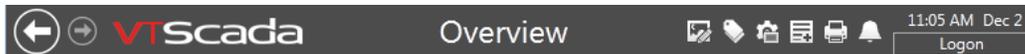
A task bar with all decorations shown:

Scale Page

Applies the application property ScaleDisplayContent to the contents of this screen. The larger of the page contents or the defined page size will be scaled to fit the operator's screen. The Idea Studio is not affected by this property.

This check box is enabled only when ScaleDisplayContent has been set to TRUE. That property can be set using the Display Properties tab of the Application Configuration dialog.

> Note: Automated display scaling works reasonably well when enlarging the page. It cannot do as good a job when shrinking a display for a smaller screen. In particular, labels embedded within buttons or widgets are more likely to be truncated than scaled down.
>
> Always design for the smallest screen that the application will be displayed upon.

Manage Parameters

By parameterizing a page, you can re-use the same page for similar sets of tags. For example, one control page can be used for a series of stations, assuming that each station has the same hardware.

Related Information:

Page Parameters & Widget Parameters

Parameter Values

Enabled only when the page has parameters. When editing a page, widgets and other items that are linked to parameters may not be displayed unless a tag has been chosen for, or a value has been assigned to the parameter. Click this button to open the Manage Parameters dialog, where you may assign values to be used during the current editing session.
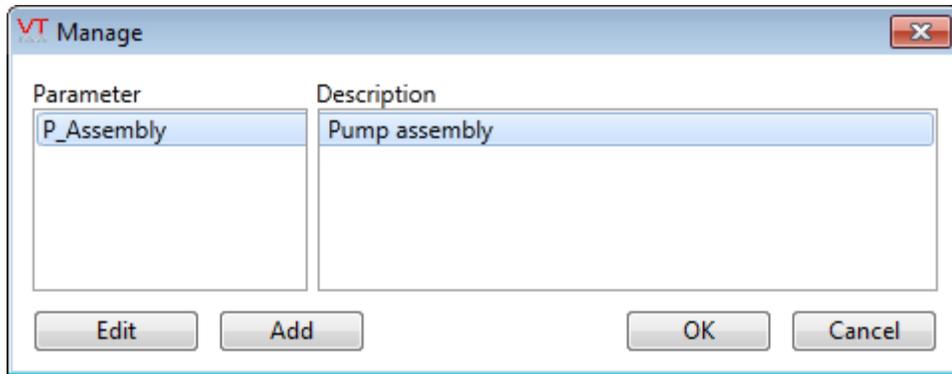
The Manage Parameters dialog will always be displayed when opening a parametrized page for editing, or when the navigational link to the page does not have values assigned to the page's parameters.

Page Security

Use this to apply an existing application privilege to the page, thereby denying access to everyone who does not have that security privilege. (Note that security cannot be applied to widgets.)
If application privileges have not yet been created, or if security has not been activated, this menu provides access to the security accounts and privilege dialogs, for developers who are authorized to access those dialogs.

Tag Types

Applies only to user created widgets. Use this dialog to control the list of tag types that can be linked to the widget.

Drawn Context

Relates to widgets, not to pages. When pressed, it opens the Tag Browser, allowing you to change the tag scope that the widget will be opened with.

Related Information:

...See: "Quick Reference Guide for Expressions" in the VTScada Programmer's Guide

...Color Selector

Related Tasks:

Page Parameters & Widget Parameters

Parametrized pages and library widgets use parameters in the same way, and share configuration dialogs.

A parametrized page is one that can be reused for similar equipment installations. For example, your application may have multiple lift stations, each housing the same (or similar) equipment. The design of each page shows the equipment and differs only in which set of tags is being shown in a particular station.

Rather than create a separate page for each station, create one. But, link the widgets to parameters in the page rather than tags in the application. Navigational links to the page control which set of tags is used for the parameters when the page is opened via a particular link.

In the following example, a single control page was created, but it can be used for both pump 1 and pump 2 because the widgets are linked to parameters and a different set of tags is used for the parameters in each case. The page title uses a parameter within an expression so that operators will know which pump is being controlled.
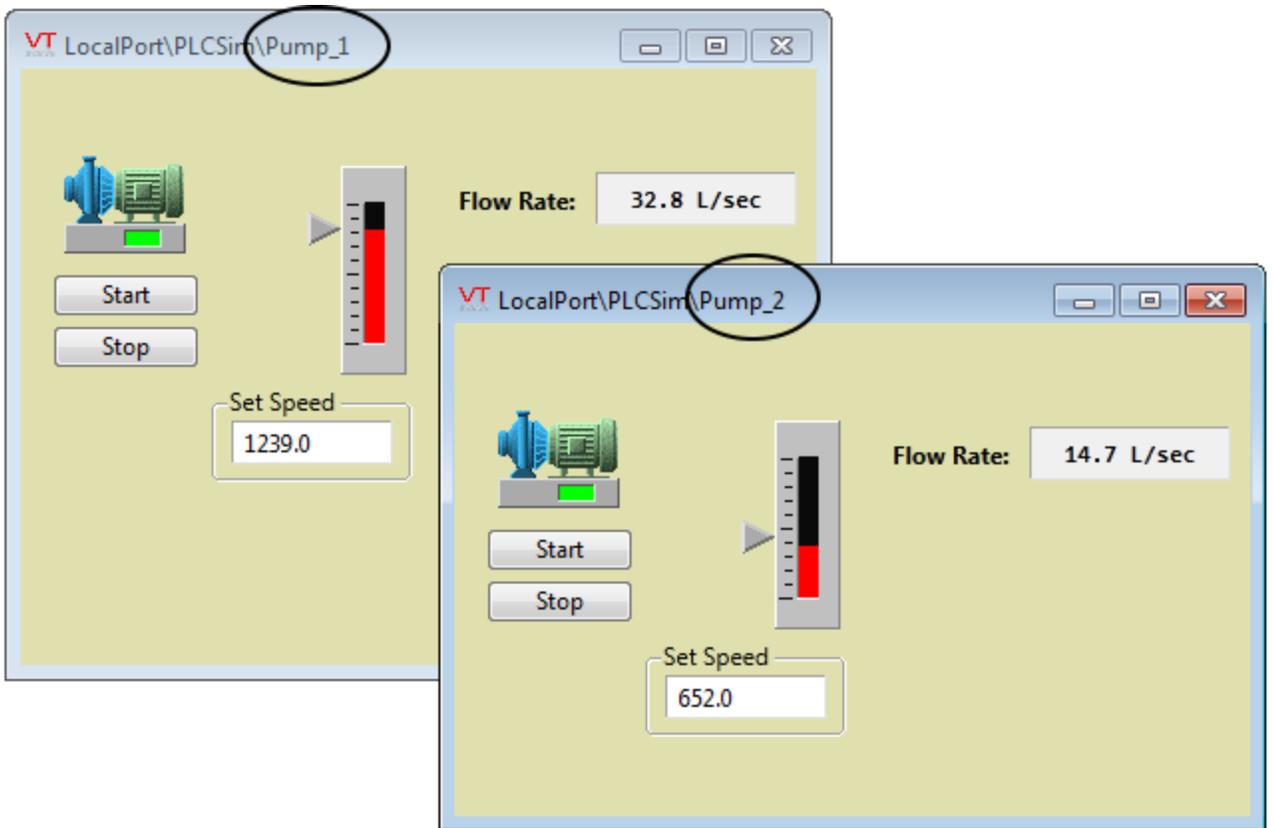
Parameters are typically linked to tags, but this is not a requirement. Parameters can be defined to hold numbers or text as well as tags. In all cases, the navigational link to the page must be configured to supply the value or tag to be used by each parameter.

Add parameters to a page using the Manage Parameters button in the ribbon, as shown. This button opens the Edit Parameters dialog, where you can view, edit or add parameters.
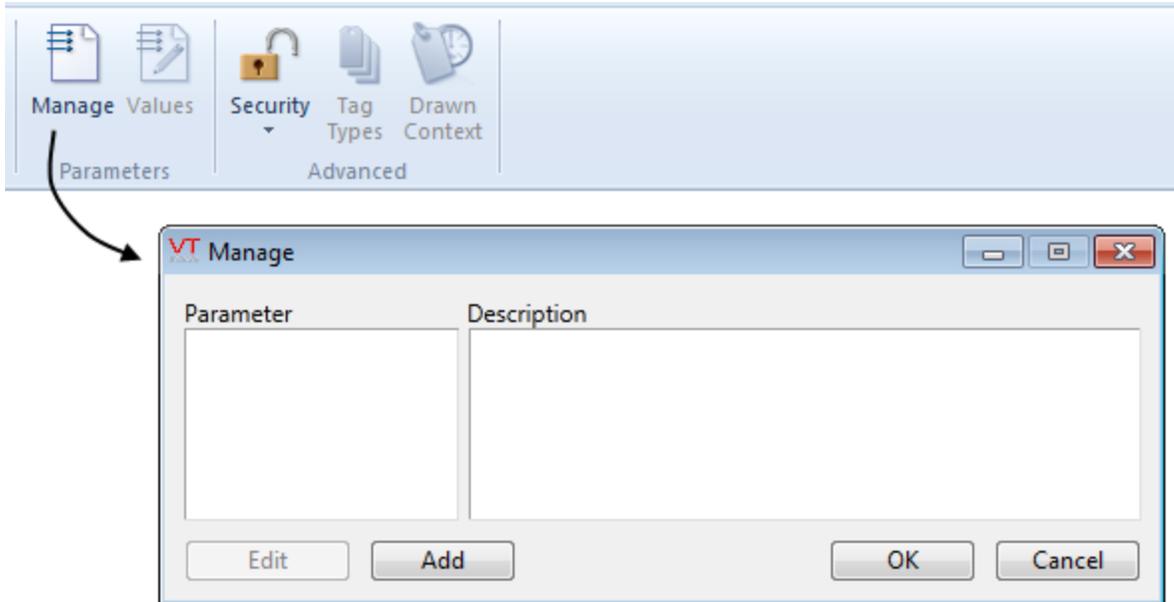
Clicking Add opens the following dialog, where you can configure the new parameter.
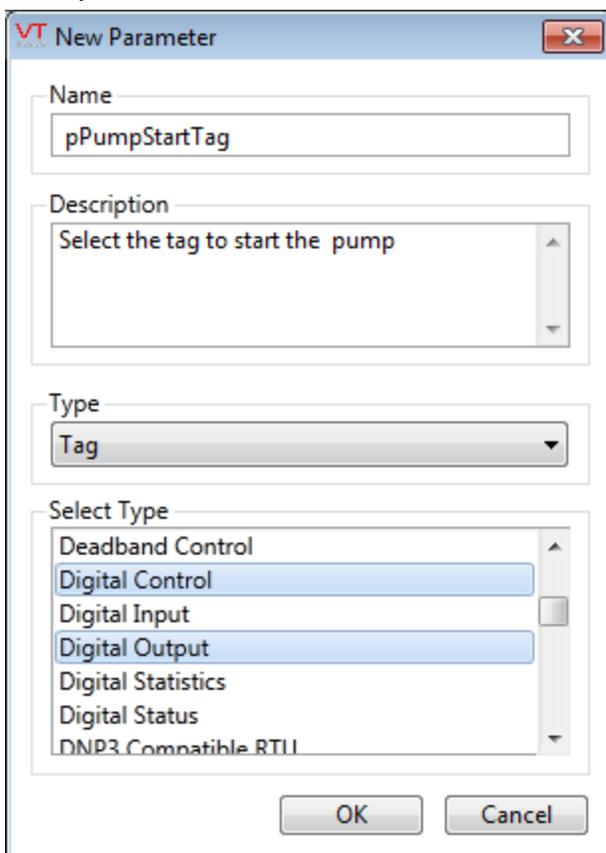
The example shows a new parameter being configured. Recommended practice is to ensure that the parameter name is unique, avoiding any

conflict with tags or other named objects. The description field should guide developers or operators when selecting the tag or value for the parameter. If the parameter is of type, tag, then you must select which types can be used. You may use the Ctrl key to select several types, as shown in the example image.

The same dialog is used when editing existing parameters.

There is no user interface control to delete parameters. If you are certain that a parameter is not needed and cannot be edited for a new purpose, then experienced developers can remove parameters by editing the page's source code.

Text Parameters & Color Selectors

The properties dialog for any page or widget that has a text parameter will provide basic text-entry field as shown:

If this parameter is meant to be used for color selection, then you should edit the code for that parameter as follows.

Original parameter definition:

```
(
  MyColor <:"Color Choice":> Text;
)
```

Modified parameter definition:

```
(
  MyColor <:"Color Choice":> Text<Color>;
)
```
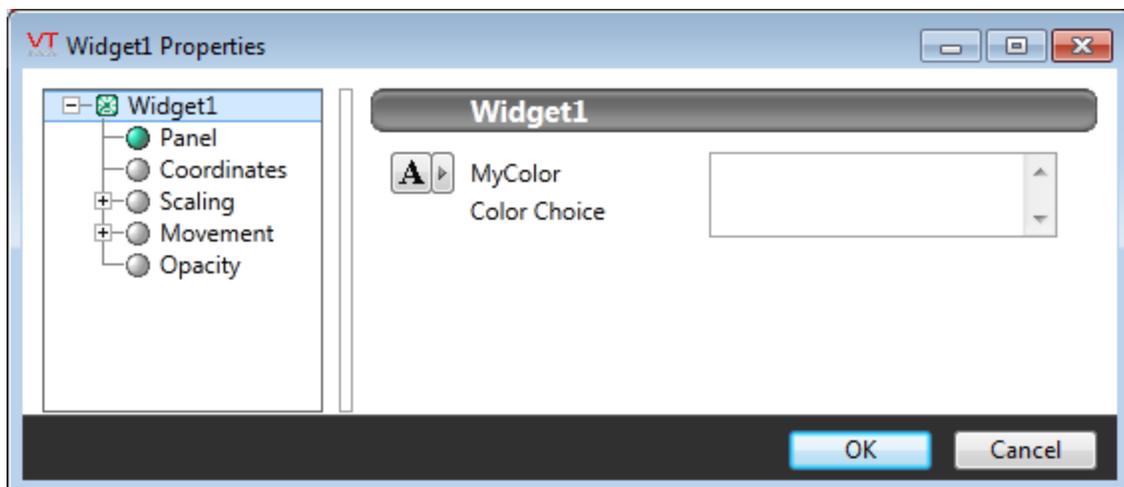
The addition of the <Color> modifier after "Text" tells VTScada to use a color picker rather than a text entry field in the properties dialog.
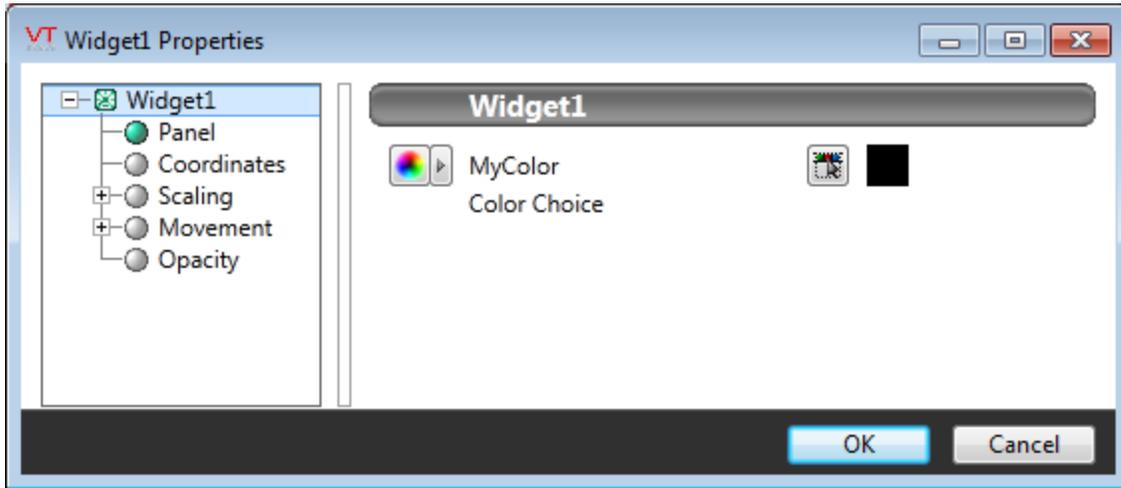
Page Size Scaling and Control for Varying Displays

It is often the case that an application built on one computer is displayed on another, or perhaps several other computers. Screen sizes and aspect ratios will vary.

One option to deal with this situation is to configure a page so that the display will scale automatically as the page is resized. This is a two-step process where step two cannot be done before step one.

First, you must enable the ScaleDisplayContent property for the application in general:
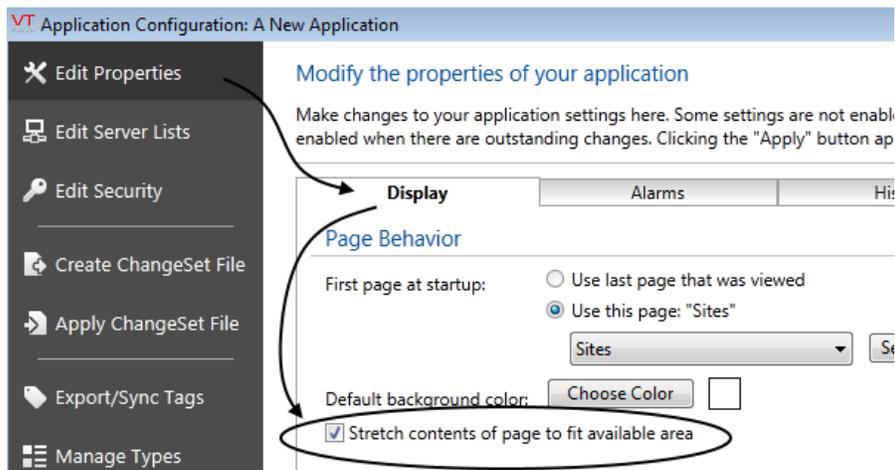
Second, you must enable or disable page scaling for each page within the application.
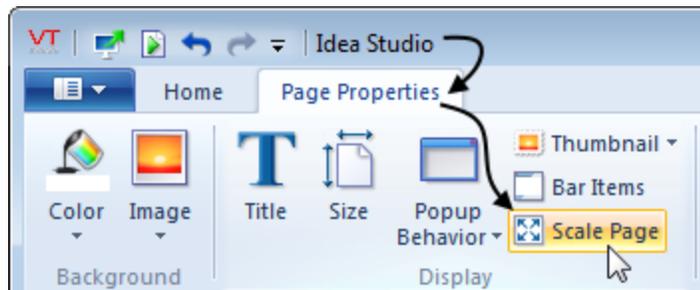
As described in the following links, other tools are available to help you control how your application will look on each workstation. Note that the Idea Studio includes zoom and pan controls, eliminating the need to modify page properties while doing development work.

Related Information:

...Scale Page – Enable the scale-to-fit option for a page.

...Size – Set the display size for pop-up pages.

...Realm Display Setup Tags – Control over page size and border decorations when viewed on an Internet client.

...Refer to the VTScada Admin Guide for:

...Display Tab of the Edit Properties Page – Enable page scaling globally, among other page display properties.

...Display Manager Properties – Reference for application properties that control the page size, aspect ratio and appearance.

Configure for Multiple Monitors

Configuration for a multiple-monitor display, or multi-page display on a single monitor, depends partly on your hardware and partly on VTScada configuration options.
Your graphics card must support either the Windows® Extended Desktop or the Windows® Expanded Desktop. If the graphics card does not support either option, you cannot proceed.

The difference between the two options is that an extended desktop will display the task bar in only one monitor while the expanded desktop extends the task bar across all monitors. Either can be used for a multiple-monitor display of your application, but additional configuration is required for an extended desktop. With an expanded desktop, each window will have a separate instance of the VTScada display manager. With an extended desktop, the separate display manager windows will be stacked side by side, which means that the VTScada page size must be configured to match the monitor sizes.

In the Application Configuration dialog of VTScada, set the properties DispMgrVert and DispMgrHoriz to the number of screens you wish to use. For example, to have a side-by-side display on two monitors, set DispMgrVert to "1" and DispMgrHoriz to "2".

Matching resolutions for an extended desktop: The display resolution of your monitors must be considered along with the configured size of the VTScada pages. For example, suppose that you want to view four VTScada pages on a single large monitor. The monitor size is 1920x1080. The default VTScada page size was set to 1024x768.

You begin by setting DispMgrVert to "2" and DispMgrHoriz to "2". The effective size required by VTScada is now 2 x 1024 = 2048 and 2 x 768 = 1536. The windows will not match the available screen size.

The solution is to set the DispMgrHeight and DispMgrWidth values to the monitor size (or total size of multiple monitors) divided by the number of pages that you want to display in each direction, allowing extra room for the Windows® title bar and border. Assuming that this is done after development has been completed, you should also enable the ScaleDisplayContent property so that your original page contents will be scaled to fit the new width and height.

Note: Automated display scaling works reasonably well when enlarging the page. It cannot do as good a job when shrinking a display for a smaller screen. In particular, labels embedded within buttons or widgets are more likely to be truncated than scaled down.

Always design for the smallest screen that the application will be displayed upon.

Related Information:

Refer to the VTScada Admin Guide for the following properties:

... DispMgrHoriz

... DispMgrVert

... DispMgrHeight

... DispMgrWidth

Sample Pages

VTScada comes with several sample pages. These are a set of widgets without tag links, drawn on a background of borders and tiled images. All can be opened for editing in the Idea Studio. You can examine these pages to learn how they were built and to gain inspiration. You can also copy sections for use in your own pages.

Dashboard A
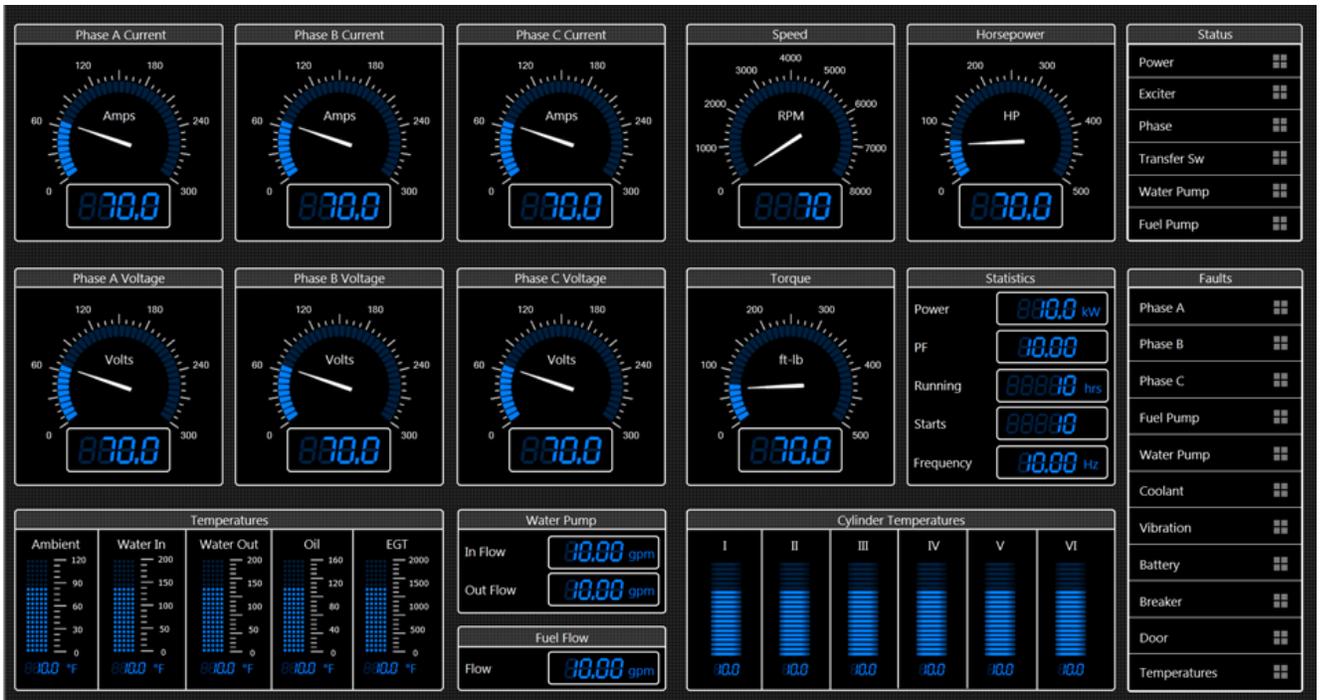
Dashboard B

Dashboard C

Define Navigation Between Pages

As you create pages in your application, it is necessary to give operators a way to change which page they are viewing. There are several ways to do this, as described in the following sets of instructions:

Define and organize the menu

Menu items are stored as tags. The contents and order of both the List Menu and the Tile menu follow the organization of menu item tags in the Tag Browser.
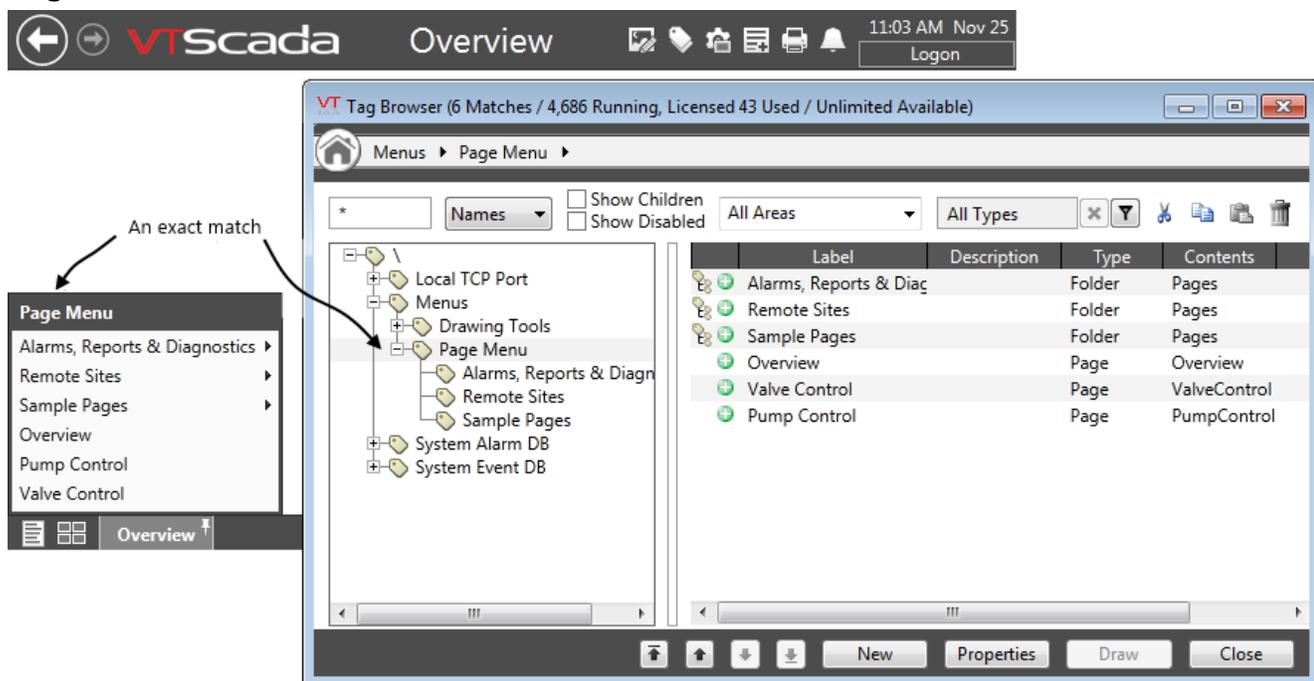
The menu structure is best edited from within the Tag Browser, but editing can also be done using the right-click context menu while viewing the Tile Menu. Features of that right-click are described in the following topic.

There are two ways to access the menu section of the tag browser:

- Open the Tag Browser and navigate to the Menu section.
- Within the Idea Studio, click the button, Page Menu, found in the Home toolbar.

Reorder the list:

To reorder the items in the menu, select one, then click the Move Up or Move Down buttons. You cannot drag items within the Tag Browser.

Add a page to the list:

Note: Pages must exist before they can be added to the menu section of the Tag Browser as new Menu Item tags. The act of adding a new menu item does not create a page to match it, nor is there a way to create the page while creating the menu item.
The act of creating a new page does prompt you to add it to the menu.

1. Open the Tag Browser to the Page Menu section.

2. Ensure that the Page Menu section is selected, or a folder within the Page menu section.
   DO NOT SELECT A PAGE WITHIN THE PAGE MENU.
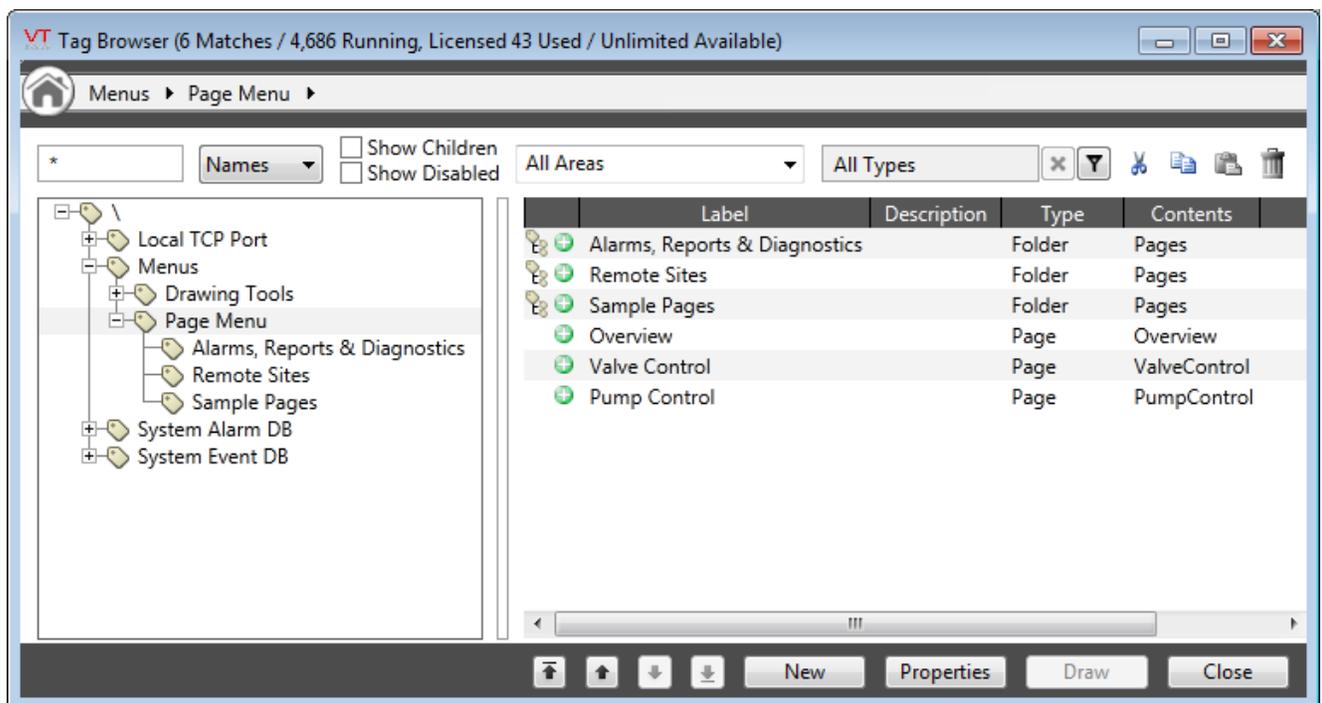
3. Click New in the Tag Browser, or right-click on Page Menu and select New Child from the menu that opens.
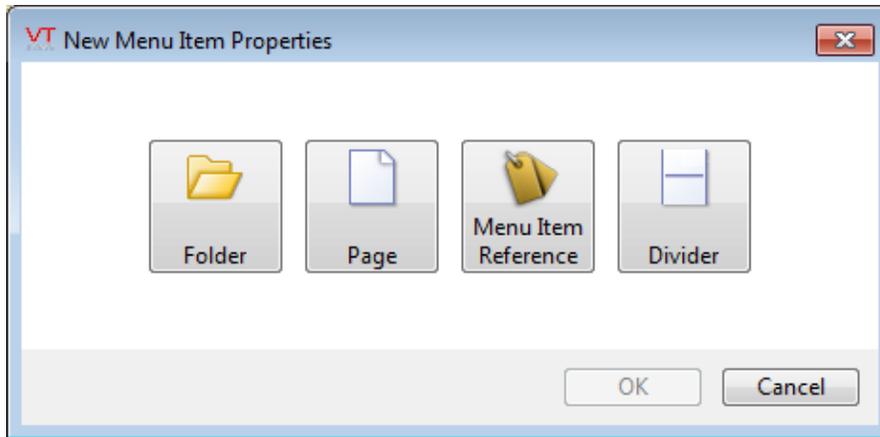
4. Click, Page

The New Menu Item Properties dialog changes to show page selection options.
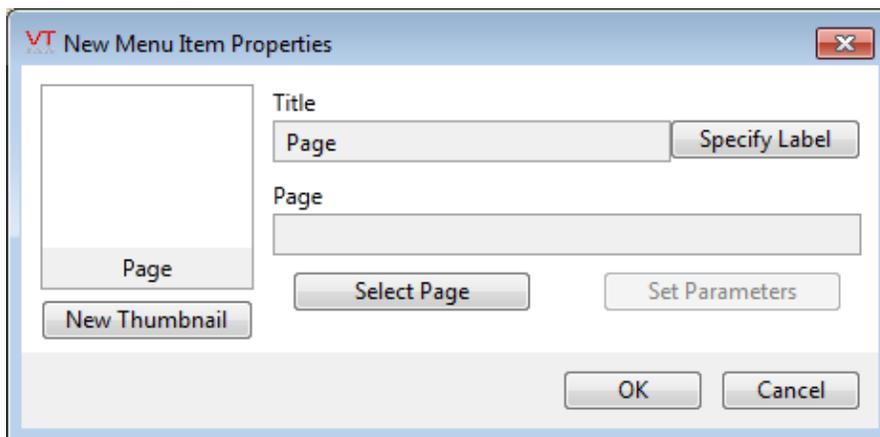
5. Click, Select Page.

6. Choose the page from the selection dialog that opens and click OK
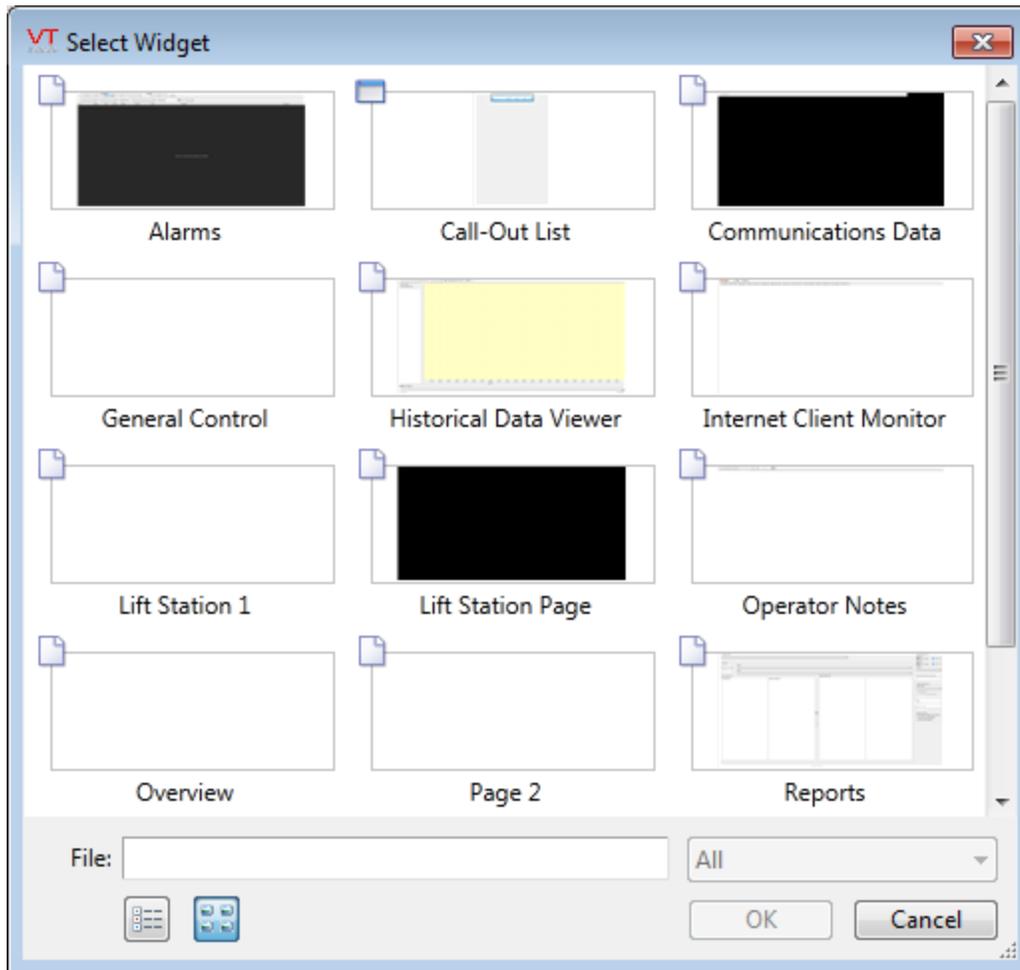
The dialog includes all the pages in your application, showing a preview of each within a tile. (Some previews may take several moments to load.)

7. If the page was defined with parameters, click the Set Parameters button to select tags or assign values for these now.

8. [Optional] Click the Specify Label button to change the title that will be assigned to this page in the menu.

A new menu item will be added to the Tag Browser, linked to your page.

9. [Optional] Use the Move Up and Move Down buttons to adjust order of the menu items.

Add a folder to the menu

Folders help you organize your page menu into fly-out or cascading menus, gathering together pages into logical groups. When using the tiled menu, you can monitor the pages within the folder on one screen. Folders can be added at the top level of the Page Menu, and can be added below other folders. They cannot be added below pages.

1. Open the Tag Browser to the Page Menu section.

2. Ensure that the Page Menu section is selected. DO NOT SELECT A PAGE WITHIN THE PAGE MENU.

   If you have already created a folder (sub-menu), and are adding another folder to it, then the folder should be selected.
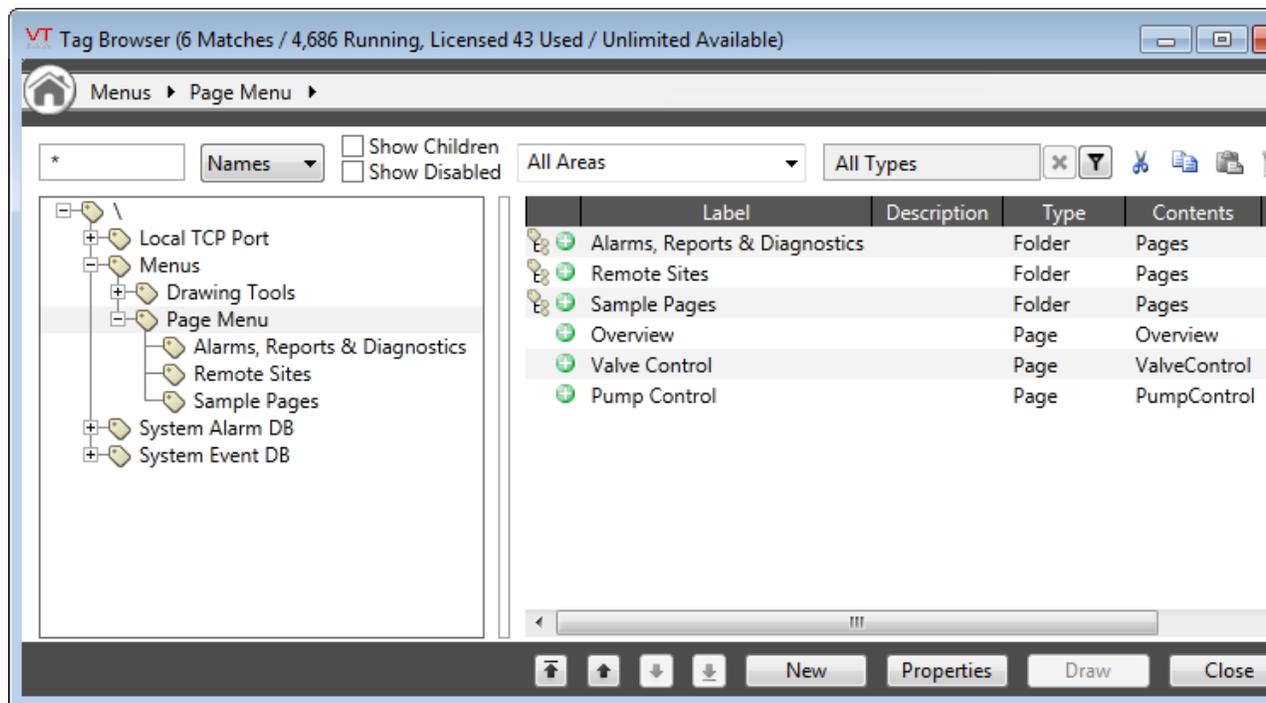
3. Click New in the Tag Browser, or right-click on Page Menu and select New Child from the menu that opens.
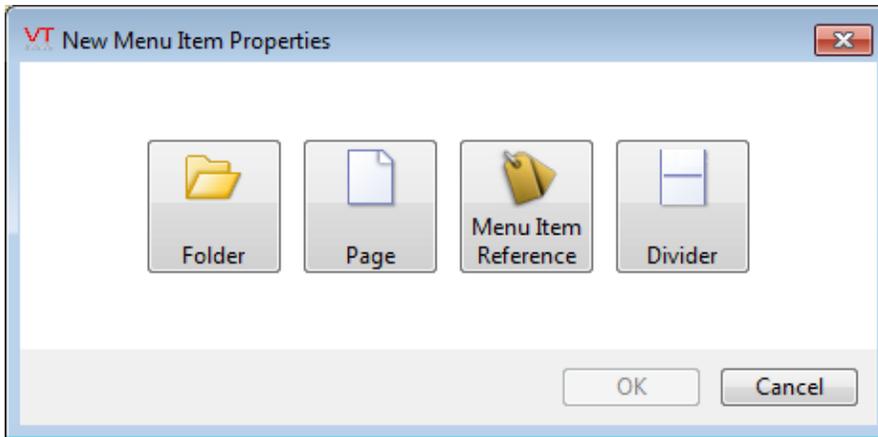
4. Click, Folder.

5. Provide a title for the folder.
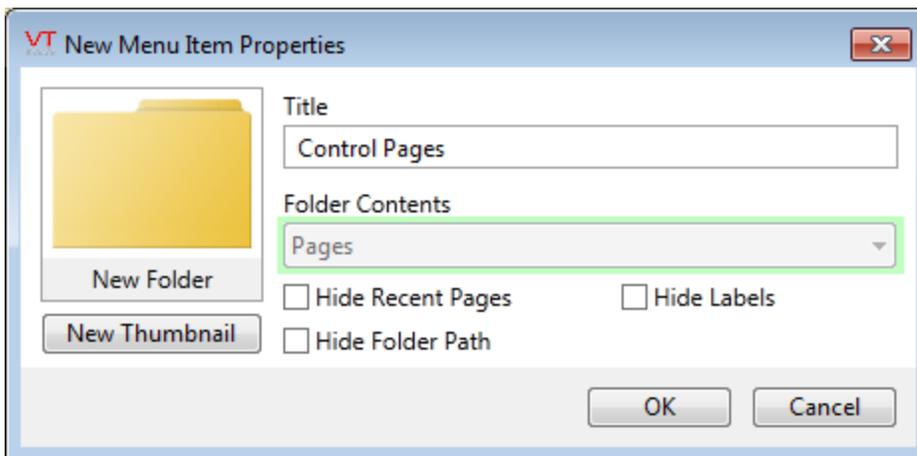
6. [Optional] If the pages that you will later add to the folder require extra space for visibility, you may choose to hide parts of the default tiled menu such as the recent pages list, folder path and page labels.
   The button, New Thumbnail, does not apply to the page menu.

7. Click OK to finish.

Add a divider to the menu

Dividers draw attention to menu items and help to visually separate one set of items from another.

1. Open the Tag Browser to the Page Menu section.

2. Ensure that the Page Menu section is selected. DO NOT SELECT A PAGE WITHIN THE PAGE MENU.

If you have created a folder (sub-menu), and are adding a divider to it, then the folder should be selected.
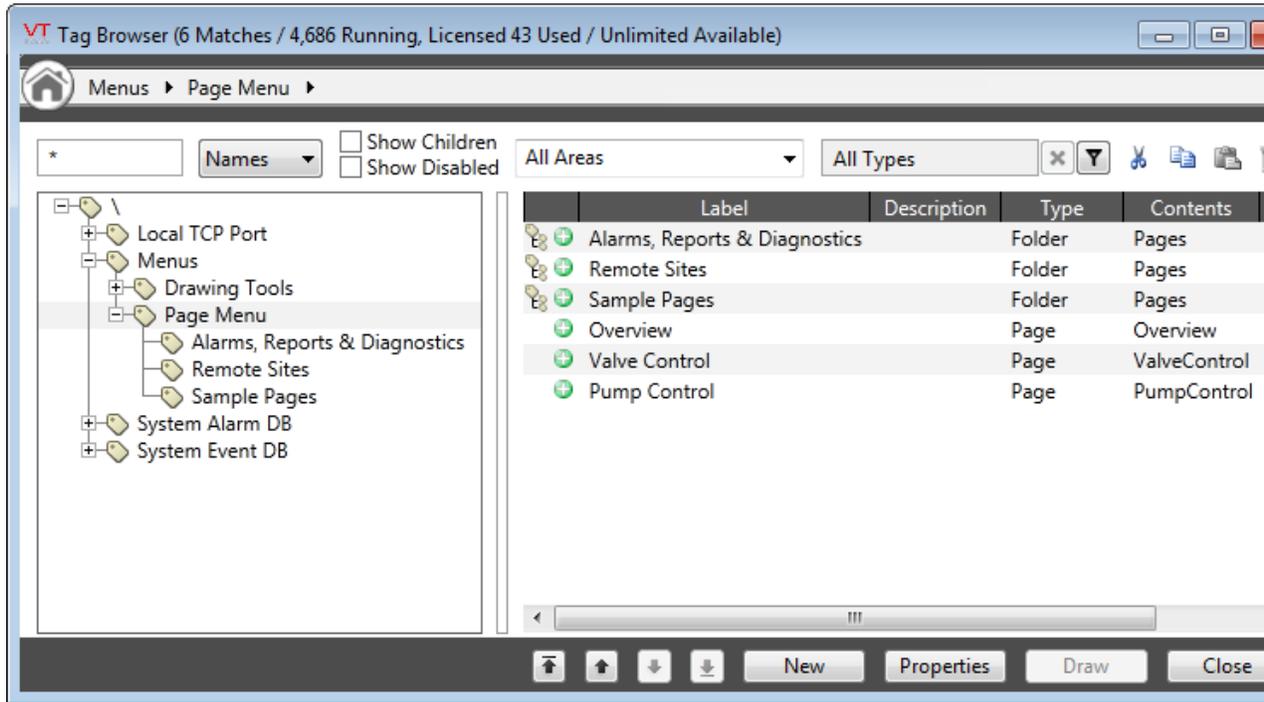
3. Click New in the Tag Browser, or right-click on Page Menu and select New Child from the menu that opens.
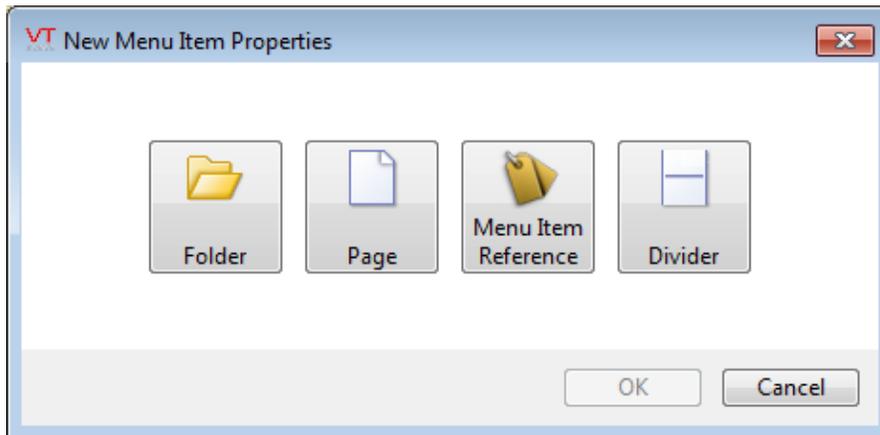
4. Click, Divider.
   Dividers have no options for configuration.

5. Click OK.
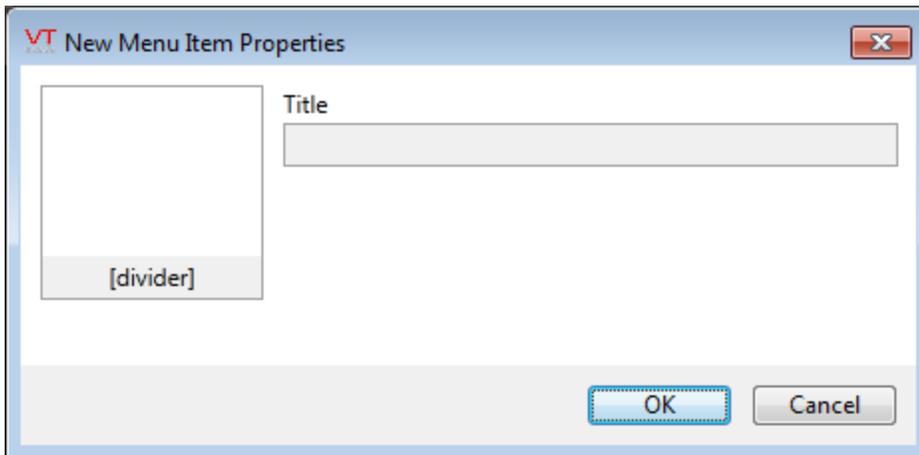
6. Click OK to finish.

Add a Menu Item Reference

A menu item reference is a copy of another menu item. This can be used when you want to link to a parametrized page using a new set of parameters.

The easiest way to add a new menu item reference is to copy and paste an existing item, then edit the properties to change the label (title) and parameters.

Add Page Hotboxes and Page Buttons.

Both tools can be found in the Standard Library folder of the Widgets palette.

1. Open the Idea Studio

2. Open the page that you want to add a hotbox or button on.
   If the page is not already open, or available for selection from the tabs at the bottom of the Idea Studio, click File, then Open.
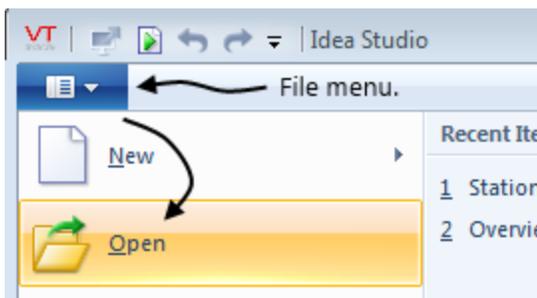
3. Open the Widgets palette

4. Find, then click to expand the Standard Library folder.

5. Drag a Page Hotbox or a Page Button onto the Idea Studio screen.

6. Adjust the size and location of the hotbox or button.

7. With the hotbox or button selected, click the Properties button in the toolbar.

8. Select the page to be opened in response to a click.

9. Adjust the appearance as desired, using the options in the hotbox.

10. If the page has parameters, select the tags or set the values to be used for this instance of the page.

11. Click OK to close the properties dialog and save your work.

Related Information:

...Navigating Application Displays – Guide to navigation, provided in the VTScada Operator's Guide

...Menu Editing Using the Tile Menu – Use the right-click context menu for page organization

...Adding New Menu Items – Reference describing each type of menu item

...Menu Item Tags – Reference

...Refer to the VTScada Admin Guide for the following properties:

...PageSnapshotCacheThresholdCount – Properties for tiled menus

...PageSnapshotCacheThresholdPct

...PageSnapshotsMaxInMemory

Next Steps:

...Build the User Interface with the Idea Studio

Menu Editing Using the Tile Menu

The advantage of the Tag Browser for menu editing is that it can provide a full overview of the entire menu structure. You can however, carry out most of the same tasks by right-clicking on the Tile Menu to open a context menu:
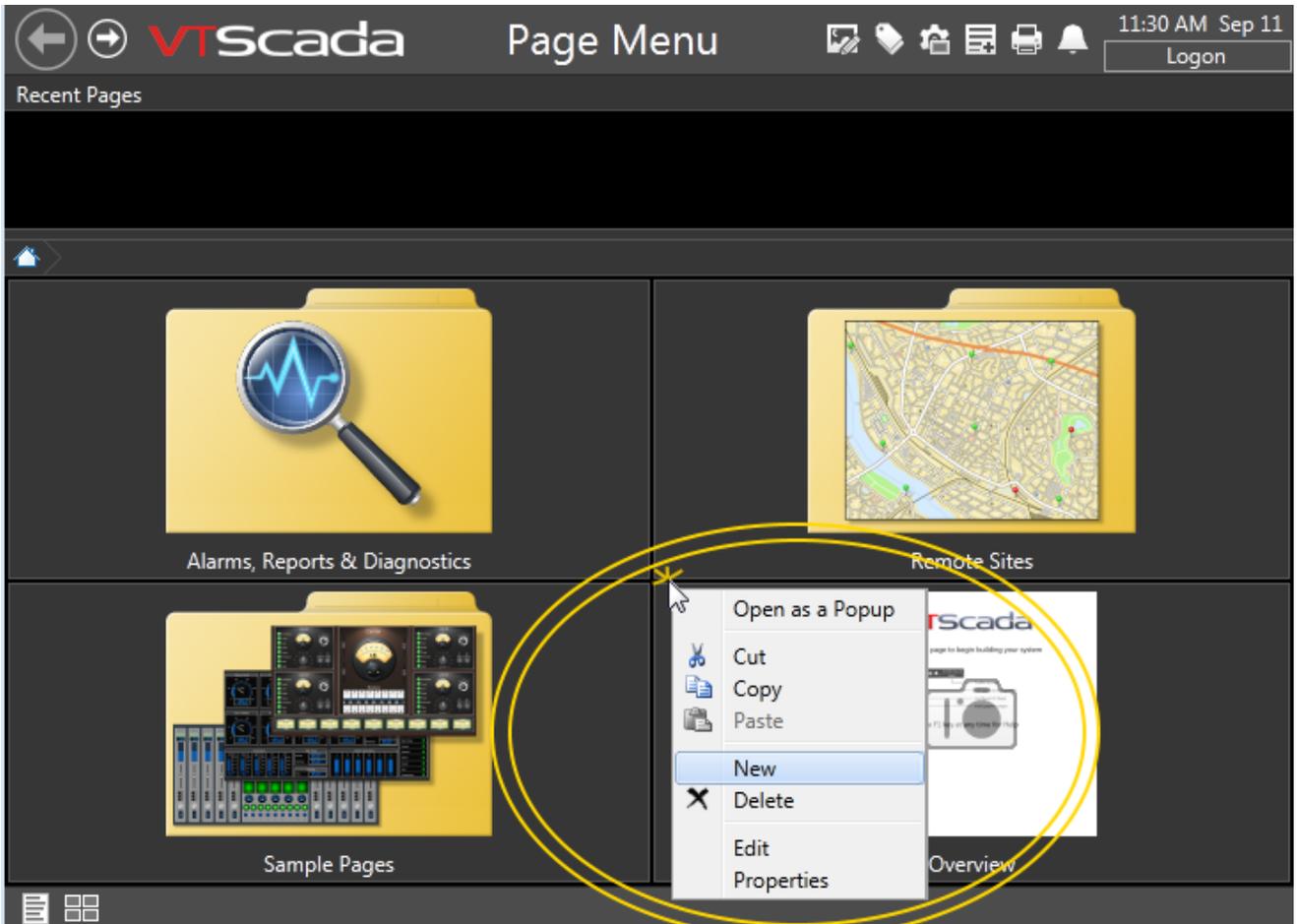
Open as a Pop-up

Applies to pages that may be opened either full-screen or as a pop-up. A left-click on the page opens it full-screen. In most VTScada menus, a right-click opens the page as a pop-up. Since a right-click here opens this context menu, the first option in the menu is to open the page as a pop-up, thus preserving the standard behavior.

Cut, Copy, Paste / Paste as Child

Use these options to duplicate or move pages (Menu Item tags) within the menu structure. You must cut or copy a page before you can paste it elsewhere.

Paste versus Paste as Child: If this context menu is opened in response to right-click on a page, the option will read "Paste". If this menu is opened in response to a right-click on a folder, the option will be, "Paste as Child". In the first case, the new menu item will be added to the same folder as the selected page. In the second, it will be added to the selected folder.

New

Opens the New Menu Item Properties dialog, to create a new Menu Item tag. (Folder, Page, Reference or Divider.)

Delete

Deletes the Menu Item tag associated with the page or folder where you right-clicked.

Edit

Opens the selected page in the Idea Studio. Does not apply to folders.

Properties

Opens the Menu Item tag's properties dialog for the selected page or folder.

Related Information:

... Define Navigation Between Pages – Menu editing using the tag Browser

...Adding New Menu Items – Properties dialog for each type of Menu Item

...Menu Item Tags – Reference

Restrict Access to a Page

You can deny access to pages for unauthorized users by applying an application privilege to the page. Only users who have been granted that privilege will be able to open the page. Further, users with configuration privileges will not be able to access the page's properties if they do not also have the required privilege for that page.

> Note: When assigning privileges: In the operator's account (or role) definition, the privileges required to open a page (or perform any other native VTScada operation) must not be limited by tag scope rules. Pages are independent of tags, therefore if the privilege has a tag-scope rule applied, the result is to deny access to the page for that account.

1. In the Idea Studio, open the page and its associated Page Properties toolbar.
2. Click the Security tool.
3. Select the privilege, which you created earlier for this purpose.
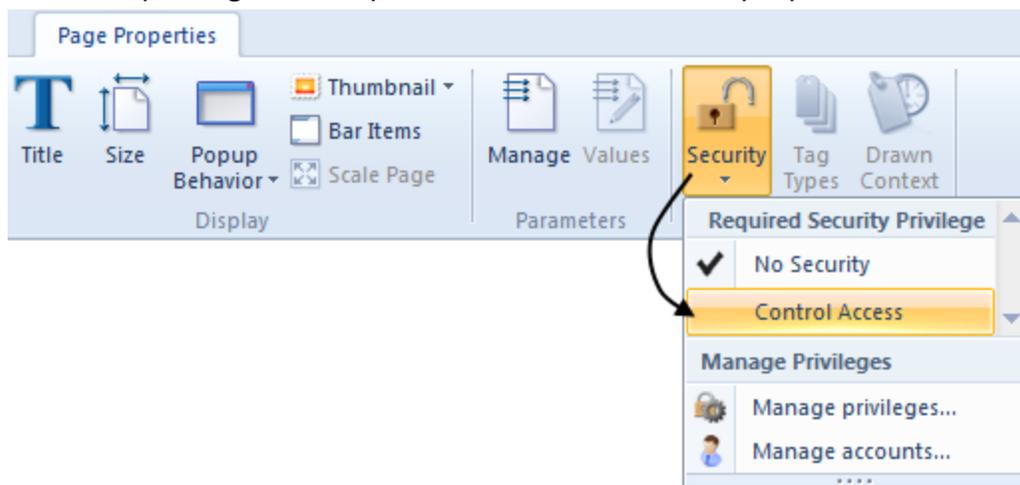
If you have not yet created an application privilege, you may click Manage Privileges to do so.

Troubleshooting:

- There is no privilege to apply.
  Create an application privilege, to be applied to the page.
- The page was open, but closed immediately after the privilege was selected.
  Your account does not possess this application privilege. You will need that privilege in order to open or make any further changes to the page.

- Operators have been granted the matching privilege to open the page, but cannot do so.
  Ensure that the privilege has not been limited by a tag-scope rule in the operator's account or the role that provides the privilege to the operator.

Related Information:

...Refer to the VTScada Admin Guide for "Accounts and Roles" – Creating and managing. Also, "Protect Pages and Output Tags" – Create your own privileges for protection.

Draw Widgets

Most widgets are linked to tags, providing either operator control for output tags, or a visual indication of tag values. They can have the form of buttons, switches, animations, text, meters, etc.

Note: Unlike images, widgets have a "Link" option to tie them to a tag.

Because developers are able to draw widgets first, then link those widgets to tags, it is possible for unlinked widgets to be left in a page. These will show simulated values.
To avoid the risk of simulated values being mistaken for real process data, the unlinked widget symbol is drawn on all unlinked widgets. If you see this symbol, know that the values being displayed are coming from a simulator, not from equipment.

Note: It is possible to turn off the display of the unlinked widget indicator for individual widgets. If this has been done, a warning indicator will blink orange in the title bar. Developers with configuration privileges can

click that indicator to toggle the individual indicator display on and off
for all operators.

Within the palette, widgets are grouped into folders according to the
type of display they are, and according to the types of tag they can be
linked to. Each widget is likely to be found in several folders.
Many of the newer widgets can be linked to a Style Settings tag. This
provides a single point of configuration where colors can be assigned to
tag states (running, stopped, alarm, LCD component, etc).

To add a widget to a page:

1. Open the Idea Studio and open the page where the widget will be drawn.
2. Open the Widgets palette.
3. Click on the folders to view and find the widget you want.
   Folders are organized by widget type, and by tag type. Navigate using the
   system that you prefer.
   Use the navigation bar at the top of the folder list to return to a higher level
   folder, including the "Home" folder.
4. Drag a widget from the palette to your page.
   Alternatively, click once on a widget to attach it to the cursor, then drop it on
   the page.
5. Adjust the size, location and other properties of the widget as required.
   A format ribbon is available for each widget type.
   Advanced configuration details can be accessed by opening the Properties
   dialog of the widget.
6. Link the widget to a tag as follows:
   a. Select the widget.
   b. Click the Link button, either in the widget format toolbar or in the
      right-click context menu.
   c. From the tag browser (opened by the Link button) choose the tag this

widget will represent and click Select.

    d. If the tag does not already exist, create it.

Troubleshooting:

- The widget is too big / too small.
  All widgets have an initial size, which may be wrong for your application. After the widget has been placed, zoom in or out on the canvas until you can see the edges easily, then select the widget and drag the corner grips to resize it.

- The widget is cycling through values / colors all by itself.
  Widgets display simulated values until they are linked to a tag.

Related Information:

...Tag Reference – Full reference of all types in VTScada

...Widget Reference – Full reference for the configuration of all widgets.

...Panels – View and edit links between widgets and tags (or other data sources.)

...Style Settings Tags – Define a color scheme to represent equipment operating states.

...Design Your Own Widgets – Create new and better ways to display information.

...VTScada Graphics Library – Provide context to the widgets by including relevant images.

...Draw Shapes – Provide context to widgets using geometric building blocks.

...Ribbons – A description of the tools in each formatting ribbon.

Next Steps:

...Select and Group – Before changing the appearance of an object, you must first select it.

...Align – Tools and techniques for well-organized pages.

Draw From the Tag Browser

You may prefer to draw tags directly from the Tag Browser, rather than drag widgets from the palette then link them to a tag. Either method will have the same result.

Preparation:

- Ensure that the page you want to draw the tag on is displayed.

To draw from the tag browser:

1. Open the Tag Browser and select a tag.
2. Click the Draw button, or right-click on the tag and select Draw from the menu.
   If the Idea Studio is not open, it will immediately open to the same page that you are viewing in the main window.

   A dialog will open, showing the possible widgets that can be used to represent this type of tag. If another tag of this type has been drawn recently, the same widget folder will open automatically for this tag.
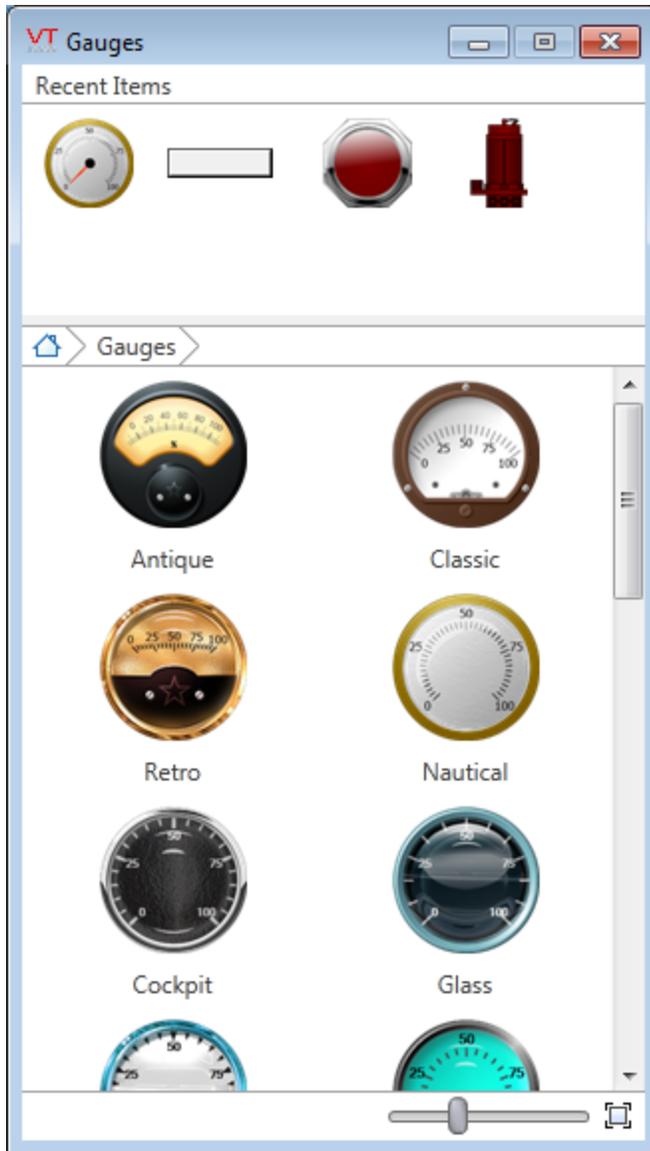
Example shows gauges for drawing an Analog Status tag.

3. Navigate to the folder containing the widget you want to use, if it is not already open.

4. Drag the widget you want to the page.
   Alternatively, click once on the widget you want to draw – an instance will attach itself to the cursor. Click on the page where you want to place the widget.

The Tag Browser remains open, allowing you to draw the tag again using another widget, or to draw other tags. Close it when you have finished.

Troubleshooting:

- Accidental click on the wrong widget, drawing it instead.

Delete it and draw again.

Link a Tag to a Widget

Most widgets exist to display tag values, or to provide operator controls that write to tags. For either purpose, the widget must be linked to a tag. Links can be changed at a later time if you decide that a particular widget should be linked to a different tag of the same type. (For example, if you copy a widget for use by a new tag.)
There are two ways to link a widget to a tag:

- Selecting the widget in the Idea Studio, then click the Link button in the Widget Properties ribbon
- Open the Properties dialog for the widget, then click the Tag Browser button found there.

Both methods open the Tag Browser, where you can select a tag for the link. Note that two or more widgets can be linked to the same tag in one operation, providing that all selected widgets can be linked to the same type of tag.
Widgets can also be linked to parameters, as is the case in a parametrized page.

Related Information:

...Selecting a Data Source – Specific Tag, Linked Tag (for widgets), Parameter, or Expression

...Panels – View and edit links between widgets and tags (or other data sources.)

Selecting a Data Source

Widgets require a data source to provide the value that is being shown. Most commonly, the data source for a widget will be a tag's value, but this is not the only option. You can also select page parameters or (in the case of custom tag widgets) properties of the associated tag. (See image.)
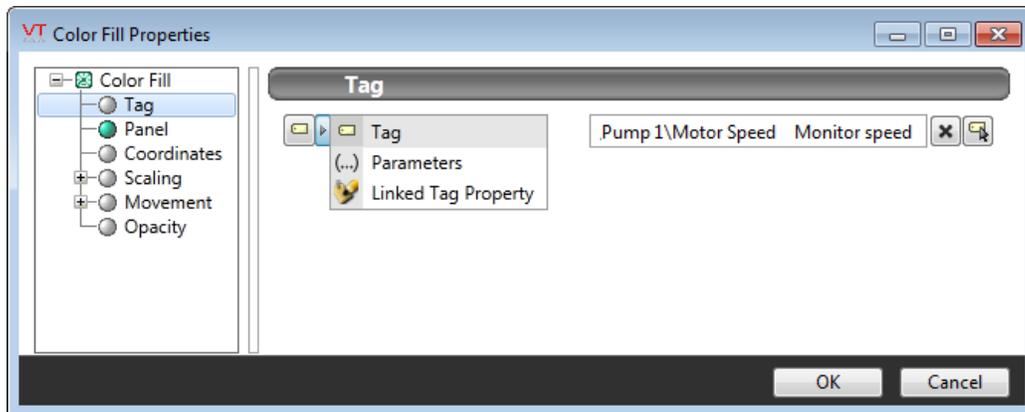
The differences between the choices are as follows:

Tag:

How: Use the Tag Browser to select any tag whose value can be represented by this widget.

Why: This method creates a one-to-one relationship between a tag and an animation on a page. For uncomplicated applications, this is the easiest way to draw tags.

Parameters:

How: The tag selection area is replaced by a parameter selection. From this control you can select any page parameter (or descendants of that page parameter) whose type has a value that can be represented by this widget.
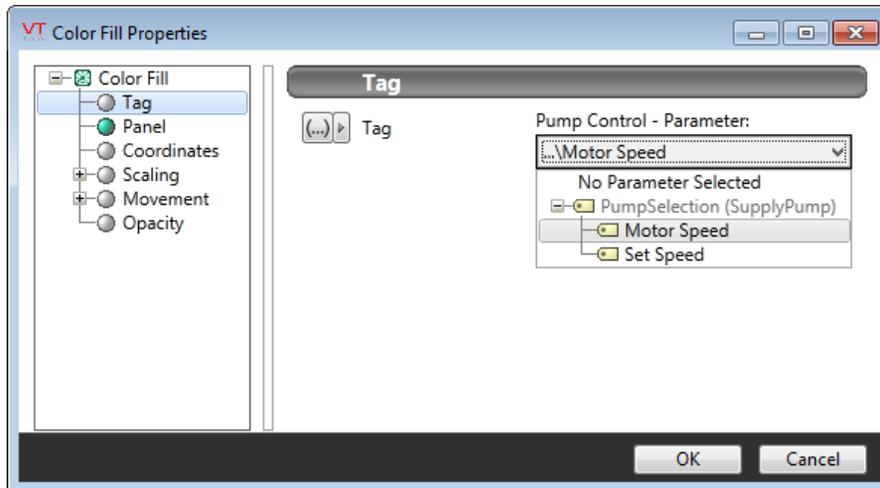
Child tags with a blue dot are those that are not part of the standard parent-child type definition. They exist (at least) in the instance of the type, selected for the parameter when starting an editing session. If you select a tag with a blue dot here, you should not expect it to exist in every instance of the type. **Why**: The same page can now be used for more than one tag. For example, if your application has several identical pumps, tanks, stations, or other equipment. In each button or hotbox that is used to open the page, you can use a different set of tags for the parameters.

Drawn Tag Property:

How: Enabled only when the widget is one that you have created. Select any of the properties of the tag instance being drawn. The selected property must have a data type that can be represented by the widget.
Why: Similar to using a parameters, this provides a way for the same object can be re-used for different tags. This technique is most often used when you have complex drawing objects, showing all the tags that are part of a complex tag type you have defined.

Related Information:

Choices for Operator Interaction

Every tag widget has been designed to interact with the operator when the object is clicked upon or pointed to. Depending on the tag type, one or more of the following three actions will be available: Trends, a Navigator Menu, and Tooltips.

Trends. For numeric types, a left-click on the widget will cause the Historical Data Viewer (trend) window to open for the associated tag. If the click lands on several overlapping widgets, all the associated tags will be plotted in the HDV.
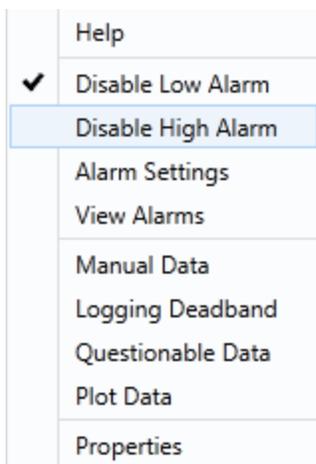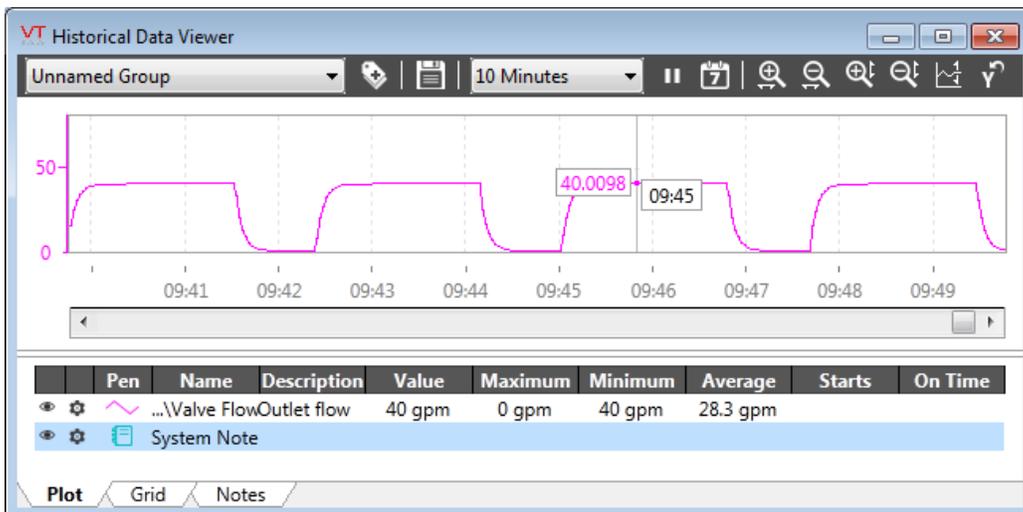
Navigator menu

For all tags, a right-click on the widget will cause a context or "navigator" menu to open. The contents of this menu will vary by tag type – the image shown here is a sample. The attached tag in this case is an Analog Status type, with two built-in alarms, and the ability to have Manual Data, a Logging Deadband, and the Questionable Data flag.

In all cases, there will be a Help option at the top, opening the relevant page in this reference for the associated type. A Properties option will be found at the bottom of the menu, allowing you to re- configure the attached tag.

Tooltip
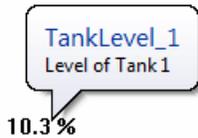
Hovering the pointer over any tag's widget will cause a tool tip to be displayed. This enables the operator to see the tag's name and description.

Disable Operator Interaction Tools

Any of these tools can be disabled in a Widget. Note that this is done on a Widget-by-Widget basis, as shown in the following screen-shots, taken from the properties dialog.

Two options, Trends and Tooltips, can be disabled globally using application settings.
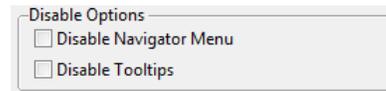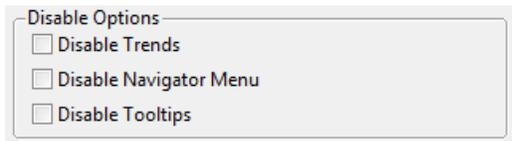
Related Information:

Refer to the VTScada Admin Guide for the following properties:

...AITrendEnable. Control access to the trend pop-up for all analogs.

...DITrendEnable. Control access to the trend pop-up for all digitals.

...ShowTip. Enable or disable all tool-tips.

Design Your Own Widgets

Anything that you have drawn can be grouped into a new, custom widget. This can be as simple as a grouping used to make alignment easier on a page, or as complex as the components of the MultiSmart and MPE Site pages.
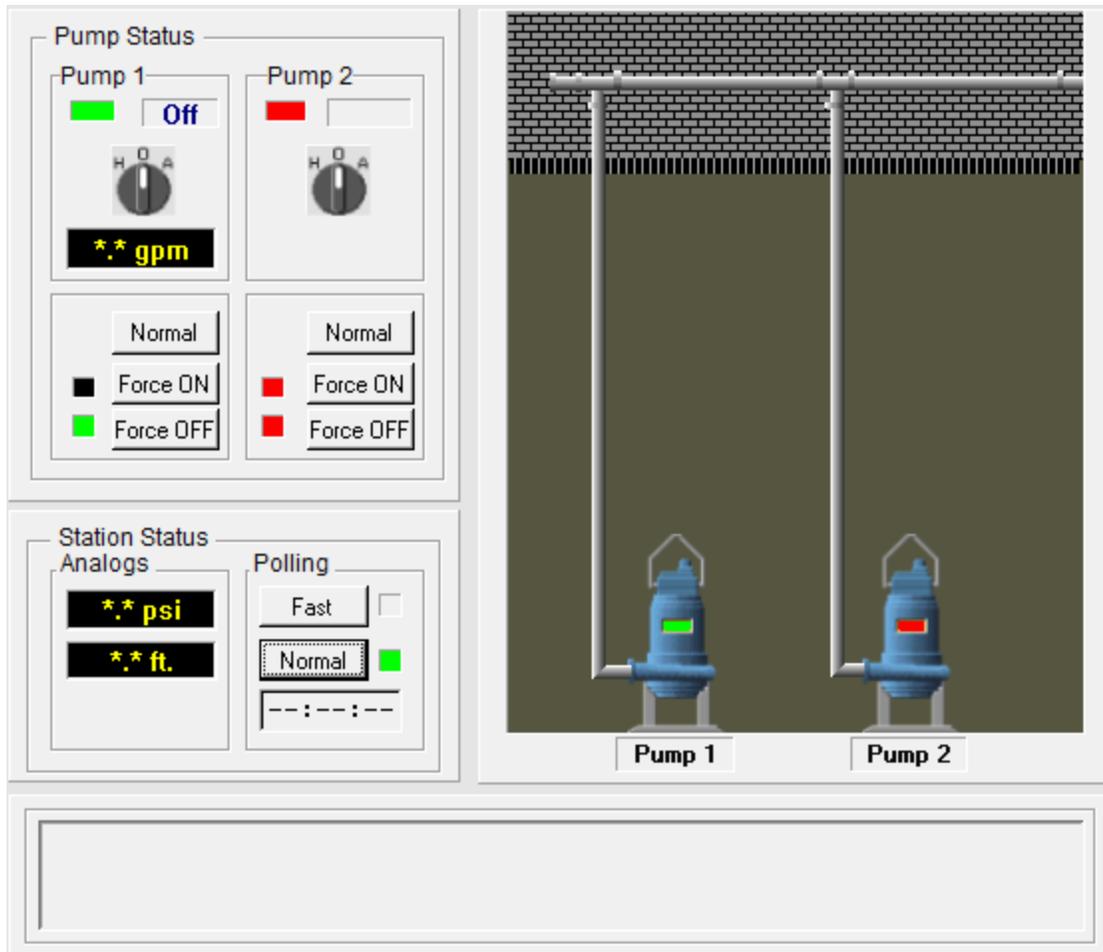
A single widget, with many components.

There are three classes of custom widget, listed in order of increasing features and capabilities:

Group	Elements joined together for the purpose of alignment and positioning on a page. Named automatically. Cannot be added to a palette. Cannot be linked to a tag, even if the elements within the group are tag-linked widgets.
Library Widget	Elements may include one or more existing widgets in addition to shapes or images. Must be given a name. Automatically added to the palette. Tag links, if any, are done through parameters rather than directly to the widget. Does not become a native widget for any tag.
Tag Widget	At least one widget must be included in the selection set. All of the selected widgets must be linked to tags.

	Must be given a name. Automatically added to the palette. Becomes a native widget for one or more tags. If the component widgets are designed to be linked to the child tags of a custom tag type, all children will be linked automatically upon linking to the parent.

When creating either a library widget or a tag widget, you will see the New Widget dialog.
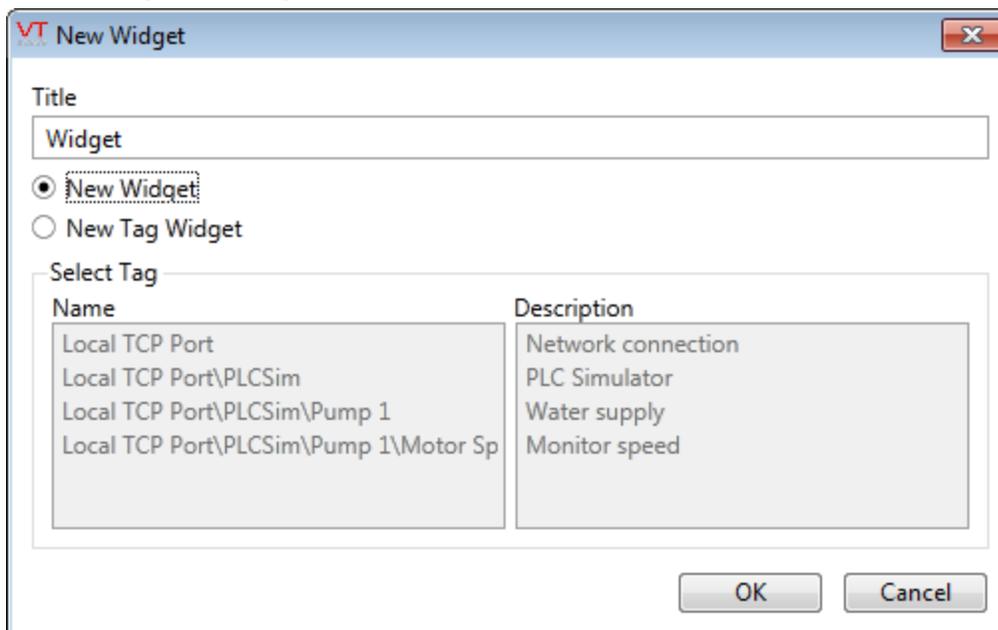
Instructions for using this dialog are provided in the subtopics, "Library Widgets" and "Tag Widgets".

Note that new widgets will be added to the top level of the Widgets palette, and also to the list of most recently used widgets. Each time that you create a new widget, you should take a moment to edit the palette, moving the menu item into a folder of your choice. By keeping the palette organized, you will save time later when looking for a widget that you want to use.

Any individual widget can be broken apart into its components by selecting the instance, then clicking "ungroup" in either the formatting toolbar or the right-click menu. Note that, if your intent is to change the appearance of either a widget, you should edit the widget definition, rather than following a process of ungroup – edit – regroup.

Widget definitions can be opened for editing the same way that a page is opened. Use the Open command in the Idea Studio's file menu to open

the widget by name, or select any instance in a page and click Edit, either in the toolbar or the right-click menu. Your changes will affect all instances of the widget or group throughout the application.

A widget can be given a background color but this will be displayed only in the Idea Studio where the widget is being edited, and in the palette. When drawn on a page, the widget's background color will not be included.

Application security privileges cannot be selected within widgets. To protect output controls, apply the security privileges to output tags or to pages.

Parameters in Widgets

Library widgets will automatically have one parameter for each tag. Other parameters may be added as required.

Tag widgets will have three parameters, corresponding to the three standard operator controls available in all VTScada widgets: DisableTrend, DisableNavigation and DisableTooltip. The default values for these are set in the widget's source code, where they may be edited by advanced developers.

The properties dialog for any page or widget that has a text parameter will provide basic text-entry field as shown:

If this parameter is meant to be used for color selection, then you should edit the code for that parameter as follows.

Original parameter definition:

```
(
  MyColor <:"Color Choice":> Text;
)
```

Modified parameter definition:

```
(
  MyColor <:"Color Choice":> Text<Color>;
)
```
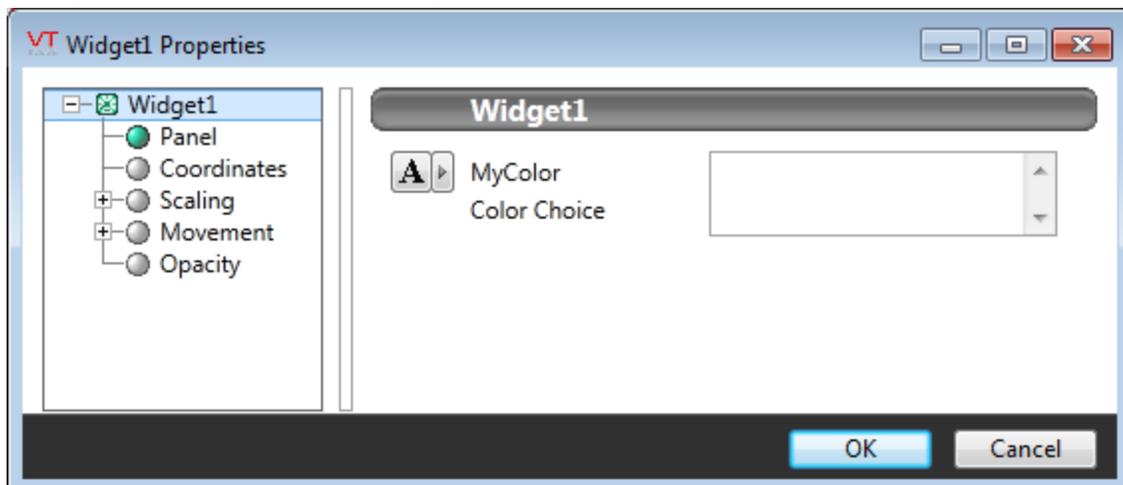
The addition of the <Color> modifier after "Text" tells VTScada to use a color picker rather than a text entry field in the properties dialog.
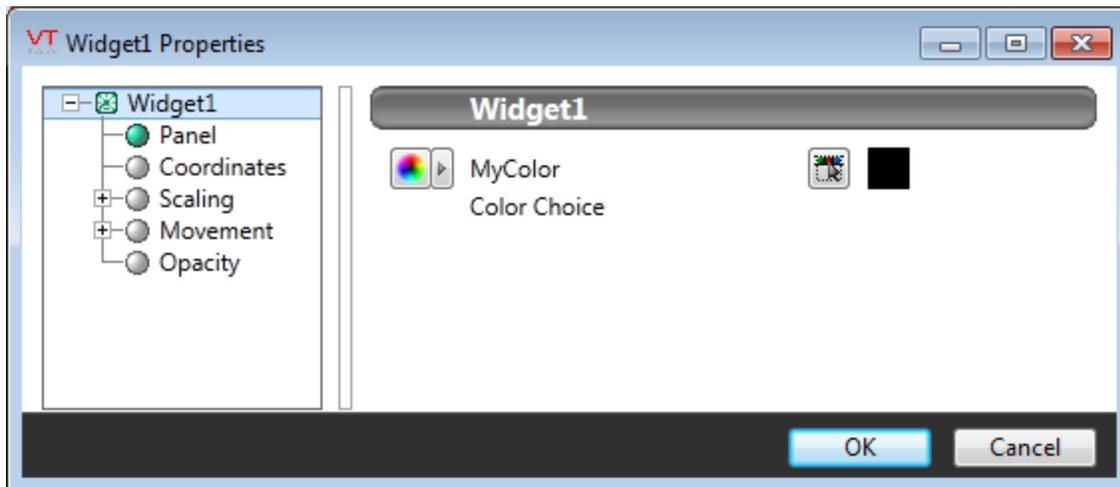
Related Information:

...Create a Group – Notes and instructions for groups.

...Library Widgets – Detailed description of library widgets. Related tasks can be found in sub-topics of this link.

...Tag Widgets – Detailed description of tag widgets. Related tasks can be found in sub-topics of this link.

...Custom Meters – How to create your own meters using the components provided.

...Import and Export Widgets – Use widgets created in other applications.

...Delete a Widget – Remove a custom widget.

...Editing the Palette MenuItem Tags

Create a Group

Groups are collections of objects that can be treated as one for the purpose of alignment and other editing operations. For all purposes, they are considered to be widgets, but are intended for temporary use only while arranging a page.
Groups are named automatically for you, as "Group1", "Group2", ...

Six separate objects.

Aligning these with the top would not work well.

Two groups of objects

Top aligned
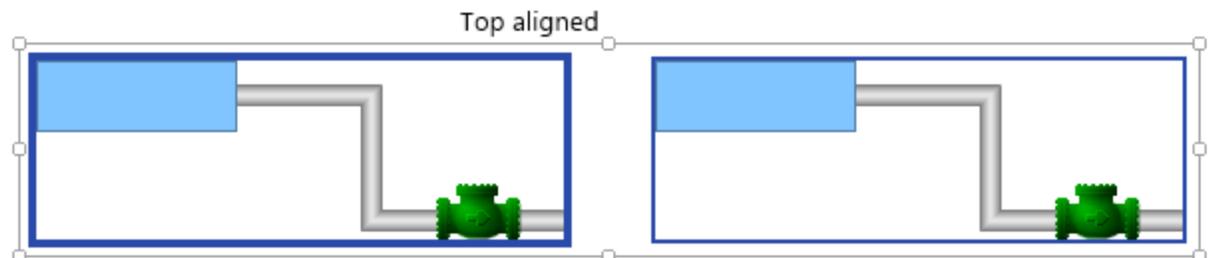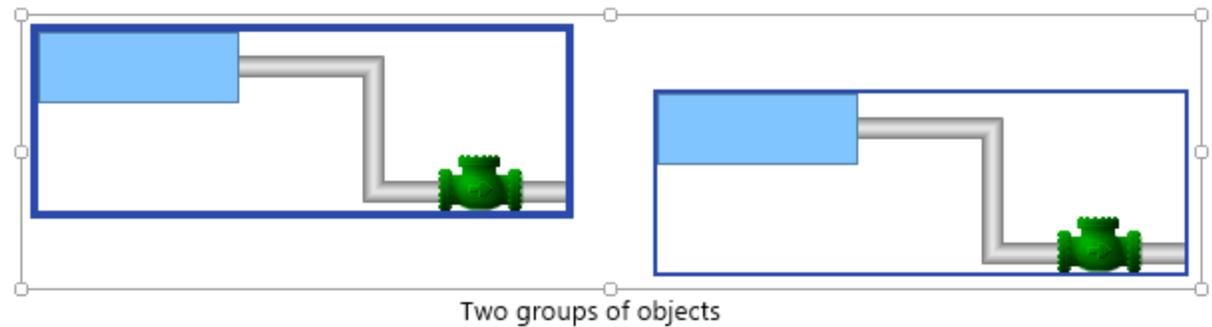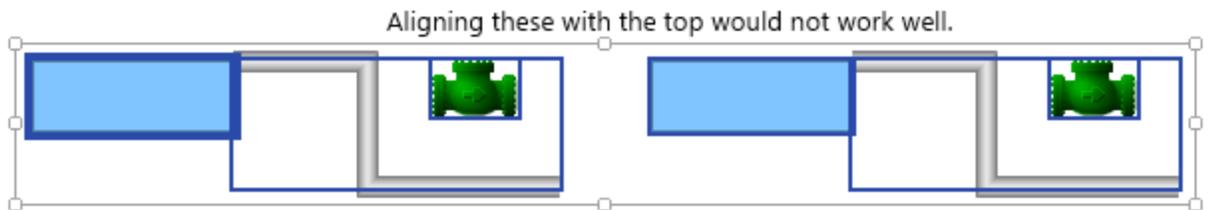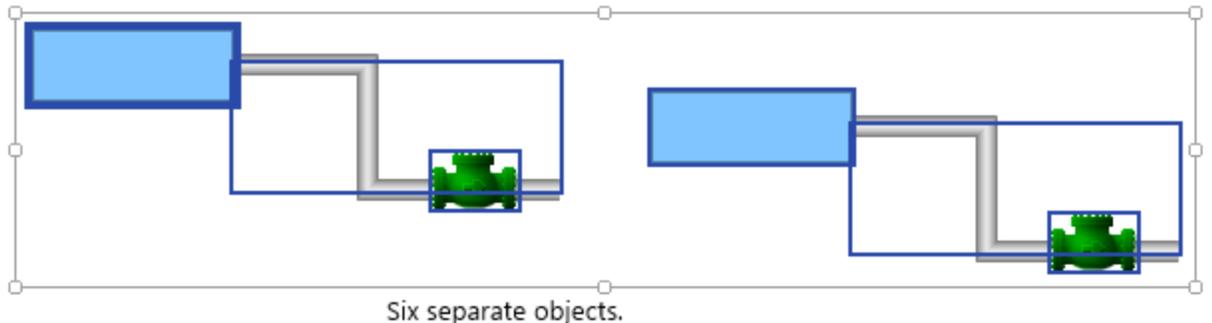
Alignment of objects, showing the difference that a group can make.

To create a group:

1. Select the objects that are to be included in the group.

2. Expand the Group option in the toolbar and click "Group".
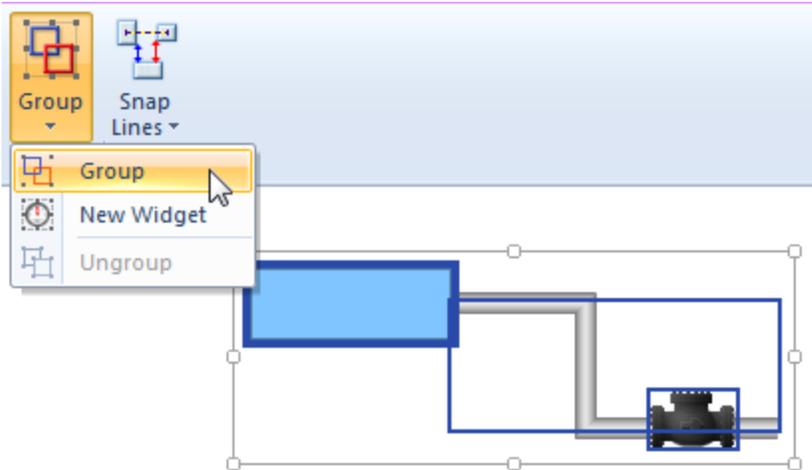
You could also right-click on any member object, and click "Group" from the menu that opens.

Related Tasks:

...Select and Group

Library Widgets

A library widget differs from a group in two ways:

- You can provide a name of your choice for the widget.
- It can be added to a palette.
- Instances can be linked to tags of your choice through parameters, provided that at least one of the components is itself a widget that can be linked to a tag[1]. In this, a library widget is similar to a parametrized page.

[1]In general, if your intention is to link a specific tag, or a parent-child tag structure to the widget, you should create a tag widget rather than a library widget. Use library widgets for extending the list of shapes in your application or for widgets that must be linked to two or more unrelated tags.

Create a Library Widget

Prerequisites:

- If all objects in the selection set are tag-linked widgets, you are strongly advised to add a bounding box or other shape. The widget you are creating will not be visible in the palette otherwise.
- All widgets in the selection set must be linked to tags before they can be included in the new library widget.

To create a library widget:

1. Select the objects that are to be included in the widget.
2. Expand the Group option in the toolbar and click "New Widget".
   The New Widget dialog opens:
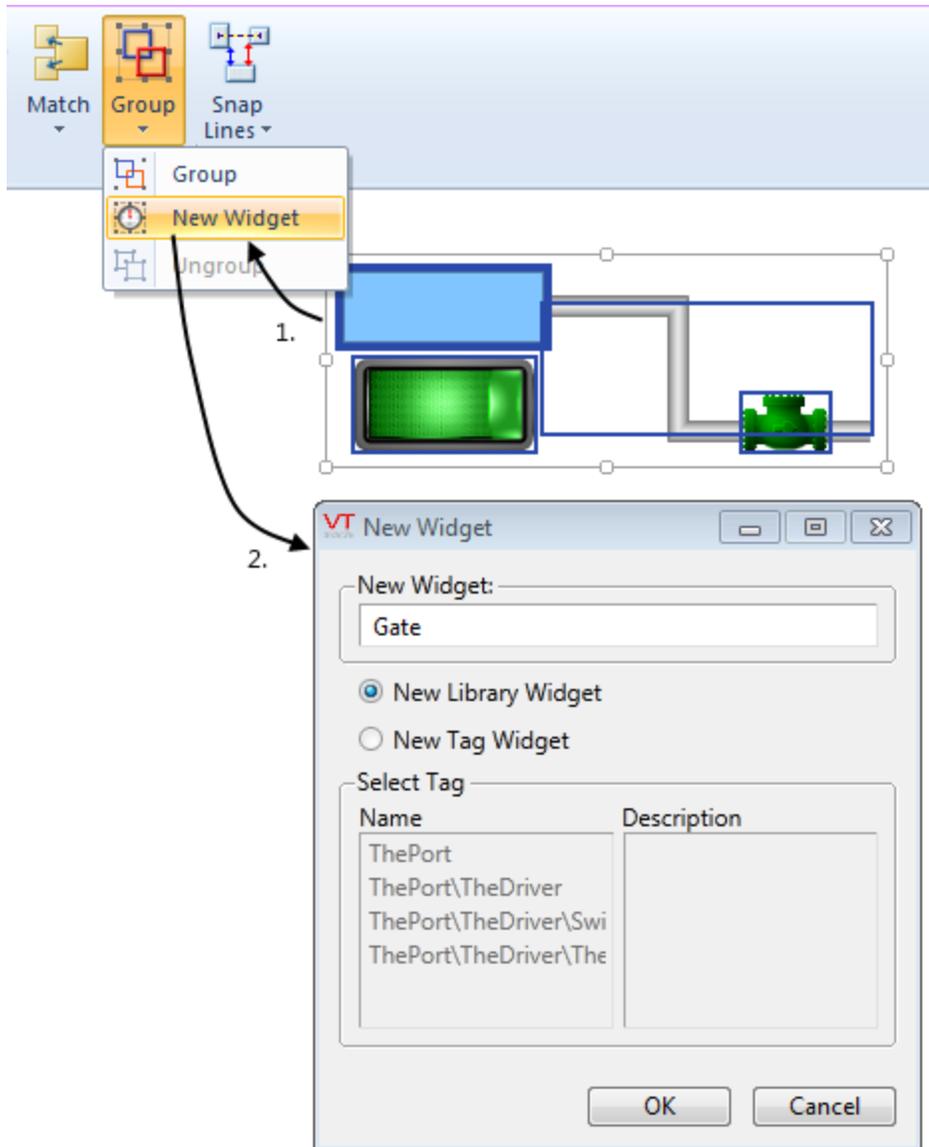
3. Select the option "Library Widget".

4. Provide a name.

5. Click OK.

There will be two results:

- The widget will be added to the Widgets palette, at the top level.
  Use the menu editor if you wish to remove it from this level and add it to a different folder.

- Any tag links to widgets within the original selection set will be replaced by parameters.
  When drawing new instances of this widget, you will need to open its

properties dialog and specify the tags to use for each parameter in each new instance.

Troubleshooting:

- An error message says that widgets must be linked to tags before proceeding.
  All widgets in the selection must be linked to tags. This includes widgets from the Equipment folder of the Widgets palette. Examine the properties of each component in the selection set to determine whether it is a widget and whether it has been linked to tag.

- In the Widgets palette, the tag-linked widgets of this new library object are invisible.

- The new library object has a label in the palette, but is invisible.
  The last two are the same problem, but this is how library widgets work. Within the palette, there is no tag selection for the parameters used by those tag-linked widget, and therefore they are not visible.

Draw a Library Widget

A library widget is drawn just like any other widget: Drag it from the palette to the page.

The one complication is that, if the library widget includes tag-linked widgets, then there will be a parameter for each. *After dragging to a page, you must open the widget's properties dialog and select the tag to use for each parameter.*

As a guide, each parameter will show the name of the original tag that it replaced within the widget. The tag selection will not be filtered for matching types unless you select appropriate types through the Manage Parameters dialog.

The Manage Parameters dialog is available in the Idea Studio ribbon, when the widget has been opened for editing.
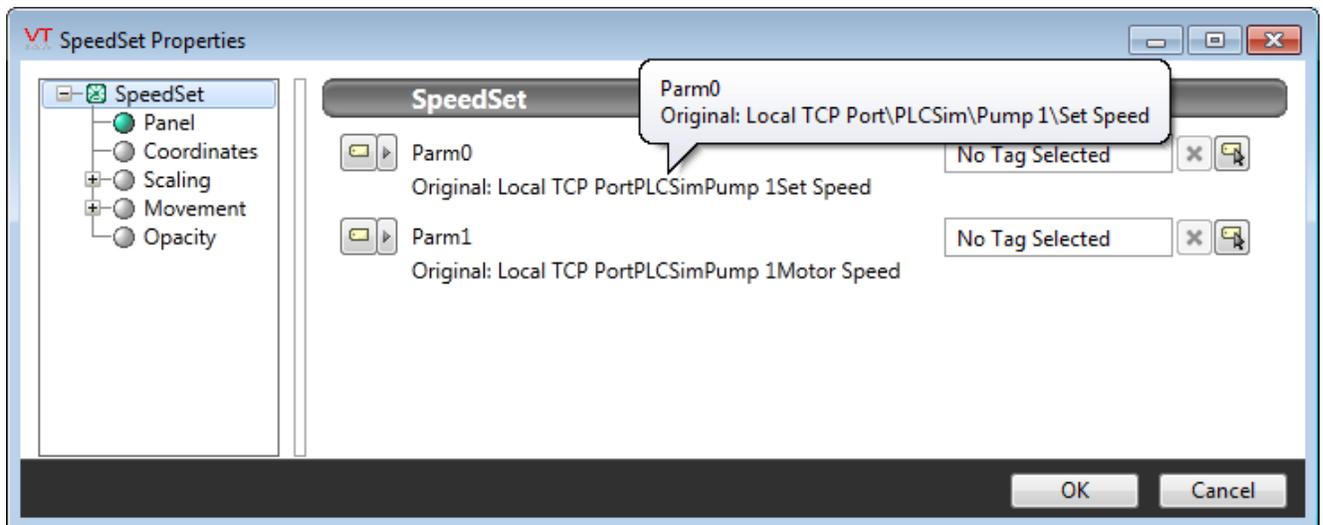
Troubleshooting:

- Parts of the library widget are invisible.
  Either, tags have not been selected for the parameters used by those tag-linked widgets, or the wrong type of tag was chosen.

Related Tasks:

...Adding New Menu Items – Organize your library widgets within the palette menu.

Related Information:

...Page Parameters & Widget Parameters – Access the dialog boxes used to configure parameters in your library widget.

Tag Widgets

Everything in the Widget palette of the Idea Studio is a tag widget. VTScada makes it easy for you to create your own, extending the options available to create a user interface that is customized for your industry or application.
Tag widgets have the following features:

- Tag widgets are designed to be linked to one or more types of tag. They are native representations of tag values, available when you click "Draw" on a tag selected in the Tag Browser.
- The tag that a tag widget is linked to may be a parent tag. When properly con-figured, all of the child tags will be assigned to components of the widget

automatically. This enables you to draw an entire station or complex equip-
ment with a single widget.

- Tag widgets do not have parameters to assign.

Example:

TankLevel
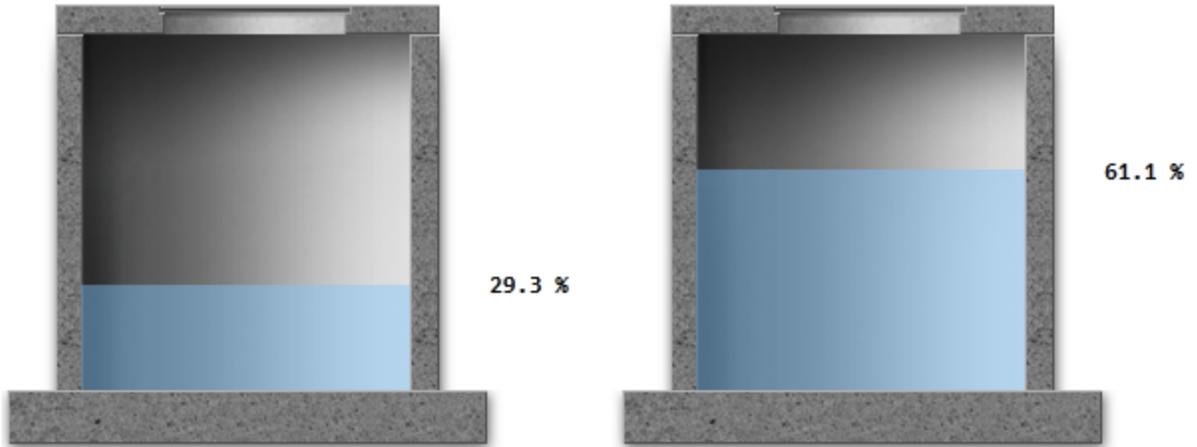
29.3 %

61.1 %

This tag widget, TankLevel, is designed to represent analog values as a
tank filling with water. The numeric display rises with the water level.
TankLevel has become a native widget[1], for use by any Analog tag:
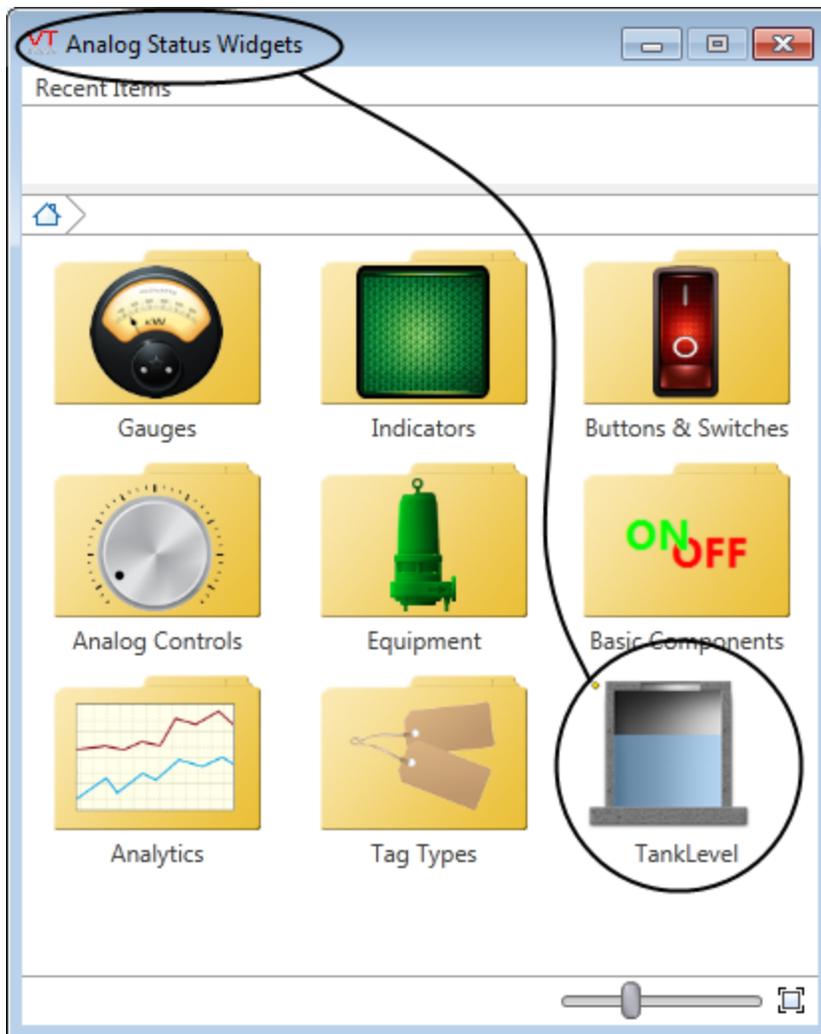
((1)Instructions to build this widget for yourself are provided in the topic, Create a Tag Widget. *TankLevel is not included with VTScada.*)

For certain types, including Alarms and various calculation types, if you have created a custom tag widget and are drawing the tag from the Tag Browser, you will find your custom widgets in a folder labeled "More...", rather than at the top level as shown in the preceding example.

In many cases, you will find that the easiest way to create a new widget is to draw it first, link the component widgets to example tags, then group the pieces. Further editing can be done if required.

It is possible to create a blank widget, using the same technique as adding a new page in the Idea Studio. If you do so, selecting Tag Widget

as the type of widget, you will be prompted to select the types of tags that you want to eventually link to your widget after it has been created.
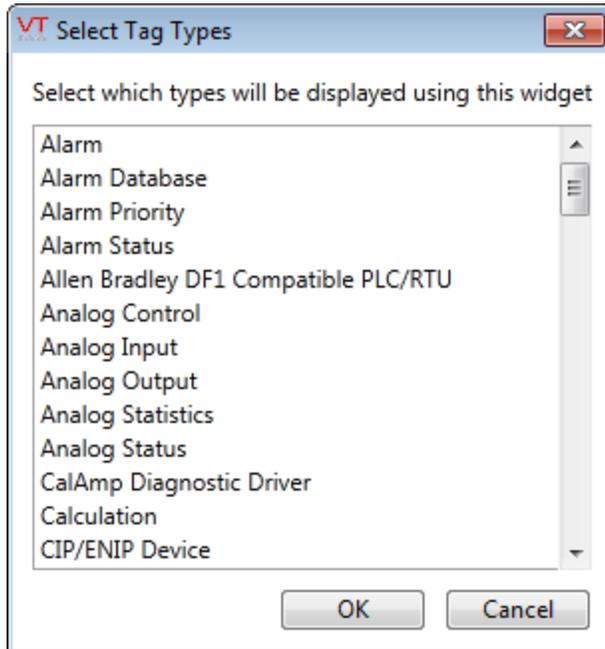
Related Information:

...Tag Icon Marker

...Linked Tag Properties

Related Tasks:

...Create a Tag Widget For One Tag

...Create a Widget For a Tag Structure

...Select Types for a Tag Widget

Create a Tag Widget For One Tag

These instructions are for creating a widget that will represent only one tag at a time. But, just as a meter can be used for many types of tag, your custom tag widget can be configured so that you may link it to any one of many types of tag.

To create a tag widget that can be linked to one tag:

1. Choose a tag and draw it using one or more widgets.
   Alternatively, draw one or more widgets and link them all to the same tag.

2. Add contextual information such as images, pipes, text and shapes.

3. Select the components that will make up the widget.

4. In the ribbon, click Group.
   You could also right-click on one of the selected objects and click Group from the menu that opens.

5. Select New Widget as the type of group.
   The New Widget dialog opens.

6. Provide a name for the new widget, replacing the default word, "Widget".

7. Ensure that the option, New Tag Widget, is selected.

8. Select the tag that was linked to the component widgets.

9. Click OK.

Results:

- The original objects are removed and replaced by an instance of the new widget. The yellow diamond in the upper left corner is a Tag Icon Marker, which may be used to indicate questionable or manual data. It can be removed from the widget if you do not want to make use of this feature.

- The new widget has been added to the Widgets palette. You are advised to edit the menu to store your widgets in a sub-folder rather than adding them all at the top level of the palette.

- As you drag instances of this widget onto pages, you can now link them to any instance of the same type of tag that was used when creating the widget. You can edit the parameters of the widget to expand the types it can be linked to.

- If you draw an instance of the tag from the Tag Browser, you will find that the new widget is now a native option for that type.

Example: Create TankLevel

TankLevel
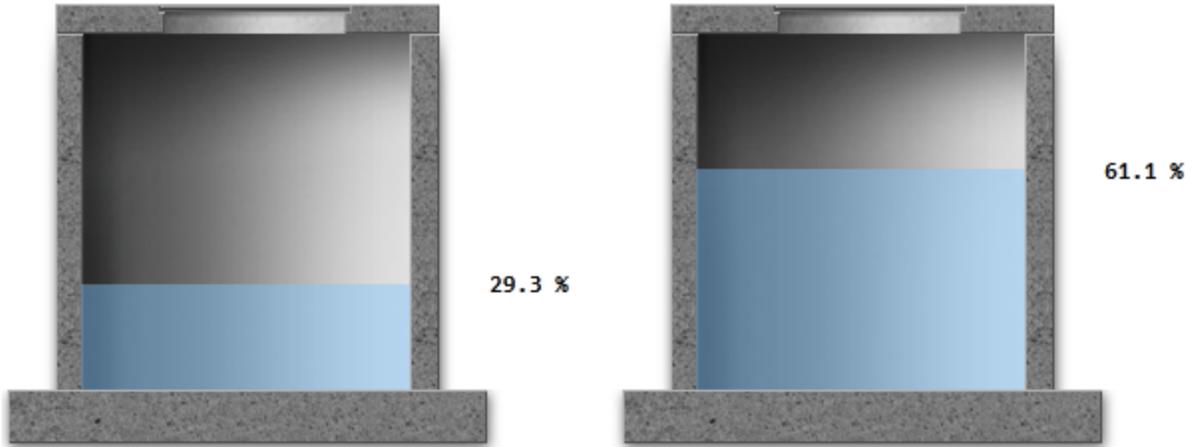
29.3 %

61.1 %

This tag widget, TankLevel, is designed to represent analog values as a tank filling with water. The numeric display rises with the water level. TankLevel will become a native widget for use by any Analog tag:
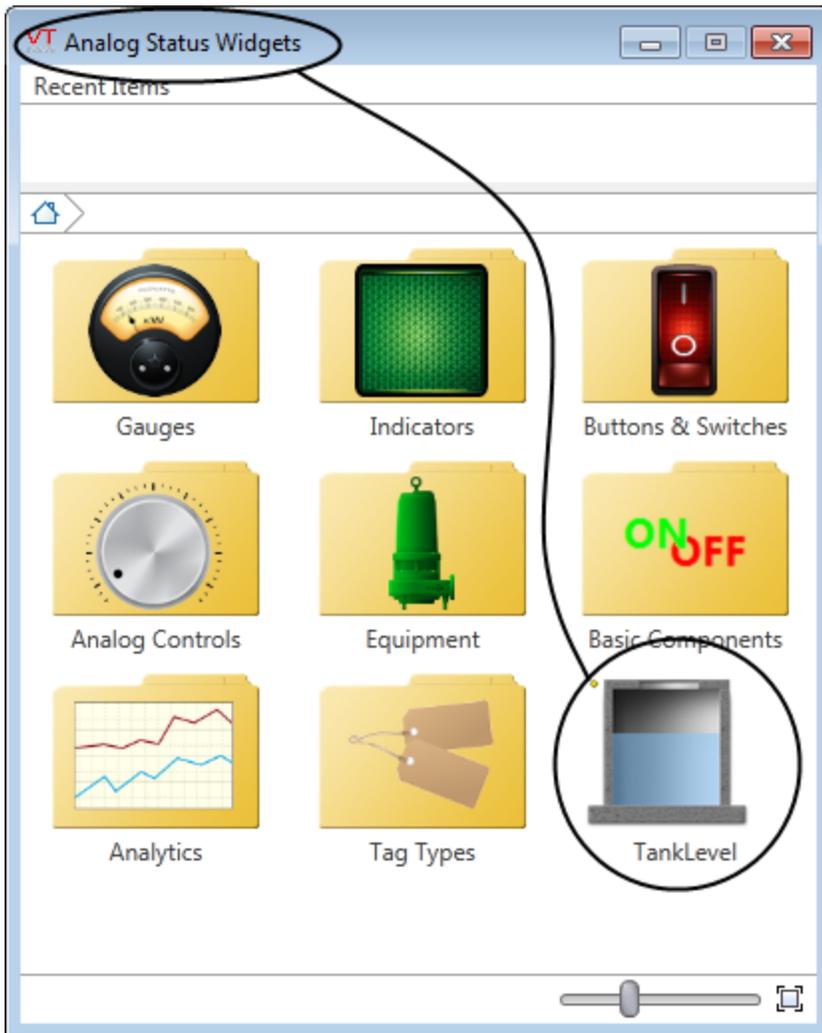
1. Create at least one Analog Input or Analog Status tag.

2. Open the Idea Studio.

3. Select the Images palette and open the Water & WW folder.

4. Drag WetWell 5 onto the page.

5. Select the Shapes palette.

6. Drag a rectangle onto the page.

7. Set the outline to "No Outline".

8. Position the rectangle so that it fills the interior of the well.

9. Adjust the properties of the rectangle to set the opacity to 50% (0.50).

10. Open the Scaling property of the rectangle, and select the Top Scaling option.

11. Select an Analog Input or Analog Status status tag. Ensure that the Use Tag Scaling option is checked.
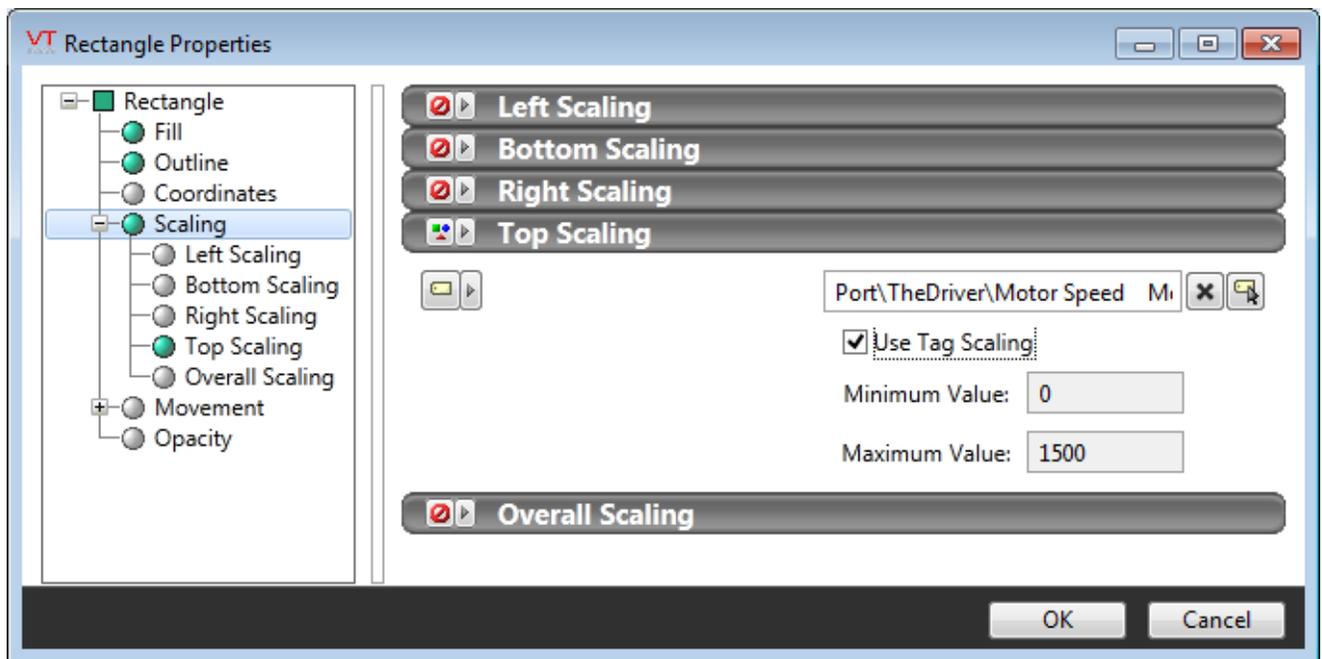
12. Select the Widgets palette.

13. Open the folder: Tag Types >> Analogs >> Analog Status.

14. Drag the widget Numeric Value onto the page.

15. Position the Numeric Value to the side of the well, even with the bottom edge of the rectangle.

16. Open the Movement properties of the Numeric Value and select the Vertical Movement option.

17. Select the same Analog Input or Analog Status tag that you used for the rectangle scaling.

18. Set the maximum pixels to 180.
    This is the height of the well portion of the WetWell 5 image. You can use the coordinate display the Idea Studio to verify this measurement.

19. Ensure that the movement direction is Up and that the Use Tag Scaling option is checked.

20. Select all three objects that you have drawn.

21. Click, Group in the toolbar, or right-click and select Group from the menu.

22. Select the option, New Widget.

23. In the New Widget dialog, provide a name, select Tag Widget and ensure that your Analog Input or Analog Status tag is selected.
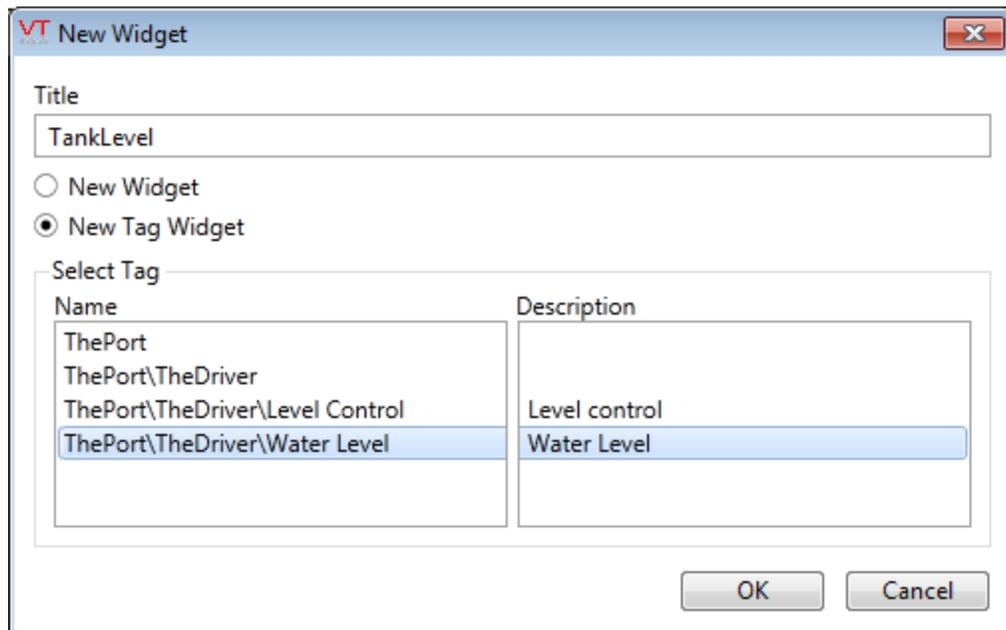
24. Right-click on the new widget and click Edit in the menu that opens.

25. Open the Widget Properties tab.

26. Click Tag Types in the ribbon.
    The Edit Parameters dialog opens. Whichever tag type you used above is shown in the list of associated tag types.

27. Add other Analog types to the list of associated tag types.

28. Click OK in the Edit Properties dialog to save your work. Close the widget editing page.

TankLevel will now be listed in the Widgets palette. You can drag it onto any page and link it to any of the types you selected in the Edit Parameters dialog.

Troubleshooting:

- On attempting to group the objects, an error message complains that some are not linked to a tag.
  All tag widgets within the selection set must linked to tags before they can be grouped. This does not interfere with your ability to use the widget for other tags later.

Related Information:

...Create a Widget For a Tag Structure – Multi-widget groups for user-created tag structures.

Next Steps:

...Select Types for a Tag Widget – Expand the list of tags that can be linked to your new widget.

Create a Widget For a Tag Structure

The various tags used for one item of equipment are often drawn together. If those tags are associated together as child tags of a new type you have created, using a context tag as a parent, then you can group the various related widgets and link the collection to the parent context. In the following (simplified) example, there is a valve with four I/O tags. The parent context tag has been turned into a new type, named ValveType and a second instance has been created.

Tags other than Valve1 and Valve2
have been hidden for this example.

The four child tags are normally shown and used as a group:

FlowRate drawn
as Meter 14

IsOpen drawn using
a valve from the
Widgets\Equipment
folder.

PositionSet
drawn as a knob

OpenClose drawn as
a selector switch

* The background is a tiled image.
* Labels are text, using the Label font with 14pt
* Radial scale drawn around the control knob using the
  Meter Parts folder of the Widgets palette.

Having drawn those four tags individually and applied labels, etc., it would be useful to group the display as a single widget that can be reused and linked new instances of ValveType.

To create a multi-tag widget:
Prerequisites: The situation just described, with a parent-child tag hierarchy that has been drawn once.

1. In the Idea Studio, select all the widgets, labels, etc. used to represent the tag structure.
2. In the ribbon, click Group >> New Widget.
   You could right-click on any of the selected objects and use these commands from the context menu.
3. In the New Widget dialog, ensure that the option, New Tag Widget is selected.
4. Provide a name for the new widget.
5. In the Select Tag area, ensure that the parent valve is selected.
   This is necessary so that the new widget can be drawn for other tags of the same type.
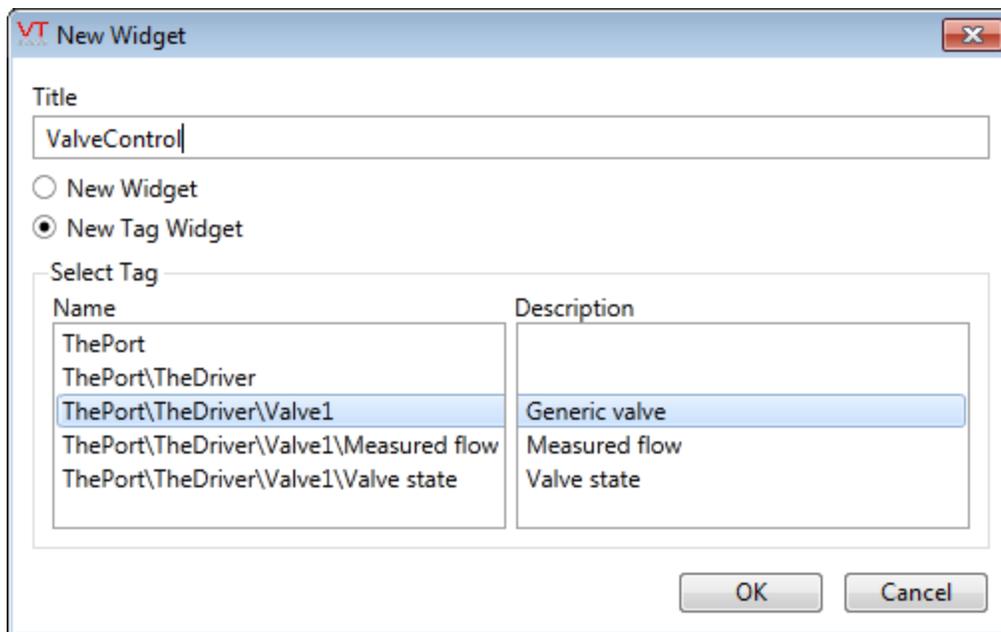
6. Click OK.

The separate objects will immediately be grouped into a new tag widget, available for drawing new ValveType tags.

The yellow marker that was added can be used to display the Questionable Data or Manual Data flag for the new widget. You are free to edit the widget to remove the Tag Icon Marker if you choose.

The new widget will have its own parameters for operator control. If you want to operate a control within the widget without the Trend window for the widget in general opening, you are advised to open the new widget's properties dialog and select Disable Trend. Most developers will also dis-

able navigation and tooltips for the widget itself, allowing operator controls for the component widgets to be used instead.
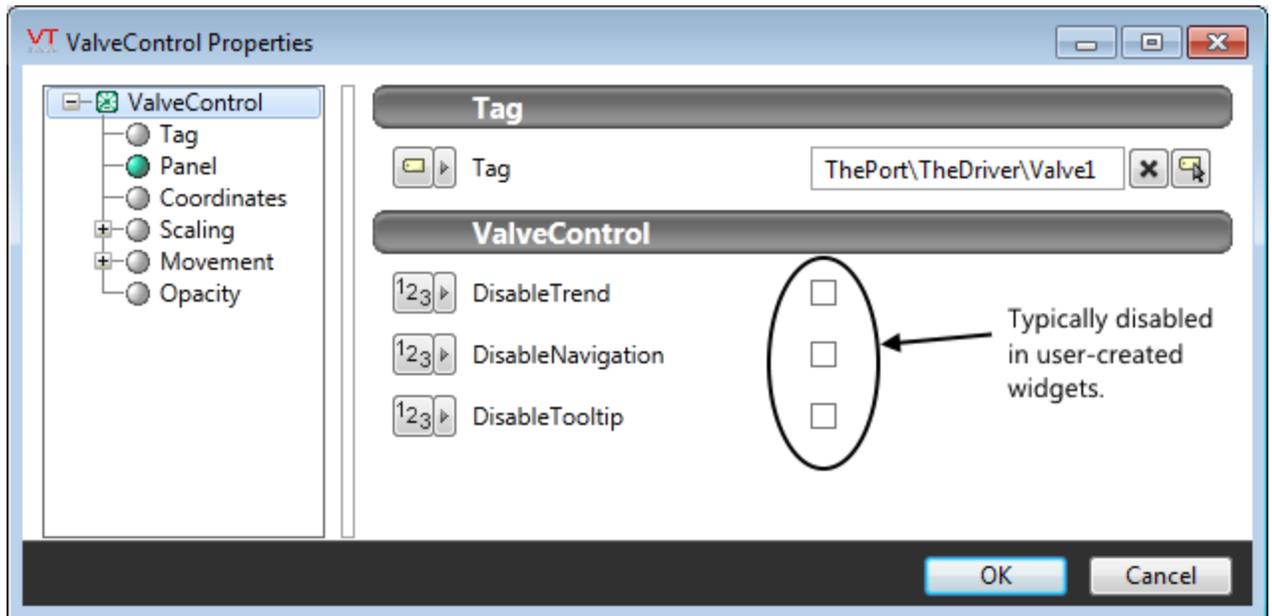
Example: Draw Valve2 using the new widget.

The new widget (ValveControl in the previous example) has been added to the Widgets palette.

1. Drag the widget from the palette to a page.

2. Link the widget to Valve2.
    The four widgets within ValveControl will link automatically to the four tags of Valve2.

3. [Optional, but recommended.] Open the widget's properties to disable trends, navigation and tooltip.

Troubleshooting:

- The wrong tag was chosen while creating the widget.
  You can change the list of tags associated with this new widget. Instructions are provided in a sub-topic.

- An extra object was included / an object was missed.
  You can edit the widget, adding and removing widgets, images and shapes.

- The label is the same for every widget drawn. This would be more useful if the label could say which particular valve was being drawn.

You can link labels (and anything else) to properties and parameters of the tag being drawn. Instructions are provided in a sub-topic.

Related Information:

...Tag Icon Marker

...Linked Tag Properties

Next Steps:

...Select Types for a Tag Widget

Tag Icon Marker

Used by: Widgets
Most VTScada widgets contain a tag icon marker. This is the yellow symbol that blinks a question mark when the linked tag's Questionable flag is set, and an exclamation mark when the linked tag is using manual data. If a widget has not yet been linked to a tag, the indicator will show a star (asterisk).

A Tag Icon Marker is added to all user-created tag widgets automatically. For it to function, the tag that the widget will link to must have at least one of the properties, "Questionable" or "ManualValue".
In the case of user-created tag widgets for tags that do not have either of those properties, you are advised to remove the Tag Icon Marker from the widget. This is most often the case when the widget is designed for use with a user-defined tag type, created from a Context type.
The properties dialog for the Tag Icon Marker widget:
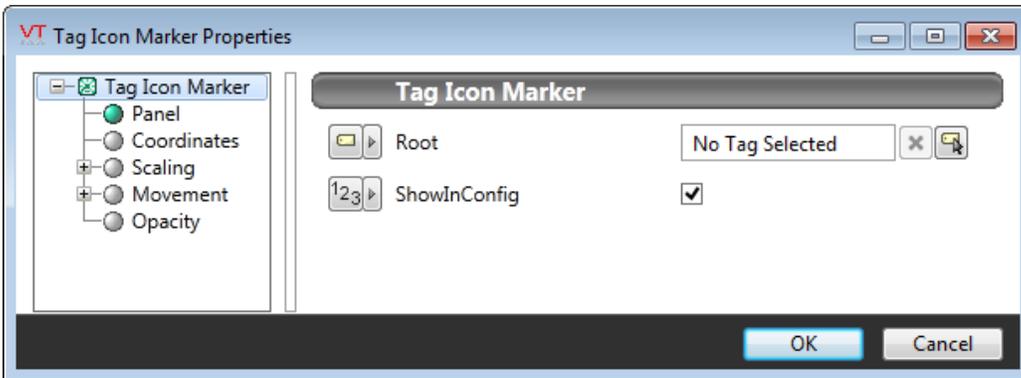
Root / Drawn Tag

If the root tag has the property, "Questionable", this will find that parameter and link to it.

ShowInConfig

If checked, the empty yellow marker will be visible on the widget, within the Idea Studio. This is not the case for VTScada widgets, all of which include a Tag Icon Marker, but may be useful for your custom widgets.

Select Types for a Tag Widget

A new tag widget can be linked only to tags of the type that the original for the widget was linked to. It is standard practice to expand this list to include similar types. For example, if the original was linked to an Analog Status tag, you may also wish to be able to link new instances of the widget to Analog Input, Calculation, and other tag with numeric values.

To modify the type list for a user-create tag widget:

1. Within the Idea Studio, open the widget for editing.
   Use the Open command in the file menu, press Control-O, or right-click on a drawn instance of a widget and select Edit from the pop-up context menu.

2. Select the Widget Properties ribbon.
3. Click the Tag Types button in the Widget Properties ribbon.
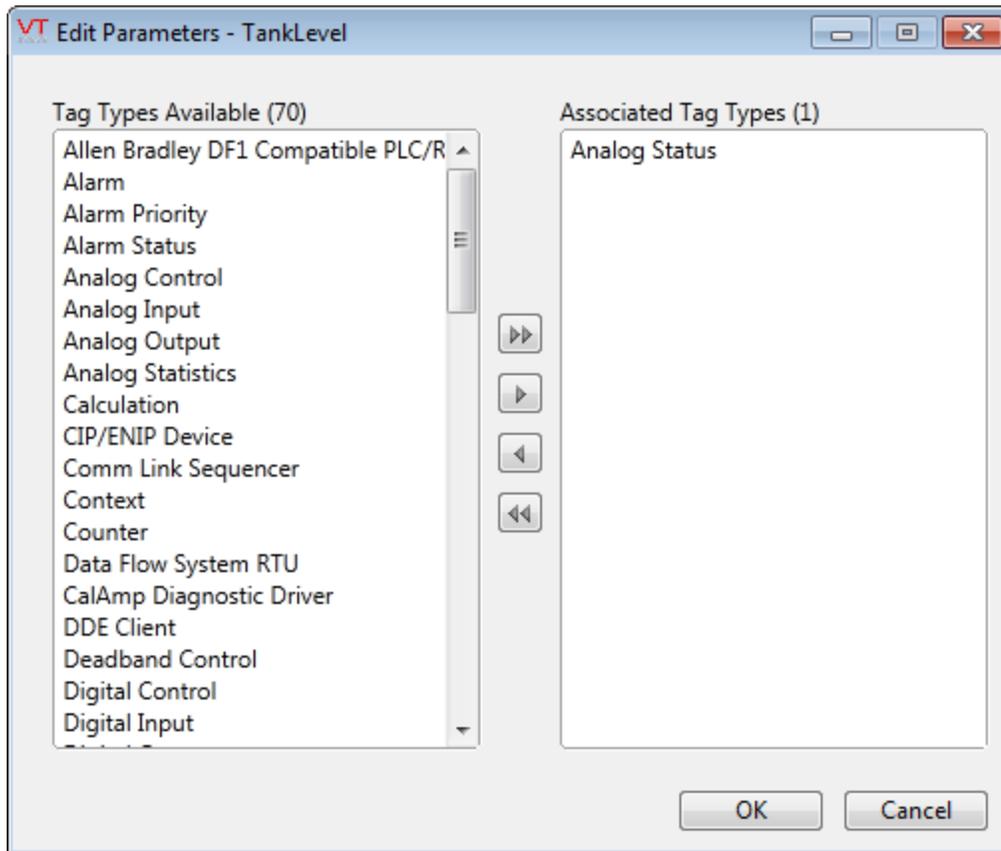
The Edit Parameters dialog opens.

4. Use the arrows to fill the column on the right with all the tags that you might want to link to your widget.

Troubleshooting:

- The Widget Properties ribbon is not visible.
  The Widget Properties ribbon is visible only when editing a widget. Use the Open command to open a widget for editing.

Linked Tag Properties

Widgets are linked to tags. Properties of those tags can therefore be accessed by, and represented in, the widget.
In the majority of cases, the only property that is used is the tag's value, but for the tag widgets that you create, there is no rule to say that only

the value may be used. You can link any tag property to any aspect of the widget.

One common example is to display the tag name or description as a label within the widget. You might also choose to display the area value, create a calculation that uses the tag's minimum and maximum scaled values, or add the engineering units to the display.

Example: Configure a text label to display the description field of the linked tag.

Prerequisites: This procedure applies only to tag widgets. The tag widget must be open for editing in the Idea Studio. Text has been added to the widget as a label.
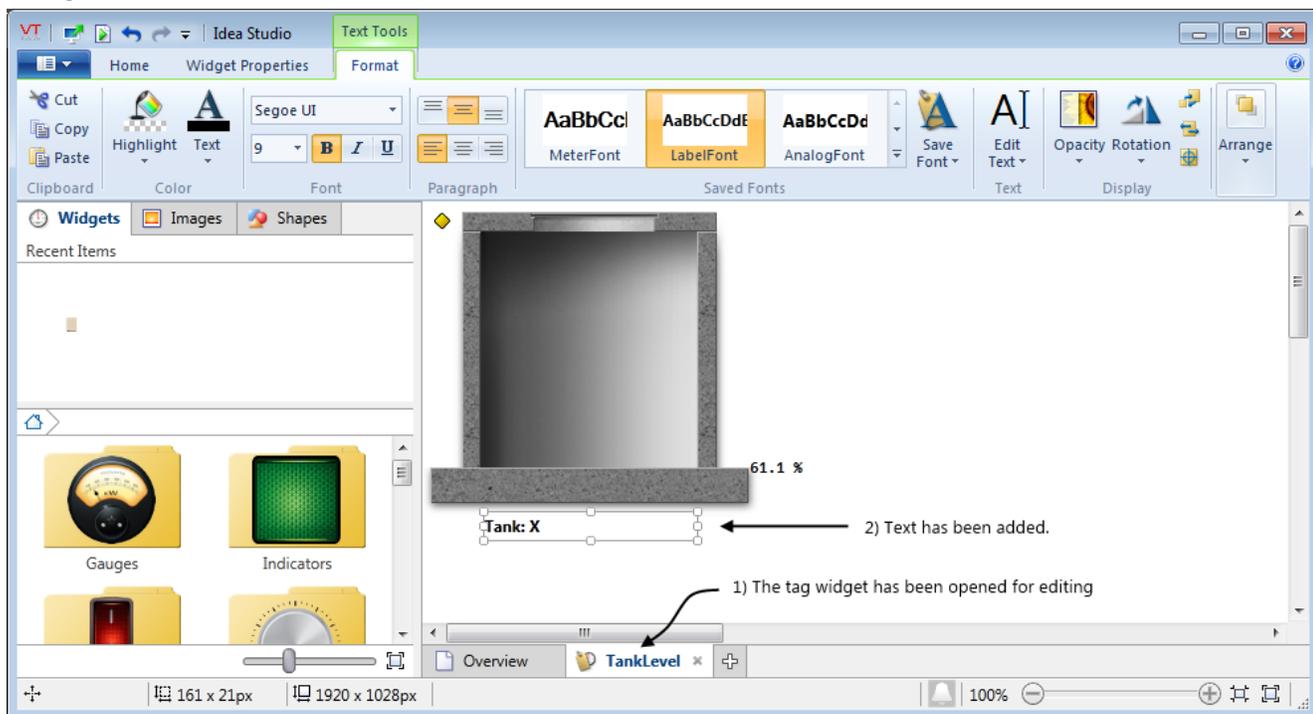
1. Open the properties dialog of the text label.
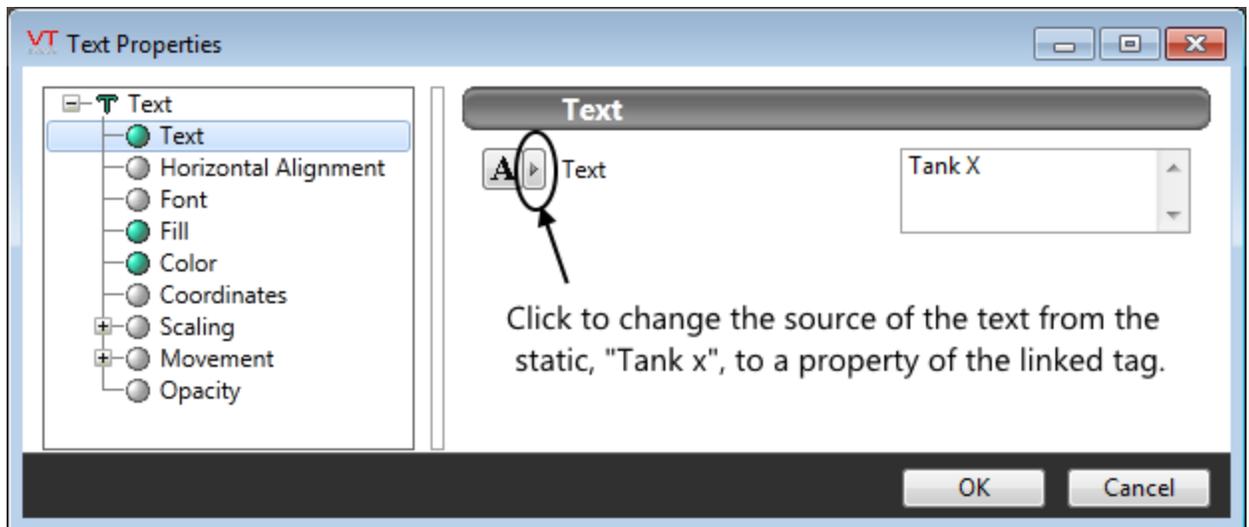
2. Click the button as indicated. This provides access to a selection of data sources for the text to be displayed.
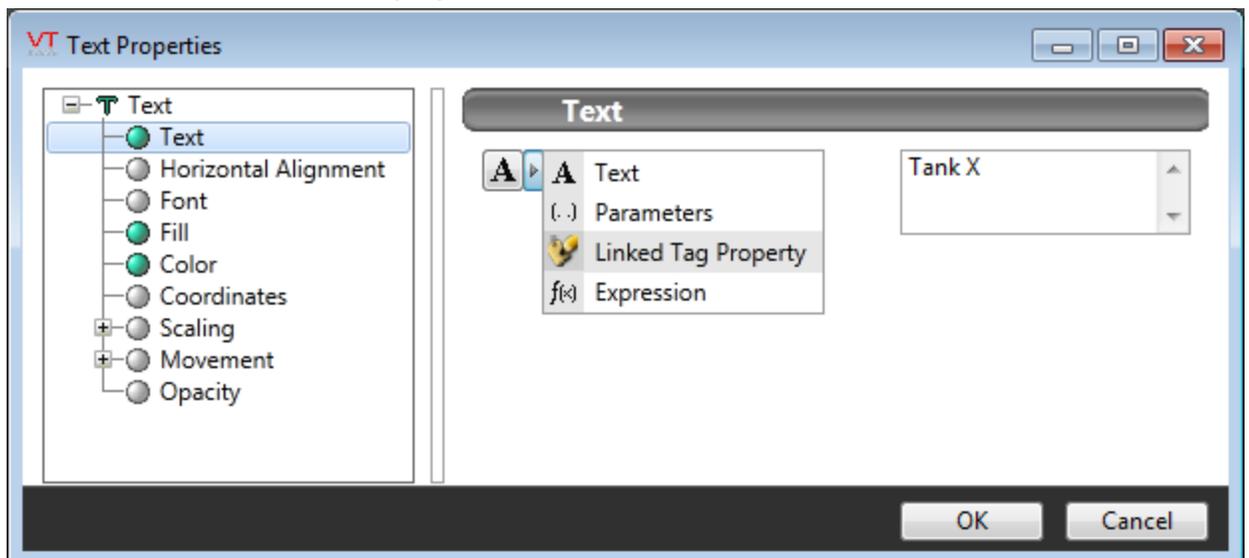
3. Choose the option, Linked Tag Property.
   The text input field will be replaced with a drop-down list of tag properties.
   The list of possible properties comes from the list of types that the widget can be linked to.

4. Expand the list to find and select Description.
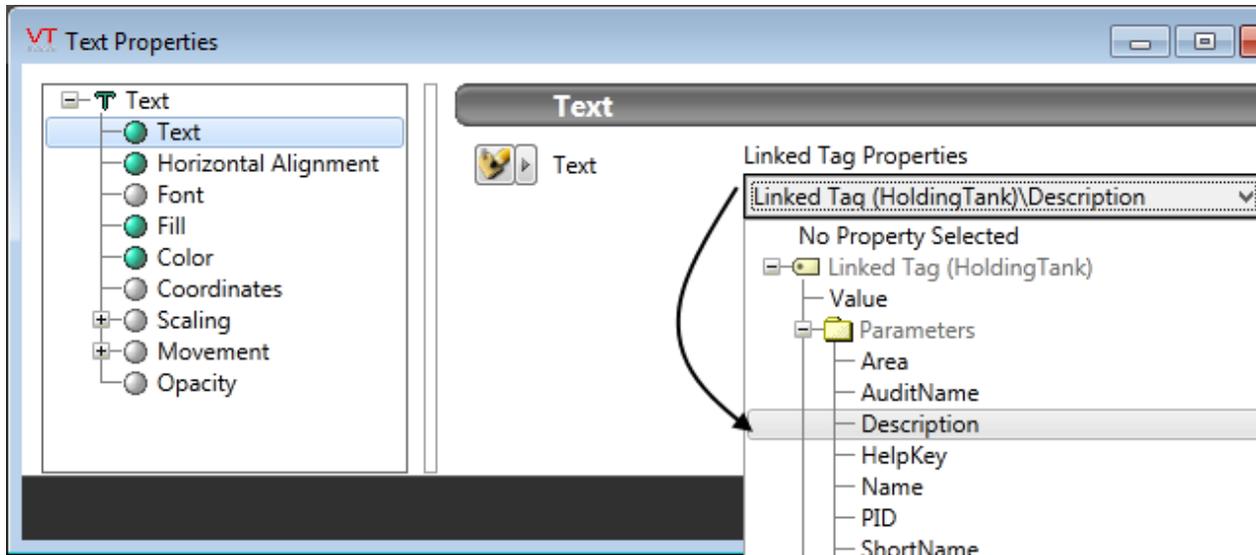
Each instance of the widget will now show the description field of the tag that it is linked to.

You might want the property to be only part of the label, or to create an expression that uses the property. After following the procedure described in the example, you can change the data source to Expression. You will find that the expression for the linked tag property is "Root\Property_Name". Use this in an expression as required.

Note: Take care: this information is for widgets in a drawn tag context, not for parameter expressions in general. Here, "Root" refers to the current Drawn Tag Context. Do not use "\Root\..." in a tag's parameter expression. \Root, as the base of the tag tree, will not be set until after the parameter expressions have been evaluated.

For example, to display the current tag's area value as part of the label "Tank in area: X", the expression is:

```
Concat("Tank in area: ", Root\Area)
```

Note: Widgets that are linked to a "Linked Tag Property", (usually the value property of a child tag) cannot be copied and pasted to other tag widgets or pages. You must change the widget's link to point to a specific tag (or leave it unlinked) before copying. To avoid damage to your custom widget, you can create a duplicate *within the current widget* of

the items you intend to copy elsewhere, change the links of those, then
delete the duplicate after copying it to another page or widget.

Related Information:

...Selecting a Data Source

...Expressions in Tags and in Widget Properties – VTScada Programmer's
Guide – Reference for the VTScada scripting language.

Custom Meters

If your site uses meters that do not look like any that are included with
VTScada, you can build your own using the components provided in the
Tools >> Meter parts folder of the Widgets palette.
A custom meter is a Tag Widget. The following topics describe the com-
ponents that are available. For the instructions to assemble these com-
ponents into a new meter please refer to Create a Tag Widget For One
Tag.

| draw scale | radial legend | radial indicator | bezel | glass overlay |
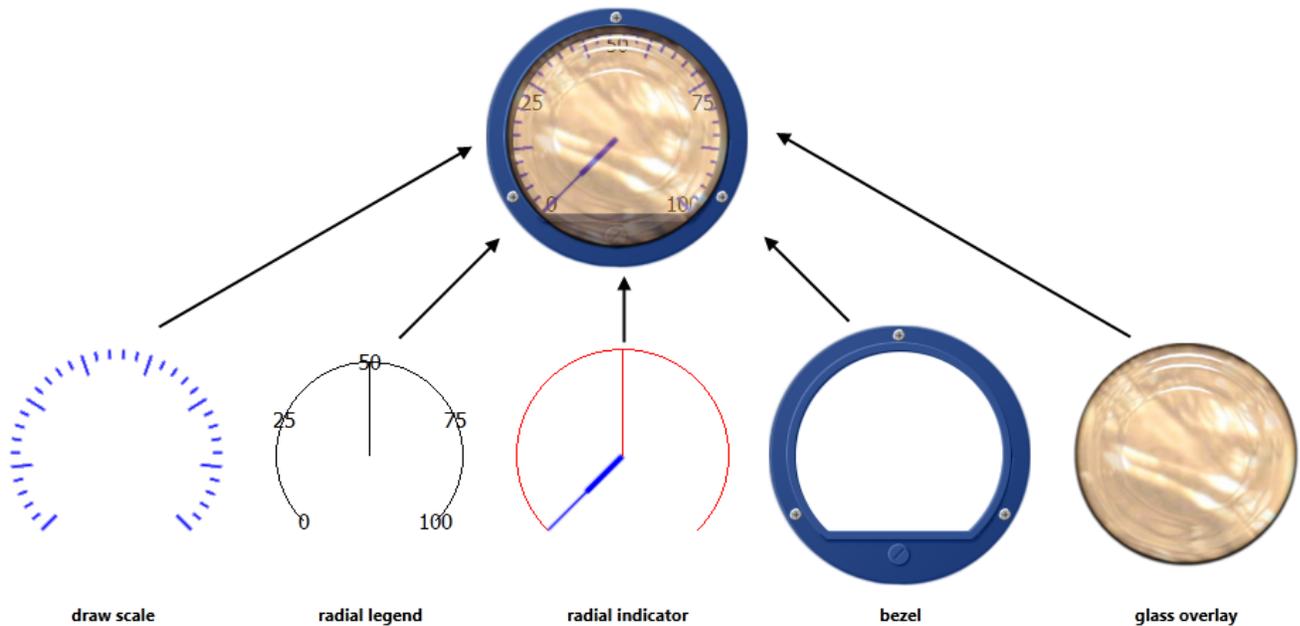
Note that the bezel and the glass overlay are images, found in the Meter
Parts folder of the Images palette.

Related Information:

...Linear Scale Widget – Tick marks used within a linear meter.

...Radial Scale Widget – Tick marks used within a radial meter.

...Radial Legend Widget – The numbers around a radial meter.

...Radial Indicator – The needle within a radial meter.

...Linear Legend Widget– The numbers along a linear meter.

...Linear Indicator – The indicator needle within a linear meter.

...Compass Indicator – The needle used within a compass.

Related Tasks:

...Create a Tag Widget For One Tag

Compass Indicator

Used by: custom compass designs of your own creation.

The compass indicator widget represents the associated tag's value as a needle that swings from the defined reference angle through a full circle in response the linked tag's value.
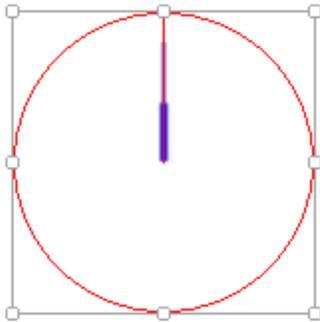
The properties dialog for the Compass Indicator:
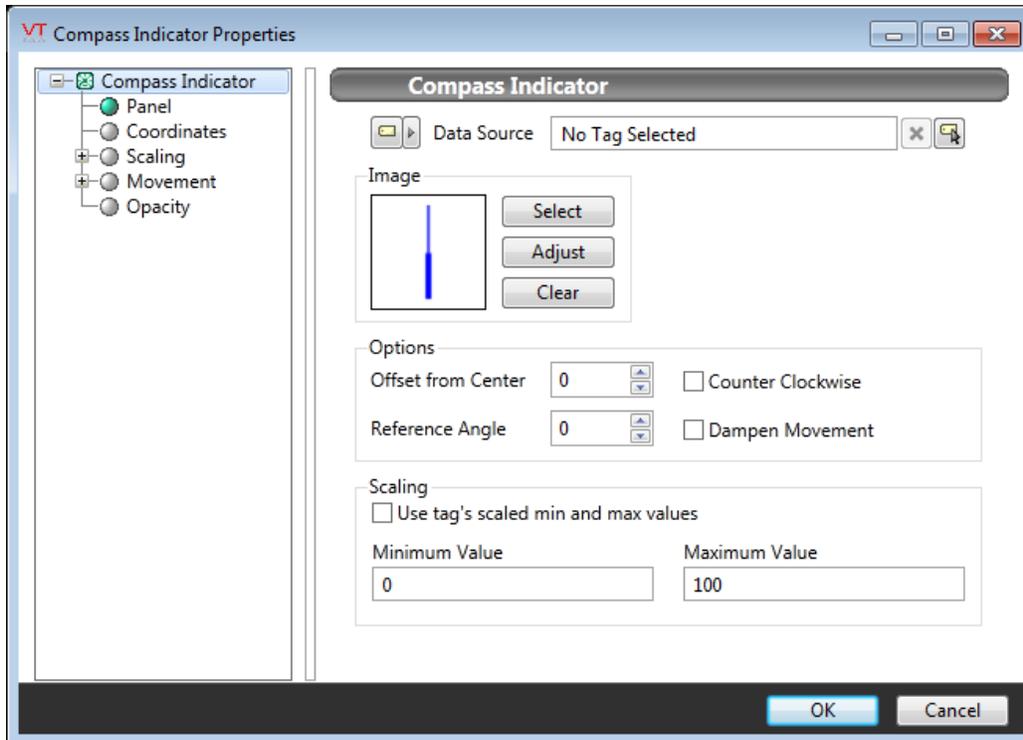
Data source

The indicator must be linked to a tag before it can be grouped into a new meter as part of a tag widget. Also, a tag must be selected before the "Use tag's scaled min and max values" option can be chosen in the Scaling part of this dialog.

Select / Adjust / Clear Image

The image to be used for the needle. In general, the image should have a vertical orientation. The Adjust option opens the Adjust Image dialog.

Options

Offset from center: A vertical offset, allowing you to position the needle so that a portion is below the center.
Counter Clockwise: Controls the direction that the needle rotates in response to increasing tag values.

Reference angle: The angle that corresponds to both the tag's minimum and maximum scaled value. Angles are measured in degrees, clockwise from vertical.

Dampen movement: When checked, applies an animation effect to needle motion so that the image appears to sweep from one value to another rather than jumping immediately.

Color options

A complete selection of options for controlling the color and opacity of the selected image.

Scaling

Used to set the tag values corresponding to the minimum and maximum angles. It is recommended that you use the tag's scaled values.

Related Information:

...Adjust Image Dialog – Tools within the Adjust Image dialog.

...Linear Scale Widget – Tick marks used within a linear meter.

...Radial Scale Widget – Tick marks used within a radial meter.

...Radial Legend Widget – The numbers around a radial meter.

...Radial Indicator – The needle within a radial meter.

...Linear Legend Widget– The numbers along a linear meter.

...Linear Indicator – The indicator needle within a linear meter.

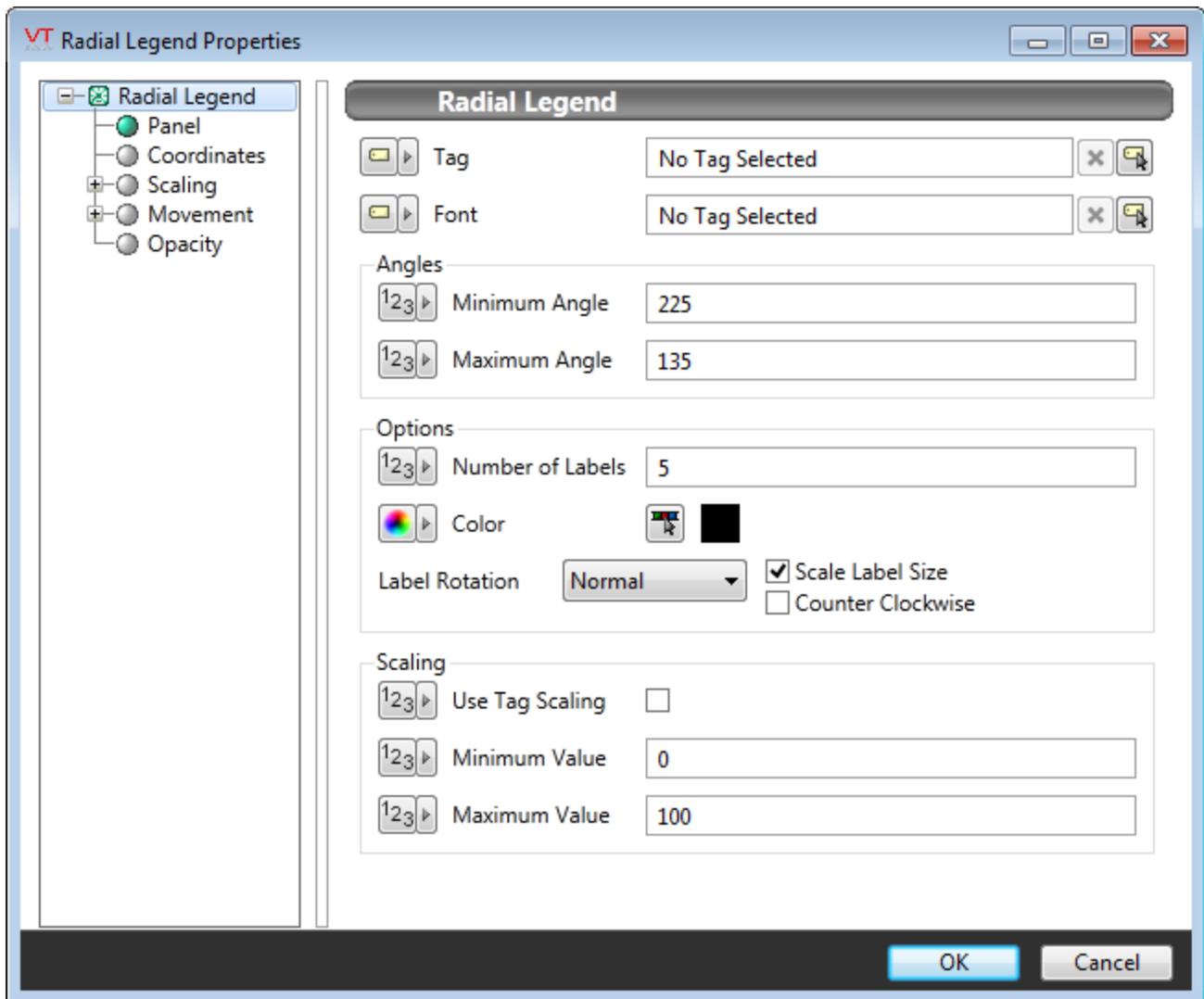...Compass Indicator – The needle used within a compass.

Linear Indicator

Used by: custom meters of your own design.

The linear indicator widget represents the associated tag's value as a line that moves across the scale range. Any image may be used in place of

the line, but it should have a clear pointer to help the operator read the current position of the meter.

The properties dialog for the Linear Indicator:
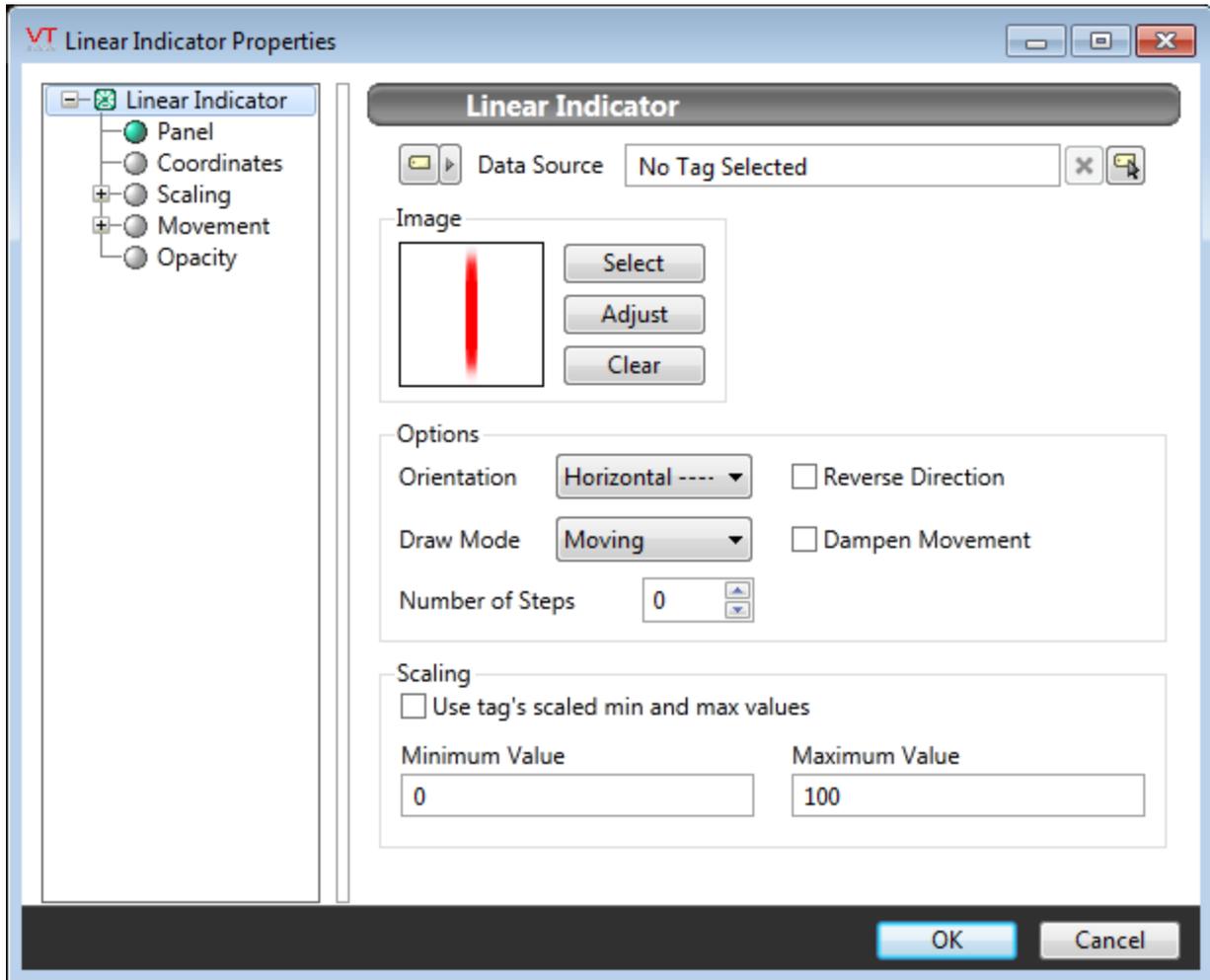
Data source

The indicator must be linked to a tag before it can be grouped into a new meter as part of a tag widget.

Select (Image)

Opens the Select Image Dialog, from which you can choose the

image to use for the indicator.

Adjust (Image)

Opens the Adjust Image dialog, with its full range of controls over image appearance.

Clear (Image)

Remove the image.

Options

Orientation: The direction that the needle image will travel. If vertical is to be selected, it is essential that you choose a horizontal needle.

Reverse direction: Control the direction in which increasing values are represented.

Draw mode:

- Moving – the value is represented as a moving line:

33.6 %

- Scale – represented by an expanding bar that fades slightly towards the maximum range.

33.6 %
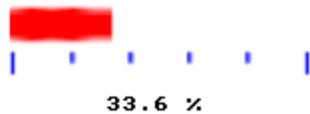
- Crop – represented by an expanding bar, that stays sharp to its maximum range, but blurs along the sides.
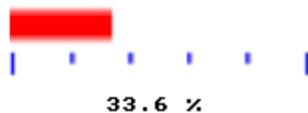
33.6 %

Dampen movement: When checked, applies an animation effect to needle motion so that the image appears to sweep from one value to another rather than jumping immediately.
Number of steps: If set to 0, the indicator shows the exact value. If set to a larger value, the indicator is divided into the given number of steps and the display will round to the nearest step. So, for example, a step of 1 would mean all on or all off. 20 steps on an indicator that ranges from 0 to 100 would round the display to the nearest multiple of 5.

Scaling

Used to set the tag values corresponding to the minimum and maximum angles. It is recommended that you use the tag's scaled values.

Related Information:

...Adjust Image Dialog – Tools within the Adjust Image dialog.

...Linear Scale Widget – Tick marks used within a linear meter.

...Radial Scale Widget – Tick marks used within a radial meter.

...Radial Legend Widget – The numbers around a radial meter.

...Radial Indicator – The needle within a radial meter.

...Linear Legend Widget– The numbers along a linear meter.

...Linear Indicator – The indicator needle within a linear meter.

...Compass Indicator – The needle used within a compass.

### Linear Legend Widget

Used by: custom meters of your own design. Often used in combination with the Linear Scale widget.
The Linear Legend Widget represents the associated tag's minimum and maximum scale range as a series of numbers in a line. The widget can be oriented either vertically or horizontally. Text can be configured to a set size, or to scale with the widget. In the following example, the horizontal

legend has a smaller width than the vertical legend, therefore a smaller font is used.

0

25

50

0   25   50   75   100

75

100
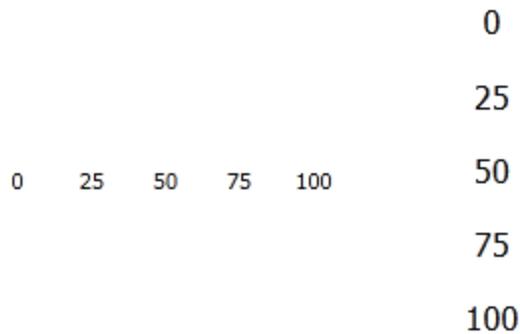
The properties dialog for the Linear Legend widget.
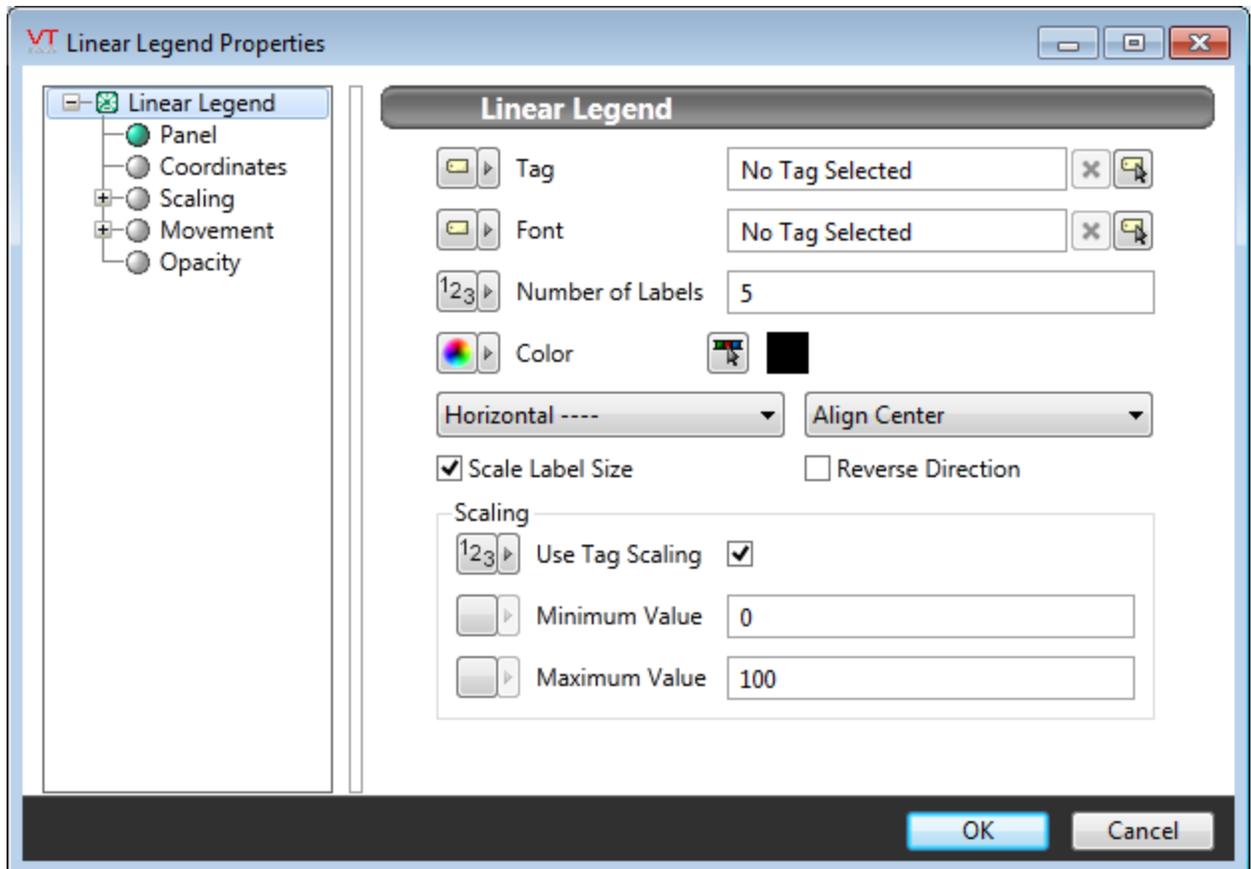
Tag

The legend must be linked to a tag before it can be grouped into a new meter as part of a tag widget. Also, a tag must be

selected before the "Use tag's scaled min and max values" option can be chosen in the Scaling part of this dialog. Note that there is no Link option for this widget – linking must be done from within this properties dialog.

Font

The font tag to be used for the numeric values of the scale.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Number of Labels

Sets the number of labels shown as part of the legend.

Color

Controls the color to be used for the text of the legend.

Orientation

Choose whether the legend should run horizontally or vertically.

Alignment

Sets the placement of the text within the bounding box. For horizontal legends, the options are Top, Center and Bottom. For vertical legends, the options are Left, Center and Right.

Scale Label Size

If checked, the font size with scale with the bounding box of the label. Select this option when the legend is to be used on pages with the Scale Page option enabled. Otherwise, you may find it easier to maintain consistent label sizes by deselecting this option and using the Font selection to control the label size.

Reverse Direction

By default, values increase from left to right (horizontal legend) or bottom to top (vertical legend). Select this option to reverse the direction in which values will increase.

Scaling

Choose whether to provide your own values for the minimum and maximum scaling or to use values supplied by the linked tag. In most cases, it is best to use scaling values from the linked tag.

Related Information:

...Linear Scale Widget

Linear Scale Widget

Used by: custom meters of your own design. Often used in combination with the Linear Legend widget.
The Linear Scale Widget represents the associated tag's minimum and maximum scale range as a series of major and minor tick marks. The widget can be oriented either vertically or horizontally.
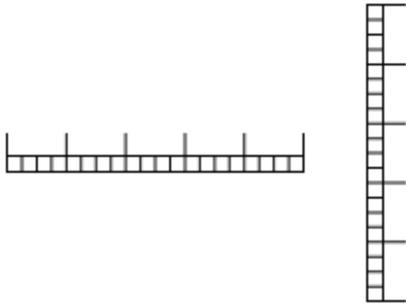
The properties dialog for the Linear Scale widget.
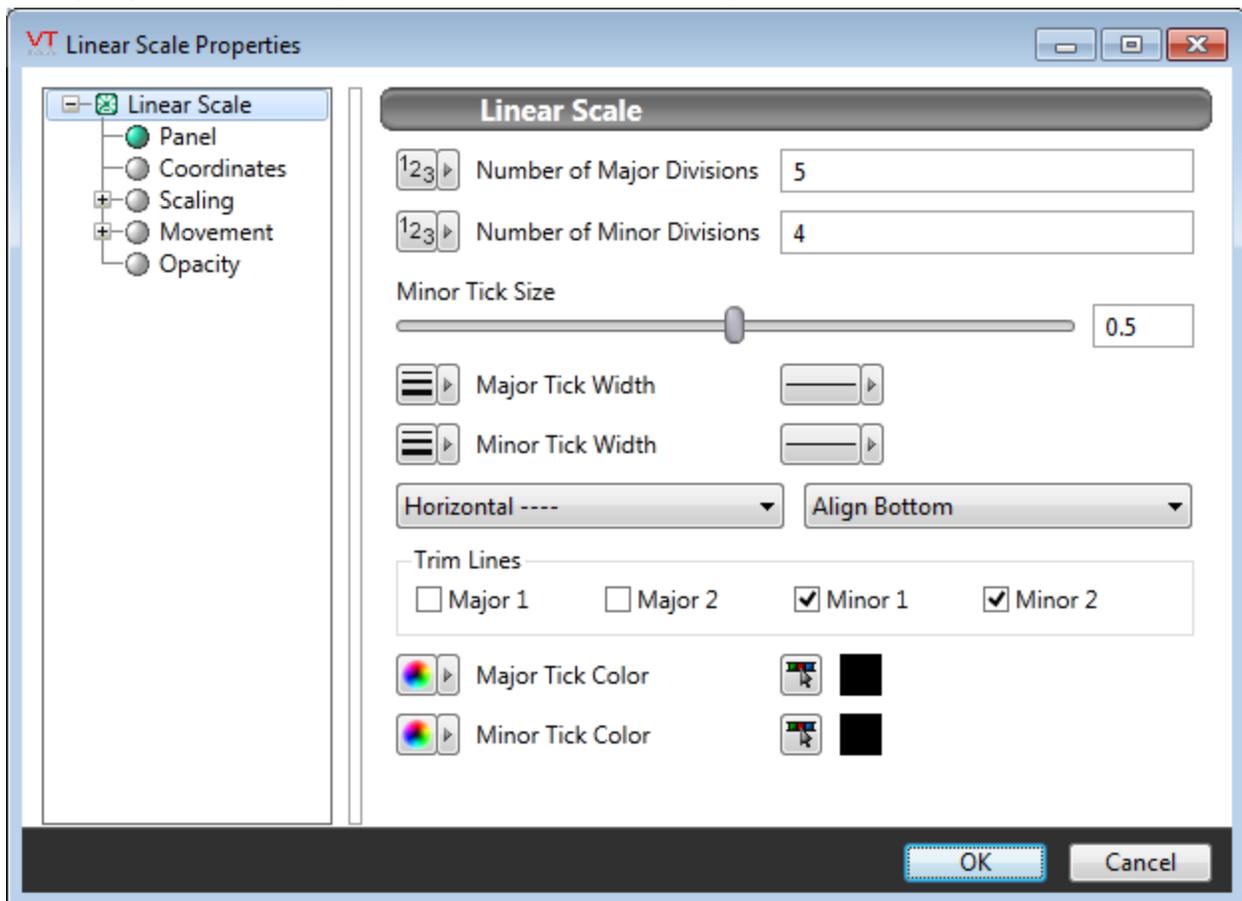
Number of Major and Minor Divisions

Controls the number of large (major) to be drawn across the scale and the number of small (minor) tick marks to be drawn between major tick marks.

Minor Tick Size

The minor tick size is set as a percentage of the major tick size.

The major tick size is controlled by the bounding box of the widget.

Major and Minor Tick Width

Sets the width of each type of tick mark, measured in pixels. Note that when the correct placement of a tick mark falls between two pixels on the screen, the widget will compensate by using both adjoining rows of pixels, varying the intensity of each row according to the percentage of the correct tick mark placement that would fall upon it.

Orientation

Controls whether the scale should be drawn left to right (horizontally) or up and down (vertically).

Alignment

Sets the placement of the *minor* tick marks within the bounding box. The major tick marks will always fill the width of the box. For horizontal scales, the options are Top, Center and Bottom. For vertical scales, the options are Left, Center and Right.

Trim Lines

Optional lines that connect the tick marks as shown. In this example trim line 2 of both the major and minor tick marks overlap, since the minor tick marks are aligned with the bottom of the bounding box.
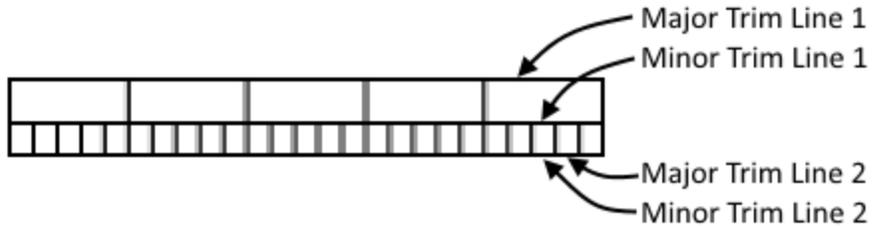
Major and Minor Tick Color

Controls the color used for the tick marks.

Related Information:

...Linear Legend Widget

...Radial Scale Widget – Tick marks used within a radial meter.

Radial Indicator

Used by: custom meters of your own design.
The radial indicator widget represents the associated tag's value as needle rotating about a fixed center. The red lines shown here are not visible to the operator. They show only in the Idea Studio, as a guide to help you place the indicator. The needle may be any image, any color.
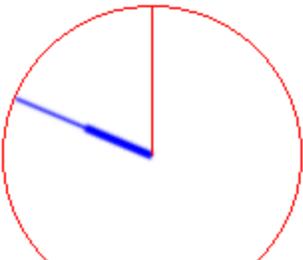
The properties dialog for the Radial Indicator:
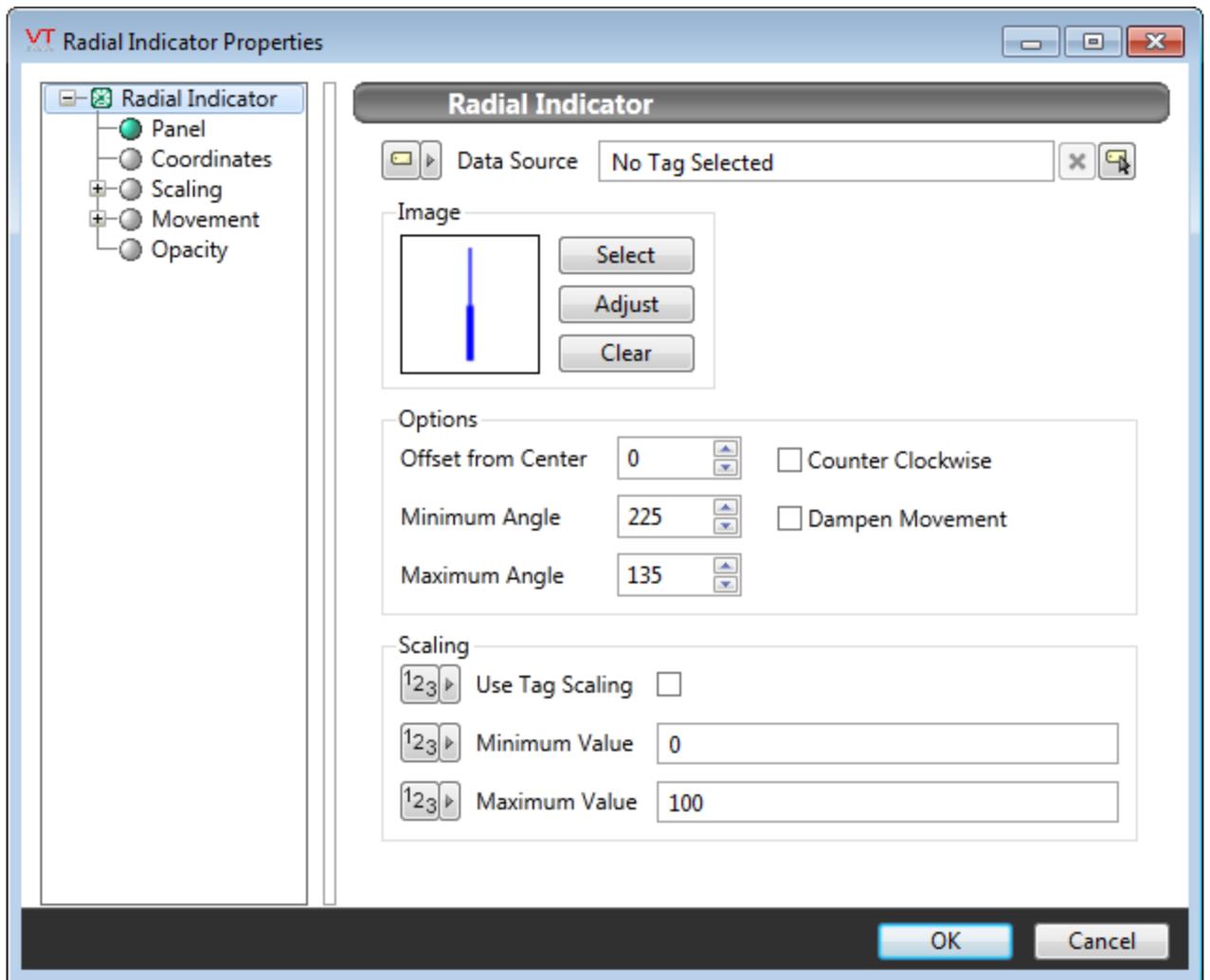
Data source

The indicator must be linked to a tag before it can be grouped into a new meter as part of a tag widget. Also, a tag must be selected before the "Use tag's scaled min and max values" option can be chosen in the Scaling part of this dialog.

Select (Image)

Opens the Select Image Dialog, from which you can choose the image to use for the indicator.

Adjust (Image)

Opens the Adjust Image dialog, with its full range of controls over image appearance.

Clear (Image)

Remove the image.

Options

Offset from center: A vertical offset, allowing you to position the needle so that a portion is below the center.
Counter Clockwise: Controls the direction that the needle rotates in response to increasing tag values.
Minimum angle & maximum angle: Controls the range of the needle motion, assuming a clockwise rotation. Values are measured in degrees, clockwise, relative to the vertical position.
Dampen movement: When checked, applies an animation effect to needle motion so that the image appears to sweep from one value to another rather than jumping immediately.

Scaling

Used to set the tag values corresponding to the minimum and maximum angles. It is recommended that you use the tag's scaled values.

Related Information:

...Adjust Image Dialog – Tools within the Adjust Image dialog.

...Linear Scale Widget – Tick marks used within a linear meter.

...Radial Scale Widget – Tick marks used within a radial meter.

...Radial Legend Widget – The numbers around a radial meter.

...Radial Indicator – The needle within a radial meter.

...Linear Legend Widget– The numbers along a linear meter.

...Linear Indicator – The indicator needle within a linear meter.

...Compass Indicator – The needle used within a compass.

Radial Legend Widget

Used by: custom meters of your own design.
The Radial Legend widget represents the associated tag's minimum and maximum scale range as a series of numbers around an arc.
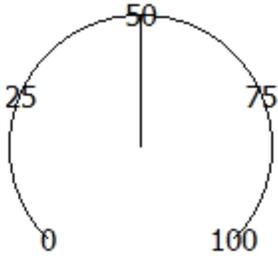
The properties dialog for the Radial Legend widget:

Tag

The legend must be linked to a tag before it can be grouped into a new meter as part of a tag widget. Also, a tag must be selected before the "Use tag's scaled min and max values" option can be chosen in the Scaling part of this dialog.

Font

The font tag to be used for the numeric values of the scale.

Angles

Minimum and maximum angles, measured clockwise around the meter. All angles are measured in degrees, starting at North (vertical) and measured clockwise.

Options

Set the number of labels to be displayed. Can be no less than one.If too many labels are shown, then they will overlap, making it difficult for the operator to read the meter.

Meters will look best if the number of labels is a function of the tick marks shown by the Draw Scale.

Scale Label Size. If checked, the text of the labels will scale with the rest of the meter when resized.

Color: Opens the Select Color dialog, from which you can choose the color to be used for the legend.

Counter Clockwise. If checked, numbers will increase from right to left, counter-clockwise across the meter

Label Rotation: Select one of the following options:

Label rotation options:
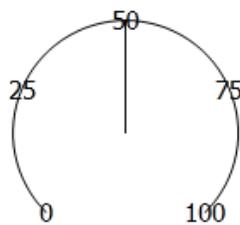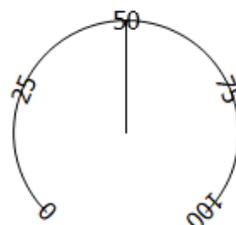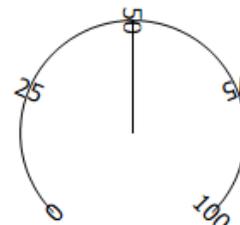
Normal   Contoured   Axis

Scaling

Used to set the smallest and largest values to be displayed on the meter. Unless you are certain that this meter will be used only for a given scale range, it is best to allow the scale to change to match the linked tag.

Related Information:

...Radial Scale Widget

...Radial Indicator – The needle within a radial meter.

...Linear Indicator – The indicator needle within a linear meter.

...Compass Indicator – The needle used within a compass.

Radial Scale Widget

Used by: custom meters of your own design. Often used in combination with the Radial Legend widget.
The Radial Scale Widget represents the associated tag's minimum and maximum scale range as a series of major and minor tick marks. The scale will always be round, regardless how the bounding box is stretched. In all cases, the smaller dimension of the bounding box will be used to set the diameter of the scale.
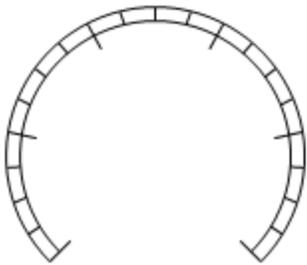
The properties dialog for the Linear Scale widget.
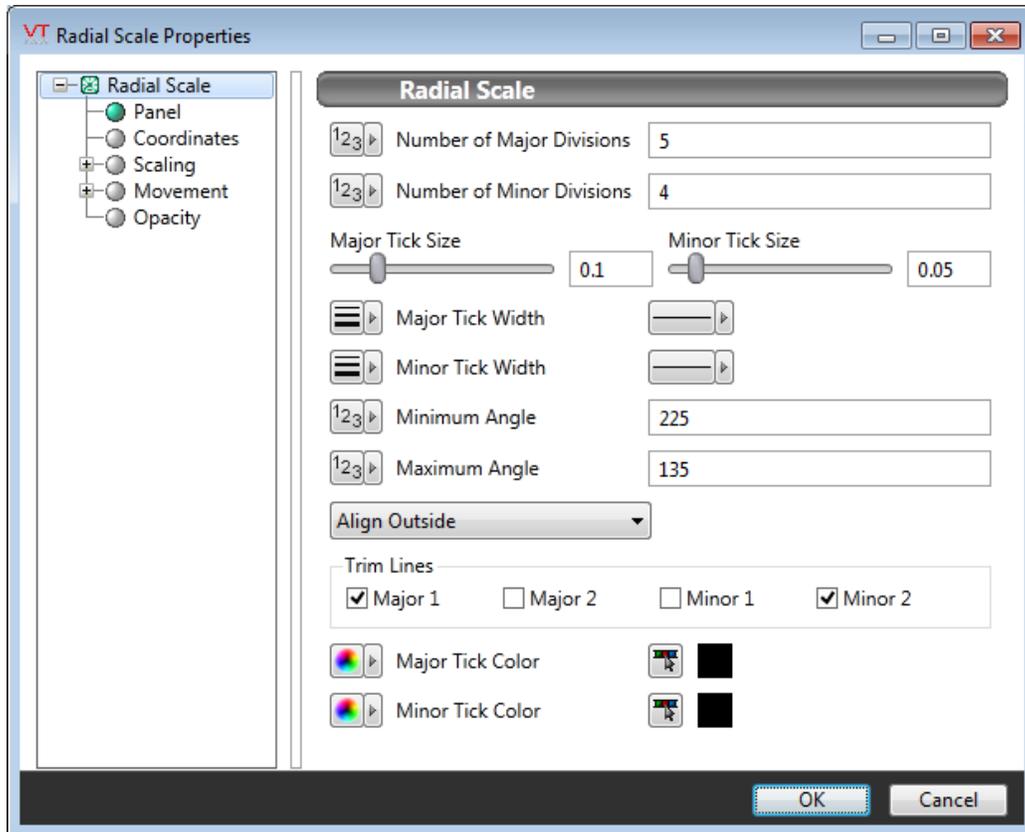
Number of Major and Minor Divisions

Controls the number of large (major) to be drawn across the scale and the number of small (minor) tick marks to be drawn between major tick marks.

Major and Minor Tick Size

Both tick sizes are expressed as a percentage of the lesser dimension of the widget's bounding box. They will scale as the display changes. By default, the major tick size is 10% of the scale's radius. The minor tick mark size is 5%.

Major and Minor Tick Width

Sets the width of each type of tick mark, measured in pixels.

Minimum and Maximum Angle

Controls the angle covered by the scale. Angles in VTScada are measured clockwise from vertical, using degrees. This is why the default minimum angle is a larger number than the default maximum. The scale sweeps clockwise from the 225 degree position, past vertical to the 135 degree position.

Alignment

Sets the placement of the *minor* tick marks relative to the major tick marks. The minor tick marks may be aligned against the inside edge, center, or outside edge of the major tick marks.

Trim Lines

Optional lines that connect the tick marks as shown. Trim line 1 (major and minor) is towards the outside of the arc and trime line 2 is towards the inside. In this example trim line 2 of both the major and minor tick marks overlap, since the minor tick marks are aligned with the outside of the arc.
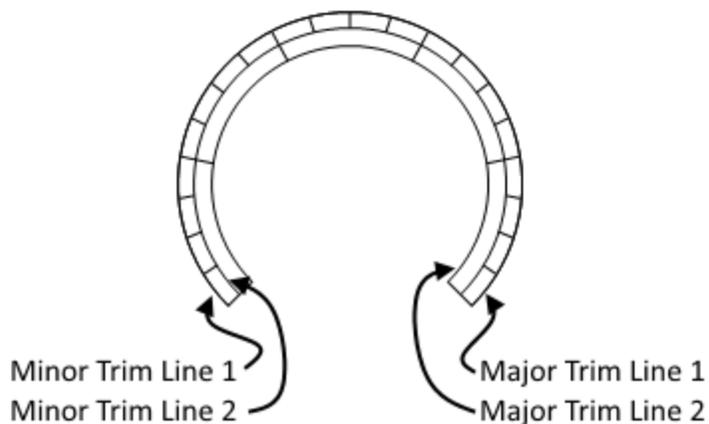
Major and Minor Tick Color

Controls the color used for the tick marks.

Related Information:

...Radial Legend Widget

Import and Export Widgets

There is no need to export your custom widgets. They are stored in .SRC files, within the \Widgets folder of your application and may be copied freely to other applications. System integrators who prefer not to distribute their custom source code can choose to distribute only the compiled .RUN files.
Resources used by the widget, such as images and custom Style Settings tags, will need to be imported separately.

To import a widget:
1. Open the Idea Studio.
2. Open the File menu.
3. Expand the Import menu.
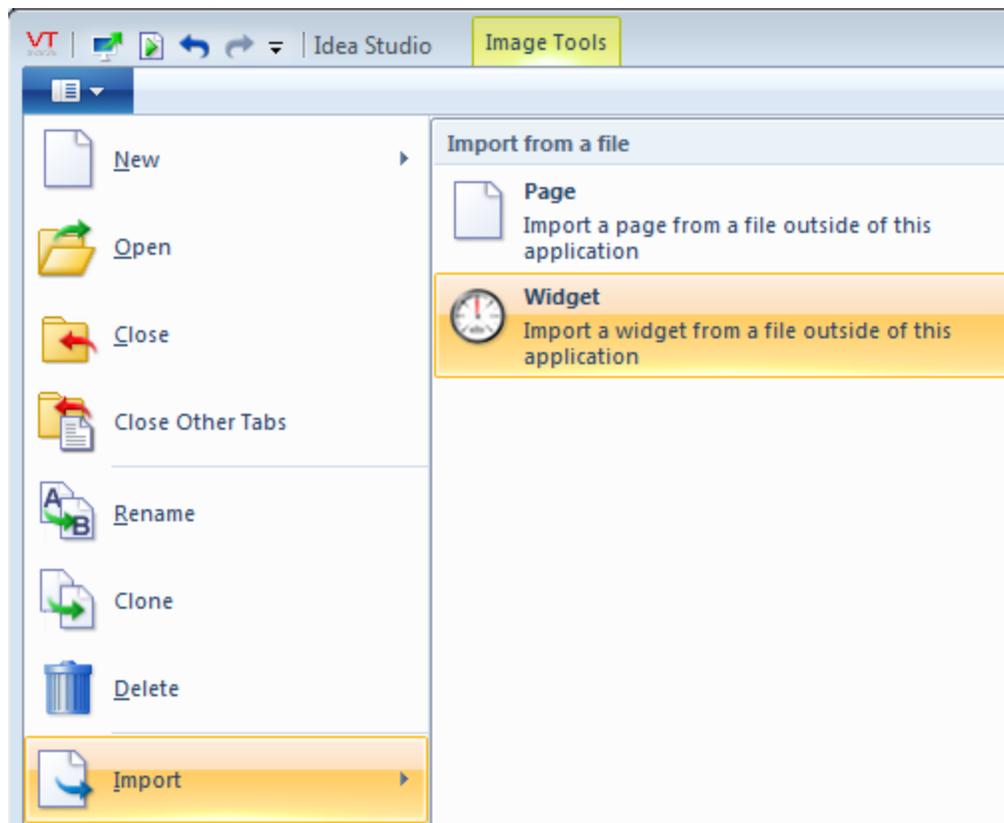
4. Select Widget.

5. Browse to select the widget's .SRC file.

If you import a tag widget, VTScada assumes that you have matching tags within your application that can be linked to the widget's parameters.

Delete a Widget

VTScada widgets cannot be deleted, but you can delete the ones that you created.

Both library widgets and tag widgets are added to the palette menu automatically. You may have edited the menu to control which folder the widget is stored within.

It is up to you to remove all instances of the widget from the palette before deleting it. You may choose to clean up the palette menu later, at which time the widget's entry will use a cryptic unique id rather than the former name.

To remove a group:

1. Select the group.

2. Right-click to open the context menu.

3. Click, Ungroup.

To delete a library widget or a tag widget:

1. Open the Idea Studio.

2. Open the widget for editing
   Right-click on an instance and select Edit from the menu, or use the File menu to open the widget.

3. In the Widget Properties toolbar, click Delete Widget.

Widget Reference

The following topics provide a complete reference to the widgets that come with VTScada. Note that you (or your systems developer) may have created customized widgets that are unique to your application.

Widgets shown in the table of contents with an asterisk can be used with Style Settings tags.

Related Information:

...Border

...Grid

...Folders

...Frame

Active Indicator Widget

Used by: Data Flow RTU Tag,   Polling Driver Tag.
* Does not use the Stye Settings tag.
The Active Indicator widget is used to create a rectangular indicator that turns green when the associated RTU is the actively connected site. The rectangle will otherwise be black.

Configuration options in the properties dialog for this widget are as follows:
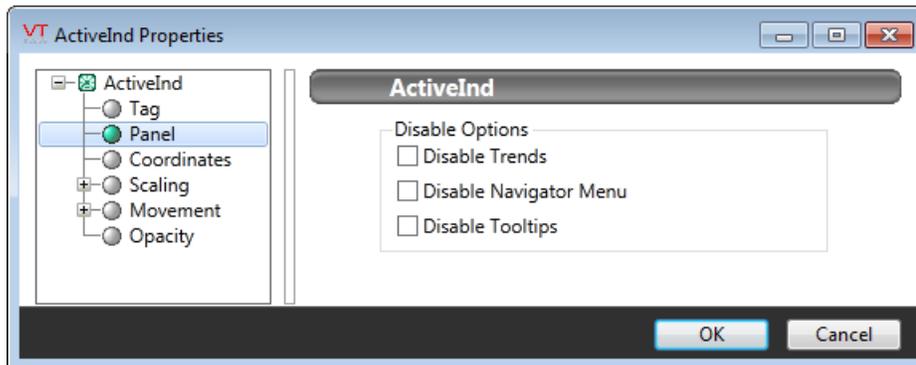
The only configurable parameters for the Active Indicator style dialog are the Disable Options check boxes for limiting operator interaction.

Widget Folders:
Analytics\Communications\Polling\
Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Add Note Button

Used by: Notebook.
* Does not use the Stye Settings tag.
Creates a button on the page that operators may click in order to add a note to the associated Notebook. Two versions of the button are available.

Configuration options for the button are as follows:
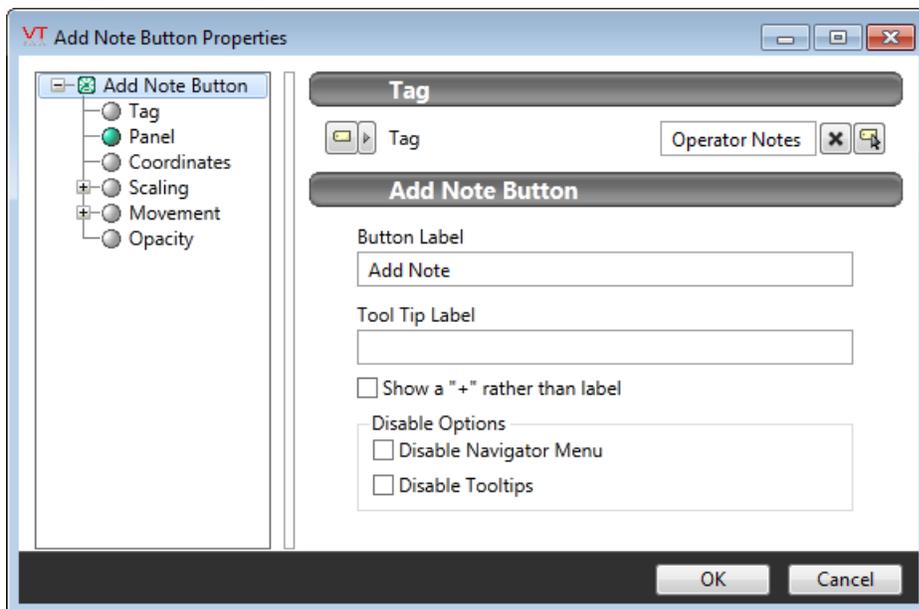
Button Label

Enter the text that will be displayed on the button. If the option, "Show a +" is selected, no label will be shown, regardless of the contents of this field.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tip Label

Provide text for the tool tip that will be displayed when the operator moves the cursor over the button. The associated Notebook tag name will also be shown – the name to the upper right and the tool tip to the lower right.

Show a "+" rather than label

When checked, the button shows only a plus symbol (as illustrated above).

Widget Folders:
Basic Components\Specialty\Notebook\
Tag Types\Data Logging & Reports\Notebook\

Add Note Field Widget

Creates a note entry field, similar to that found in the Operator Notes page. Operators can use this to add notes to a pre-selected Notebook without leaving the page to which this widget was added.
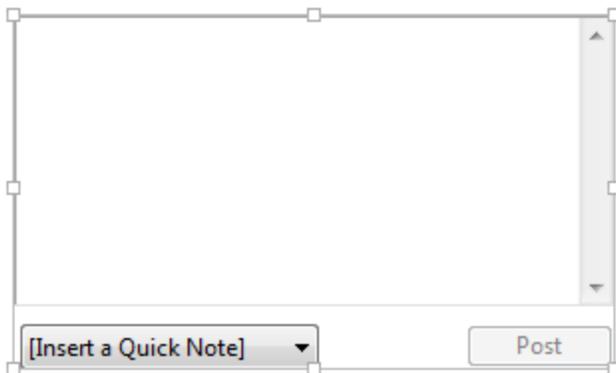
Configuration options for the field are as follows:
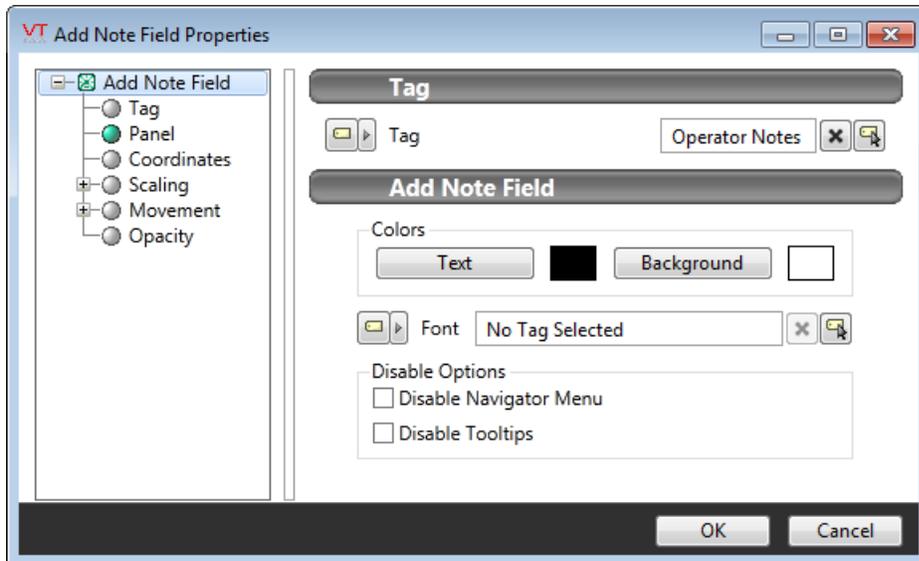
Color – Foreground and Background

Set the color to be used by text in the editing field and the background color of the field itself.

Font

Select the Font tag that will control the appearance of text within the editing field.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Specialty\Notebook\
Tag Types\Data Logging & Reports\Notebook\

Alarm Priority Box

Provides a visible indication when the linked tag is in an active or unacknowledged alarm state, and when it is either shelved or disabled.
The box will not be visible for alarms that are unacknowledged but no longer active unless the property

AlarmPriorityIndicatorShowNormalUnacked is set to 1. (Note that "trip" alarms are never in an active state.) The icon will always be displayed when the alarm is shelved or disabled.

You can link this widget to a parent tag rather than a specific status tag. When linked to a parent, it will become activated when any child tag in the hierarchy is in an active alarm state. The icon will display the highest priority of the set of active alarms.
If the alarm (or set of alarms in child tags) includes a shelved alarm, then the widget will be activated and display an S. If the linked tag (or child tags) includes a disabled alarm, the widget will display an X.
If linked to a parent tag, and within the set of child tags there are unacknowledged alarms, active (but acknowledged) alarms, shelved alarms and disabled alarms, then the order of precedence for the display is as follows:

1. Unacknowledged (icon blinks with the highest priority alarm of the set)
2. Active (icon shows the highest priority alarm of the set)
3. Disabled (icon shows an X)
4. Shelved (icon shows an S)

This is identical to the Alarm Priority Icon, but adds a box that you can place around the image representing the equipment that will trigger the alarm.
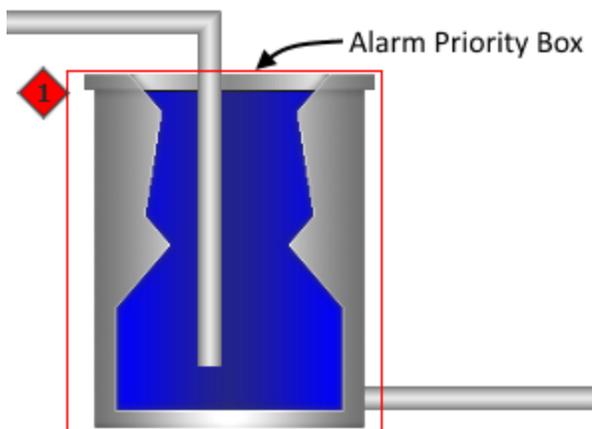
The icon is visible only when the linked tag is in an active alarm state. The shape, color, and text shown in the icon are controlled by the configuration of the Alarm Priority Tag used for that alarm.

As with any other widget, operators can right-click to open a pop-up menu for the linked tag, provided that they have sufficient privileges, and that you have not configured the icon to disable the pop-up menu. This menu can be used to navigate to the Alarm Page, where they can acknowledge the alarm. Note that only the icon is a valid target for the right-click; the box portion of this widget will not respond to operator actions. This is done so that the box does not interfere with controls or linked widgets within its area.

The properties dialog for the Alarm Priority Box:
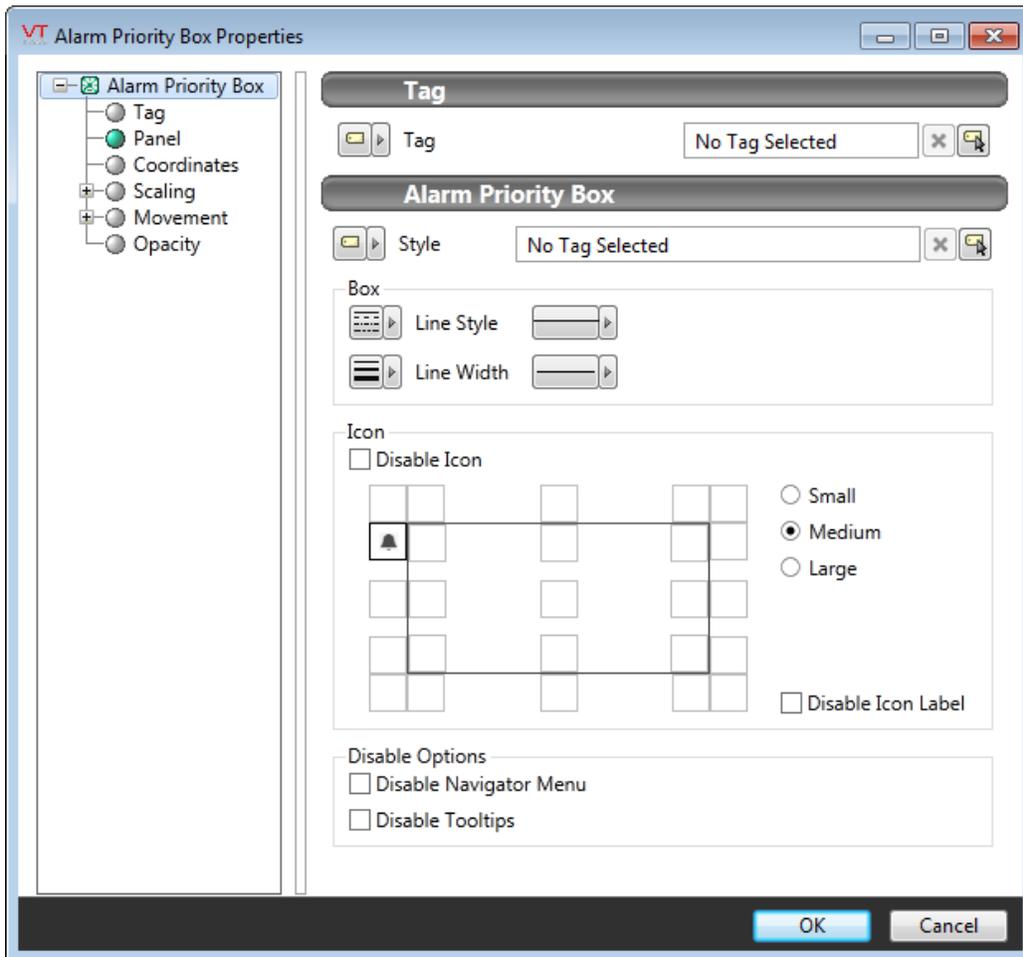
Tag

Link this widget to the tag whose active alarm state is to be signaled by the icon.

Style

May be linked to a Style Settings tag to control whether the icon should blink while the alarm is unacknowledged. Only the Alarm Priority affects the color of this widget, not the linked Style Settings tag.

Disable Icon

Select this option to remove the icon, leaving only the box to signal the active alarm state.

Icon – Position grid

Not available if the icon has been disabled. Click any of the squares to place the icon at that position relative to the box.

Icon – Small / Medium / Large

Not available if the icon has been disabled. Select the size of the icon to be displayed. This option defaults to medium. Choose a smaller size to avoid overlap with other objects in a crowded page, or a larger size to improve the visibility of the indicator.

Disable Icon Label

Select this option to hide the text portion of the alarm priority.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Tools\Alarm Tools\

Related Information:

...Alarm Priority Icon

...Alarm Priority Tags

...Choices for Operator Interaction

...See: "AlarmPriorityIndicatorShowNormalUnacked" in the VTScada Admin Guide

Alarm Priority Icon

Provides a visible indication when the linked tag is in an active or unac-knowledged alarm state, and when it is either shelved or disabled.
The icon will not be visible for alarms that are unacknowledged but no longer active unless the property AlarmPri-orityIndicatorShowNormalUnacked is set to 1. (Note that "trip" alarms are never in an active state.) The icon will always be displayed when the alarm is shelved or disabled.

You can link this widget to a parent tag rather than a specific status tag. When linked to a parent, it will become activated when any child tag in the hierarchy is in an active alarm state. The icon will display the highest priority of the set of active alarms.
If the alarm (or set of alarms in child tags) includes a shelved alarm, then the widget will be activated and display an S. If the linked tag (or child tags) includes a disabled alarm, the widget will display an X.
If linked to a parent tag, and within the set of child tags there are unac-knowledged alarms, active (but acknowledged) alarms, shelved alarms and disabled alarms, then the order of precedence for the display is as follows:

1. Unacknowledged (icon blinks with the highest priority alarm of the set)
2. Active (icon shows the highest priority alarm of the set)

3. Disabled (icon shows an X)
4. Shelved (icon shows an S)

Place the Alarm Priority Icon on a page, close to whatever image or shape represents the equipment that will trigger the alarm.
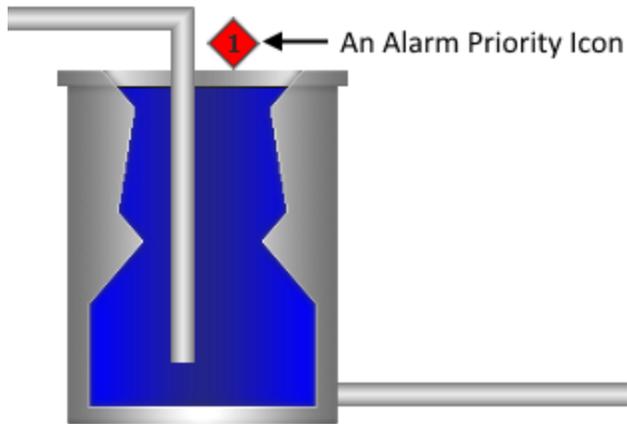
The icon is visible only when the linked alarm is in an active state. The shape, color, and text shown in the icon are controlled by the configuration of the Alarm Priority Tag used for that alarm.

As with any other widget, operators can right-click to open a pop-up menu for the linked tag, provided that they have sufficient privileges, and that you have not configured the icon to disable the pop-up menu. This menu can be used to navigate to the Alarm Page, where they can acknowledge the alarm.

The properties dialog for the Alarm Priority Icon:

Tag

Link this widget to the tag whose active alarm state is to be
signaled by the icon.

Style

May be linked to a Style Settings tag to control whether the
icon should blink while the alarm is unacknowledged. Only the
Alarm Priority affects the color of this widget, not the linked
Style Settings tag.

Disable Icon Label

Select this option to hide the text portion of the alarm priority.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tools\Alarm Tools\

Customized Alarm Display

You can add an alarm list, configured to show some or all of its tools, to any page in your application. The list can be customized in advance so that operators will see only the list relevant to that page.
All the tools required to work with an alarm list are built into the list widget, but you might decide to suppress everything other than the alarm display and provide additional buttons that perform only specific tasks. For example, rather than providing the full toolbar, you can suppress it and provide an Acknowledge Shown button and a Mute button.
The Alarm Tools folder contains a Legacy Tools sub-folder. These provide backward-compatibility with legacy applications (those built using VTScada 11.1 or earlier), which used an older form of the Alarm List.

> Note: Most of these tools are designed to interact with a named alarm list that you have drawn. Therefore, you are advised to start your custom alarm display with an Alarm List, configured to have a unique name.

...Ack Selected Button – Acknowledge a single, selected alarm from a named Alarm List.

...Ack Shown Button – Acknowledge all alarms shown in a named alarm list.

...Legacy Alarm Banner – Creates a display of alarms that cannot be filtered or sorted.

...(Legacy) Alarm Date Selector – Choose which date to view. If alarm history is stored in multiple files, the Next and Previous Log File buttons may be used.

...(Legacy) Alarm Go To Most Recent Button – Provides a quick way to reset the list.

...Alarm List Button – Change the Alarm List to display a different category of alarms.

...(Legacy) Filter Dialog Button – Opens a dialog to apply one or more filters to the selected Alarm List.

...(Legacy) Filter Field Button – Filter the current alarm list by a single, pre-selected option.

...(Legacy) Log File Date – Displays the date of the log file being viewed when the list shows alarm history.

...Mute Button – Silence all alarms, both old and new.

...(Legacy) Next Log File Button – Step to the next file of logged alarms when viewing alarm history.

...(Legacy) Prev Log File Button – Step to the previous file of logged alarms when viewing alarm history.

...Silence Button – Silence the most recent unacknowledged alarms.

...(Legacy) Sort Button – Sort the alarm list by any numeric property.

Analog Clock

Not linked to any tag
* Does not use the Stye Settings tag.

The Analog Clock widget is a dynamic image, displaying the current time.

The properties dialog for an Analog Clock:

The only configurable property for the clock is the selection of time zone.

Widget Folders:
Basic Components\Clock\
Tools\Standard Library\

Animated Image Widget

Used by: Alarm tag, Alarm Status tag, Analog Control tag, Analog Input tag, Analog Output tag, Analog Status tag, Calculation tag, Comm Link Sequencer tag, Counter tag, Deadband Control tag, Digital Control tag, Digital Input tag, Digital Output tag, Digital Status tag, Function tag, History Statistics tag, Modem tag, Network Status tag, Pump Status tag, Rate of Change tag, Script tag, Selector Switch tag, SMS Appliance tag, SQL Logger tag, Totalizer tag, Trigger tag.
* Does not use the Stye Settings tag.
The Animated Image widget is used to display a short movie that runs while the tag's value is non-zero. When used for tags that have analog

values, the speed of the animation reflects the value of the tag. For tags with simple on or off values (digitals) the animation either runs or not. In all cases, you can set the maximum speed of the animation in frames per second when configuring the widget.

A configured Animated Image (shown in multiple frames).

Animated Images do not use animated GIF files or other single-file animations.

Note: You should NOT use the Animated Image or the Color Blink widgets on pages that will be accessed via a VTScada Internet Server, due to possible latency and bandwidth issues. If you must use Animated Image widgets, please ensure that you set the frames per second rate low.

The Animated Image properties dialog will look like the following:
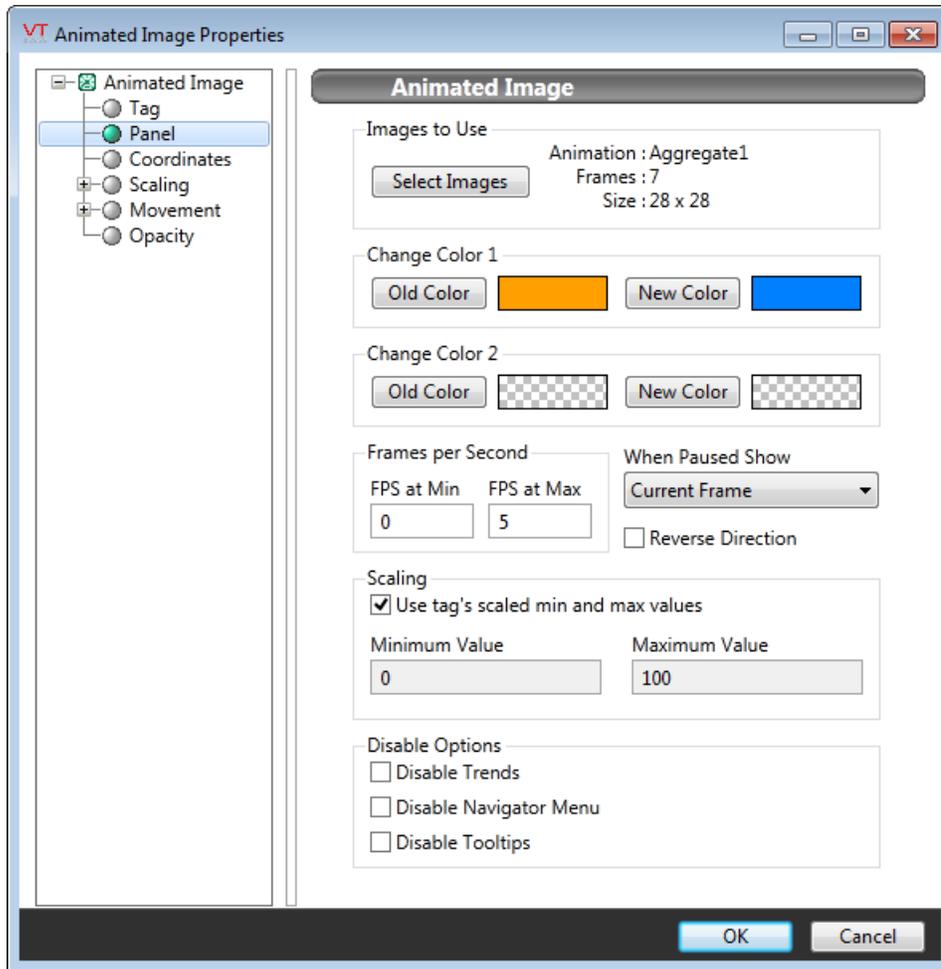
Tag

The Tag field displays the name and description of the tag that is linked to this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Select Images
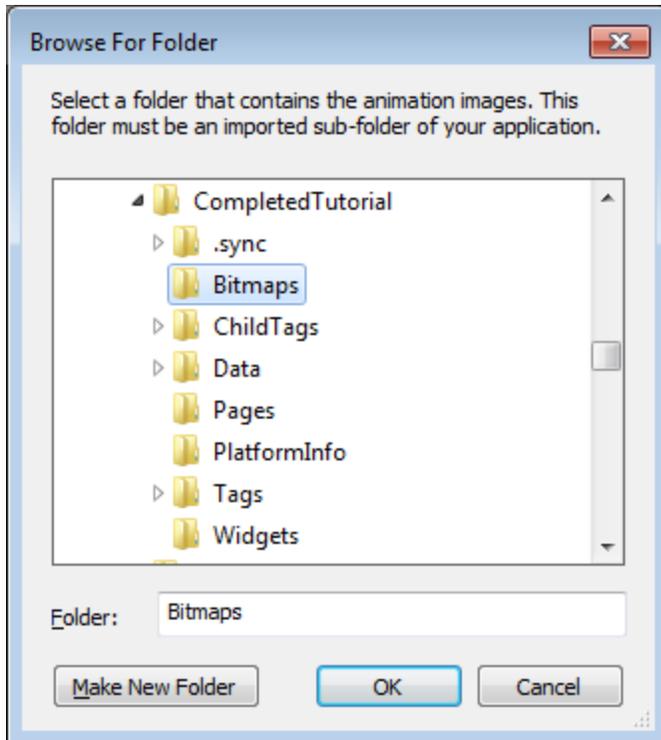
Opens the Browse for Folder dialog, which you can use to select any folder containing a custom animation. The folder must be within your application's folder structure. Use this option to select a custom animation, not to a different VTScada animation to another.

The folder and its files must have previously been imported into your application. Refer to the topic, Adding Custom Animations.

Old Color (1 and 2)

Used in conjunction with the New Color button. Select a color that will be replaced with another in the image. You may replace up to 2 colors.

For example, several of the provided animations have been created with active orange (RGB 255, 160, 0) so that they may be overlaid with a Color Change widget to dynamically adjust their color. You might wish instead to use blue for the image.

To do this, select 255, 260, 0 (active orange) as the old color and blue as the new.
The tool used to choose a color is the Select Color Dialog.

New Color (1 and 2)

Used in combination with the Old Color button. Select a new color to replace another in the image. You may replace up to 2 colors.

FPS at Minimum and at Maximum

Use these fields to set the minimum and maximum animation speeds relative to the associated tag's scaled minimum and maximum values. The animation speed is measured in frames per second (FPS). The values you set here must be within the range of 0 to 30, but there is a way to override these by applying your own minimum and maximum scaling
The default minimum value is 0, indicating that the animation should be stopped when the value of the associated tag is at its minimum. The default maximum is 15, which generally provides an acceptable animation without excessively taxing either your computer's CPU or display card.

Reverse Direction of Animation

Select this option to run the animation from the last frame to the first frame, rather than from the first frame to the last frame, thereby reversing the apparent direction.

When Paused Show

Select which frame you wish the Animated Image object to show when it is paused. This can be one of:

- Current Frame: The frame of the animation that last played will be displayed when the animation is paused.
- First Frame: The first frame of the animation will be displayed when the animation is paused.
- No Image: No image will be displayed when the animation is paused.

Use Tag's Scaled Min and Max Values

Selected by default. Indicates that the tag's minimum and maximum scaled process values should be used. If you deselect this box, then you may enter your own scaling values for the tag by using the Minimum Value and Maximum Value fields (see following). These values will only apply to this one animation of the tag.

The net effect of scaling, and therefore the speed at which the animation will run, will depend on the relationship between the tag's scaling values and those that you set here.

For example, given the following:

- A tag with scaled values from 0 to 100.
- A FPS rate at minimum of 0.
- A FPS rate at maximum of 30.

Setting your own Minimum and Maximum scale values will change the FPS rates as follows:

- Minimum set to 0 and Maximum set to 50: FPS at minimum is 0 and at maximum is 60.
- Minimum set to 0 and Maximum set to 200: FPS at minimum is 0 and at maximum is 15.
- Minimum set to -100 and Maximum set to 100: FPS at minimum is 15 and at maximum is 30.
- Minimum set to 50 and Maximum set to 100: FPS at minimum is -15 and at maximum is 30.

Minimum Value and Maximum Value

These two fields are enabled only when the Use Tag's Scaled Min and Max Values check box is not selected. See Use Tag's Scaled Min and Max Values for a full explanation of the effect of these values.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Animations\

Related Information:

...Choices for Operator Interaction

...Color Blink Widget

...Adding Custom Animations

...Active Orange

Adding Custom Animations

Animated graphics are an ordered series of individual graphic images, each of which differs from the last by a small amount. When viewed in rapid succession, the visual effect of the differences will appear as motion to the human eye.

General Steps
  (Details are covered in the following notes)

1. Add a folder below your application's Bitmaps folder. Name it for the animation you will create.
2. Create an image for each frame of the animation.
3. Open the VAM and use the application's Import File Changes button to import both the folder and the image files.
4. Back in the Idea Studio, add an animation widget to a page.
   Any animation may be chosen, since you will change it in the next step.
5. Open the Properties dialog for the animation widget.
6. Click the Select Images button.
   A Select Folder dialog opens.

7. Browse to and select the folder containing your animation images.

8. Click OK. Your animation replaces the original.

Image File Details:

There are two parts to a VTScada animated image:

- The directory it is stored in.

- The file names used by the still images.

All should share a common root file name. Any of the file formats recognized by VTScada may be used.

> Note: Your VTScada application will not use new files and folders until they have been imported by someone who has the Edit Files privilege.

For example, if you were creating an animation named "Bounce," you would first create a sub-folder of the application's Bitmaps\ folder with that name.

C:\VTScada\YourApp\Bitmaps\Bounce

Custom images should always be stored in your application directory structure, or the OEM directory that your application is based on.

The still frames of the animation will be a series of image files that you save to the Bounce folder.

> Note: The color black (#000000) will be treated as transparent in VTScada. Do not use pure black in your animation's images for any area that should be visible.

The images are ordered by adding a sequence character (A, B, C …) after each file name. For example:

C:\VTScada\YourApp\Bitmaps\Bounce\BounceA.png

C:\VTScada\YourApp\Bitmaps\Bounce\BounceB.png

C:\VTScada\YourApp\Bitmaps\Bounce\BounceC.png

The sequence character need not be upper-case. Capital letters are used here for ease of viewing.

There should be a minimum of three still frames in order to create an effective animation. The longest animation that comes with VTScada has 18 frames. Most have four to six.

Related Information:

...See: "Import/Export Files" in the VTScada Admin Guide
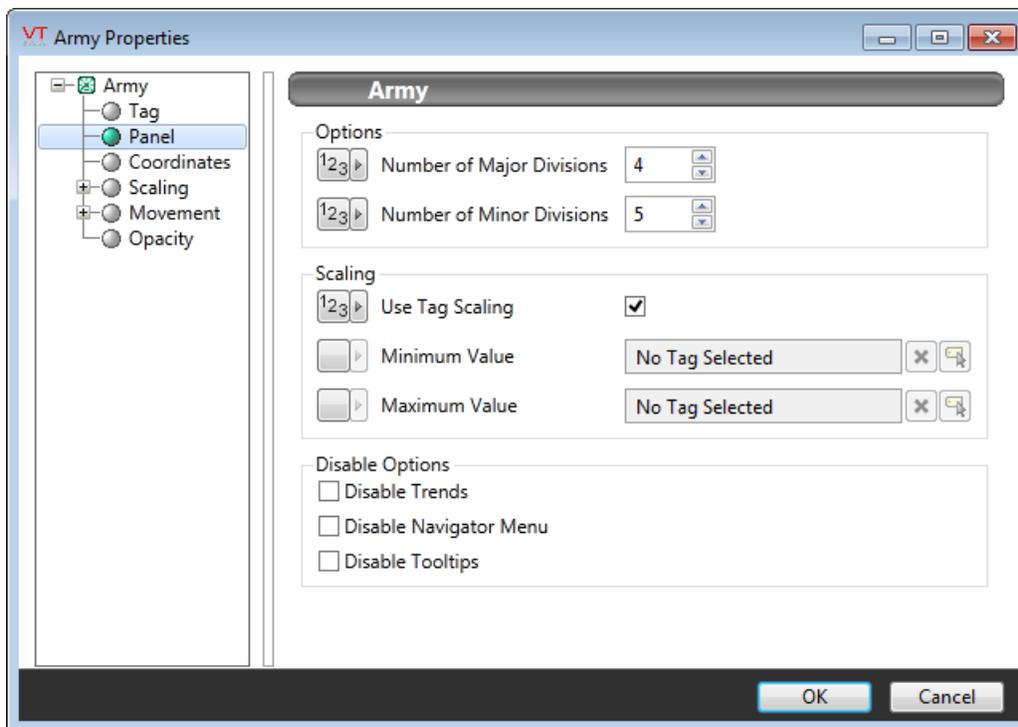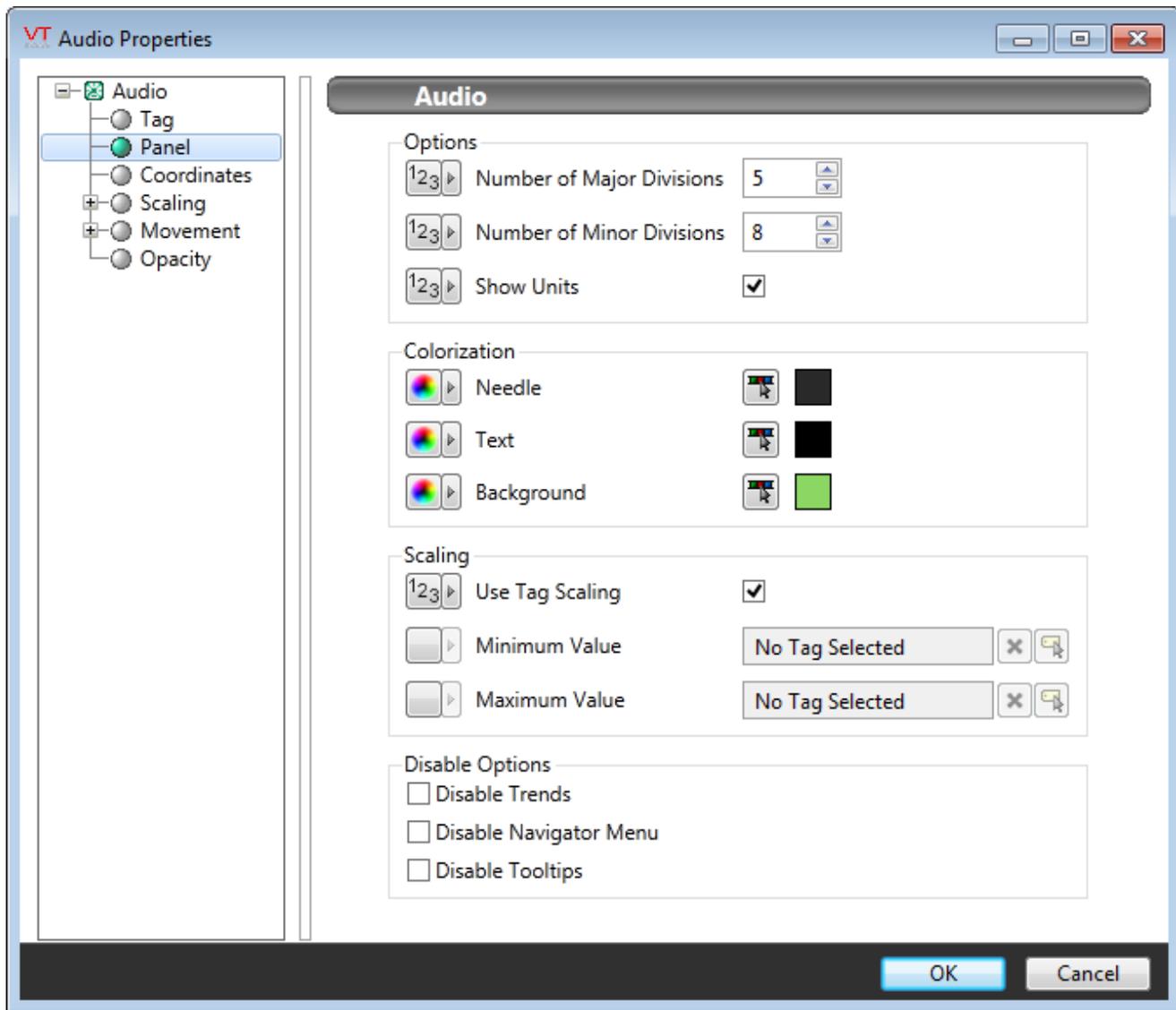
Antique Bolted Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Antique Bolted widget represents the associated tag's value as a photorealistic meter

The properties dialog for the Antique Bolted widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and 5.

Choose whether to include the text used for engineering units within the gauge. Defaults to show units.

Colorization

Use the color selector dialogs to set the color of the needle, text and gauge background. This widget does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

*Related Information:*
Choices for Operator Interaction

Antique Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
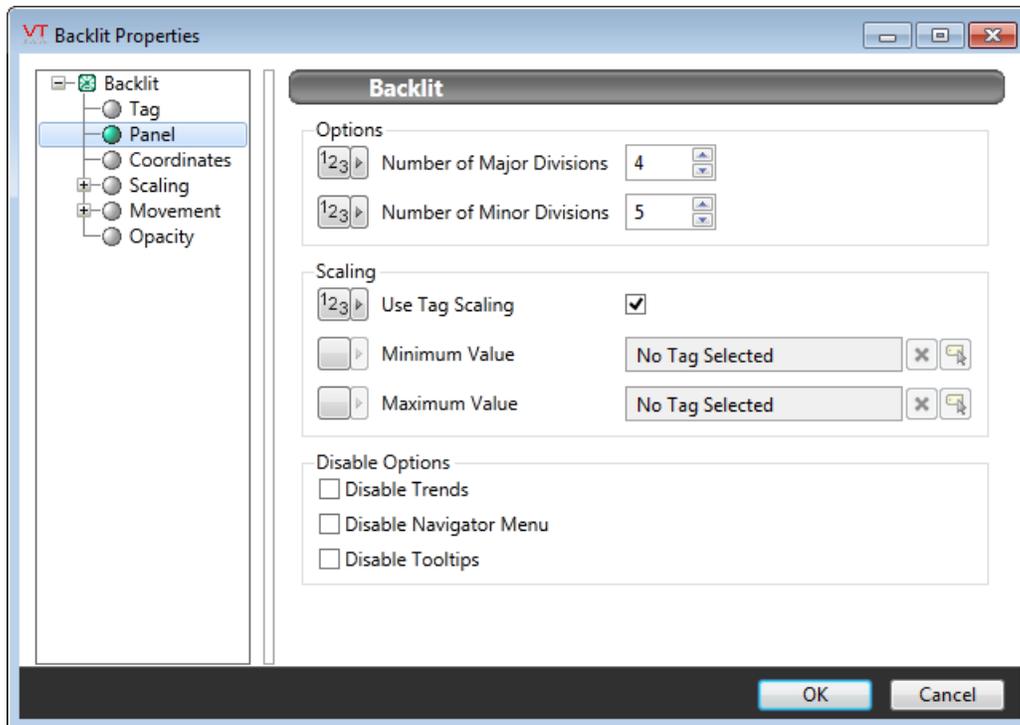* Does not use the Stye Settings tag.
The Antique Gauge widget represents the associated tag's value as a photorealistic meter. The divisions used to display the scale range are

pre-set for this meter. The engineering units of the linked tag are dis-
played below the scale.

The properties dialog for the Antique Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the
meter. Each labeled division will also be drawn with a longer

line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Aqua Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Aqua Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Aqua Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

...Choices for Operator Interaction

Arc Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.

The Arc widget represents the associated tag's value as portion of an arc. The darker portion of the arc starts at the angle you specify, where vertical (North) is zero degrees. Increasing values are indicated by the darker portion sweeping in a clockwise direction. The colors are set by the LED (analog) values of a selected Style Settings tag, and values above the linked tag's alarm setpoint can be indicated.
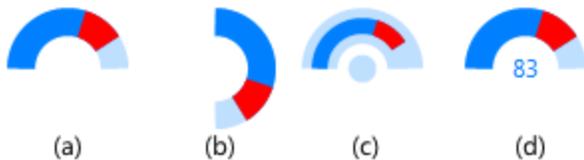
Four variations on a theme, showing the control you have over arc configuration.

The properties dialog for the Arc widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

The Tag field is only present when editing a completed widget object.

Style

Optionally, select a Style Settings tag to control the colors

shown by the arc, in response to the linked tag's value and alarm state. Style Settings Hierarchy.

Minimum Angle & Maximum Angle

Sets the starting point and the ending point for the sweep of the arc. Defaults to 270 degrees for the start angle and 90 degrees for the end angle. The angle is measured in degrees, and increases in the clockwise direction.
Example (b) shows an arc with minimum angle zero and maximum angle of 180 degrees.

Lines

Use to adjust the width of the unlit (background) and lit (foreground) portion of the arc.
Example (c) shows an arc with an unlit width of 18 pixels and a lit width of 8 pixels. The alignment is Center. Alignment applies only when the lit and unlit widths are of differing values, giving you the option of aligning the lit portion against the outside edge, inside edge, or centered within the unlit portion.

Show Alarms

If selected, the arc will indicate when the associated tag is in an active alarm state by using the alarm indication settings as configured in the associated Style Settings tag. Only that portion of the sweep above or below the alarm setpoint will use the alarm indication.
This option is in use for all four of the example images.

Counter Clockwise

If selected, sections of the arc will sweep (increase) in a counter-clockwise direction rather than the standard clockwise.

Show Center Circle & Spacing

These two properties should be used together. The center circle will be displayed using the unlit color of the arc. The spacing controls the size of the circle by setting the number of pixels from the unlit ring of the arc to the perimeter of the circle.
Example (c) shows a center circle in addition to showing lines with varying widths.

Show Value

If selected, the arc will display the current value of the linked tag. Units are not shown.The color will always match that of the lit portion of the arc.If a center circle is shown, then the value will be superimposed on the circle.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
Choices for Operator Interaction

Army Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Army Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Army Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Audio Meter Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Audio widget represents the associated tag's value as photorealistic meter.

The properties dialog for the Audio widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and 5.

Choose whether to include the text used for engineering units within the gauge. Defaults to show units.

Colorization

Use the color selector dialogs to set the color of the needle, text and gauge background. This widget does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
Choices for Operator Interaction

Backlit Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
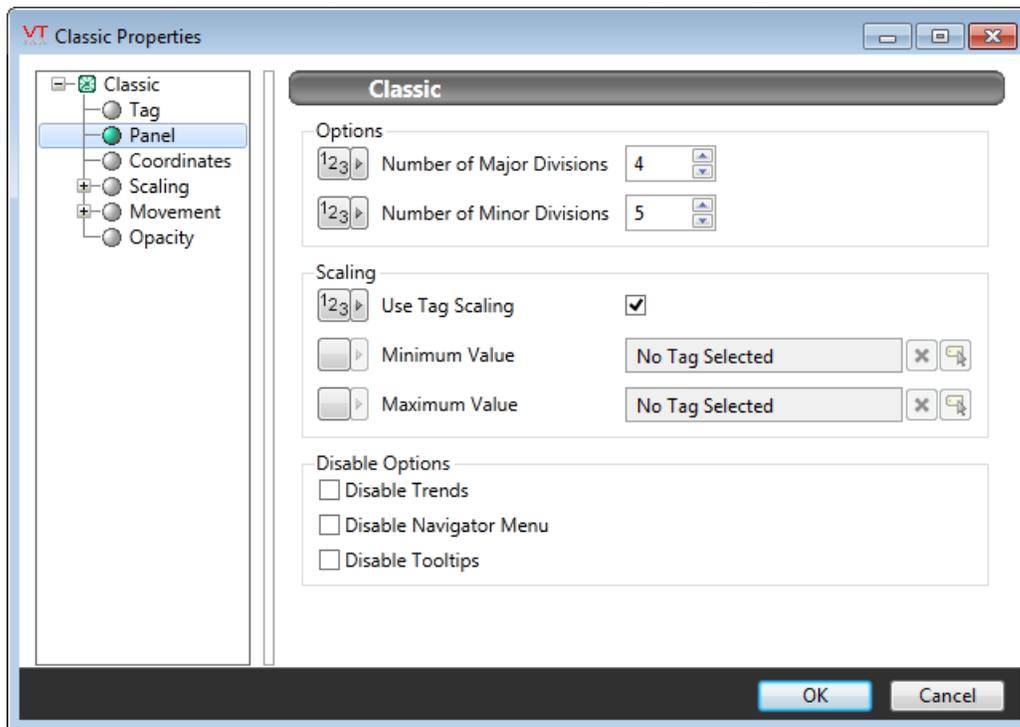* Does not use the Stye Settings tag.
The Backlit Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and

unlabeled minor divisions, the dial of which fluctuates according to the tag's value.
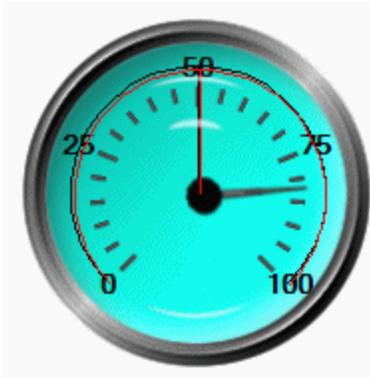
The properties dialog for the Backlit Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:
...Choices for Operator Interaction

Bottom Bar Widget
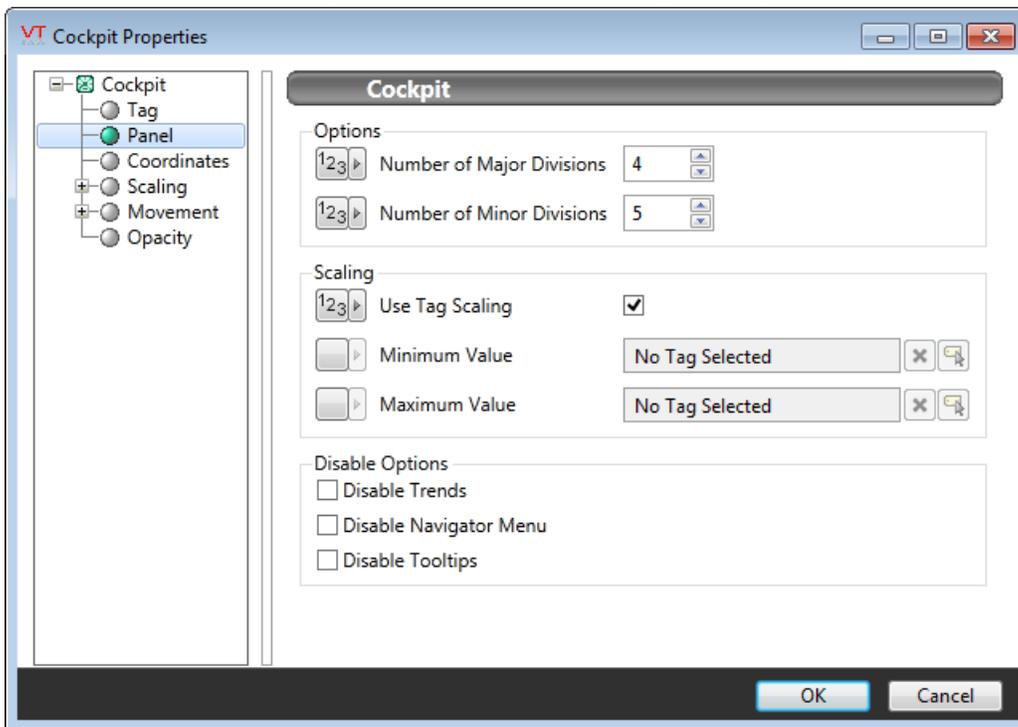
Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Deadband Control,

Function, History Statistics, Script, Selector Switch, SQL Logger.

* Does not use the Stye Settings tag.

The Bottom Bar widget is used to display a tag's value as a vertical bar, the length of which increases from the top towards the bottom as the tag's value increases. When the tag's value is at a minimum, the bar will be shown as a thin line at the top of its range. When the tag's value reaches its maximum, the widget will be displayed as a rectangle, filling the range.

at minimum   at maximum   being edited at 54%
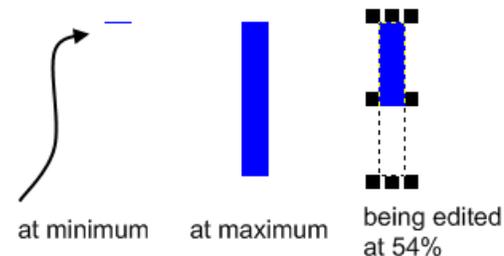
The properties dialog for the Bottom Bar widget will look similar to the following:

Set Color

Opens the Select Color Dialog from which you can select a color for this object.

Low Scale Range and High Scale Range

In most cases, these fields will be display-only, showing the Scaled Process Data Min and Scaled Process Data Max values configured for the associated tag.

These fields become editable when the widget is used to display a function or Calculation tag. The most common use is to provide an output range of 0 to 100 (%) given a data source whose numbers do not match these values (perhaps ranging from 20 to 80, for example). Set the low scale range to the value expected from the data source when it is at its minimum and the high scale range to the value expected from the data source when it is at its maximum.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Gauges\Basic Analogs\
Basic Components\Bars\

Related Information:

...Choices for Operator Interaction

Checkbox Switch Widget

Used by: Selector Switch.
* Does not use the Stye Settings tag.
The Checkbox Switch widget is used to create an On/Off control for a two-position SelectorSwitch tag. Each time an operator clicks on the box or attached label, the switch will toggle between positions and a check mark will be displayed for position 1.
This is a control-type widget that is only available for selector switches – it is not available for use as an indicator with Digital Input or Digital Status tags.

> Note: When displaying more than one check box on a page, each one must be given a unique value for its Focus ID.

☑ Automatic Mode

The properties dialog dialog for the check box is as shown:
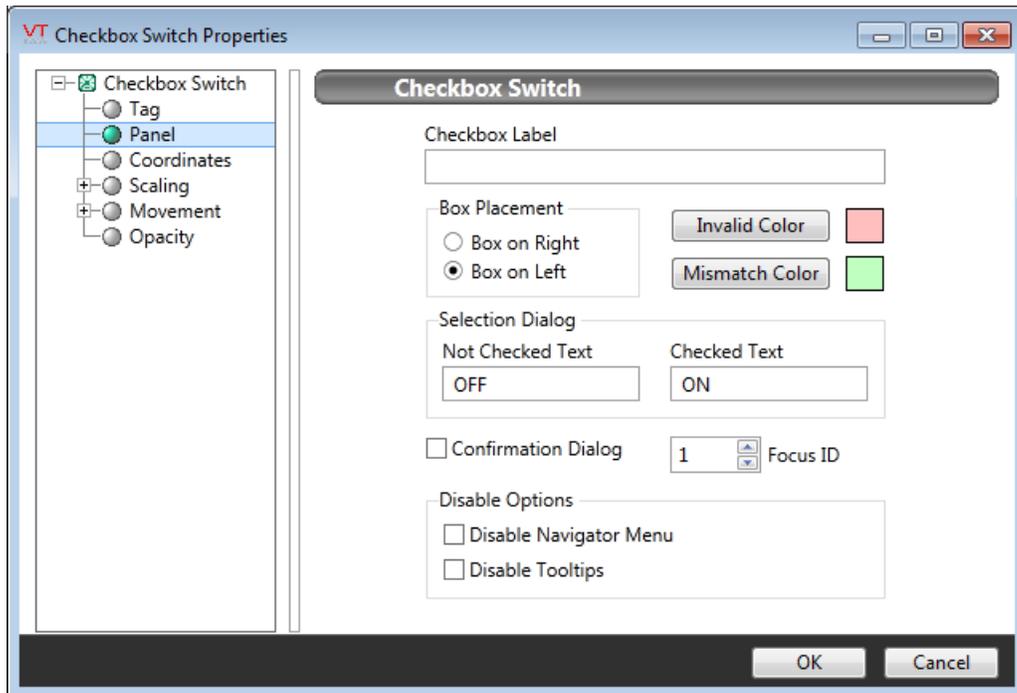
CheckBox Label

Provide a descriptive label for the check box. May be left blank.

Box Placement

Select whether the box is to go to the right or the left of the label. The label itself will change justification to match the box (left-justified when the check box is on the left and right-justified when the box is on the right).

Invalid Color

Sets the color that will be displayed when the attached Selector Switch tag is in an Invalid state. May indicate either an error in

communication or a Selector Switch with no feedback that has not yet been set to any position.

Mismatch Color

Sets the color that will be displayed when the Selector Switch's position feedback indicates that the equipment is in a state other than the one most recently set using this widget.

Selection Dialog

When the check box is clicked for the first time and there is no feedback to show the current position of the Selector Switch tag, a dialog box will open with a choice of initial state. In that dialog, the "Not Checked Text" describes position 0 and the "Checked Text" describes position 1 of the Selector Switch.

Confirmation Dialog

If selected, the operator will be prompted to confirm every change made by clicking on the buttons.
An example of the confirmation dialog is as follows:
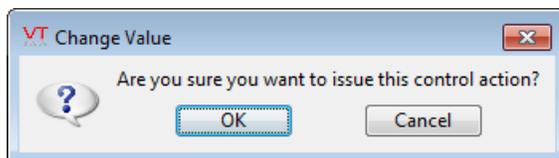
Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)
When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the

page that can receive focus. (i.e. the first graphics statement in the page's source file).
You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

CIP Control Widget

Used by: CIP Driver.
* Does not use the Stye Settings tag.
The CIP ENIP Control tag widget creates a panel of controls that can be used to send output to the hardware associated with a CIP driver tag.

There are no configurable options for the CIP Control panel.
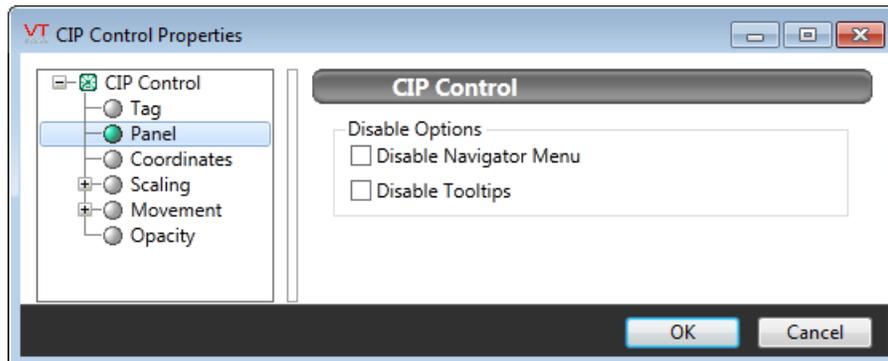
Analytics\Communications\

Analytics\Diagnostics\

CIP Information Widget

Used by: CIP Driver.

* Does not use the Stye Settings tag.

The CIP Info tag widget is used to create a beveled display, listing inform-
ation about the associated hardware.

```
┌─CIP_ENIP_Driver_01 Information ─────────────┐
│ Connect Status:                             │
│ Session Handle: Invalid                     │
│ Vendor ID: Invalid                          │
│ Device Type: Invalid                        │
│ Product Code: Invalid                       │
│ Revision: Invalid                           │
│ Status: Invalid                             │
│ Status: Invalid                             │
│ Product Name: Invalid                       │
│ State: Invalid                              │
│                                             │
└─────────────────────────────────────────────┘
```
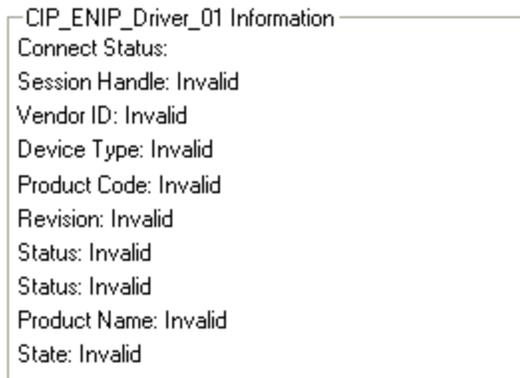
The CIP Info properties dialog:
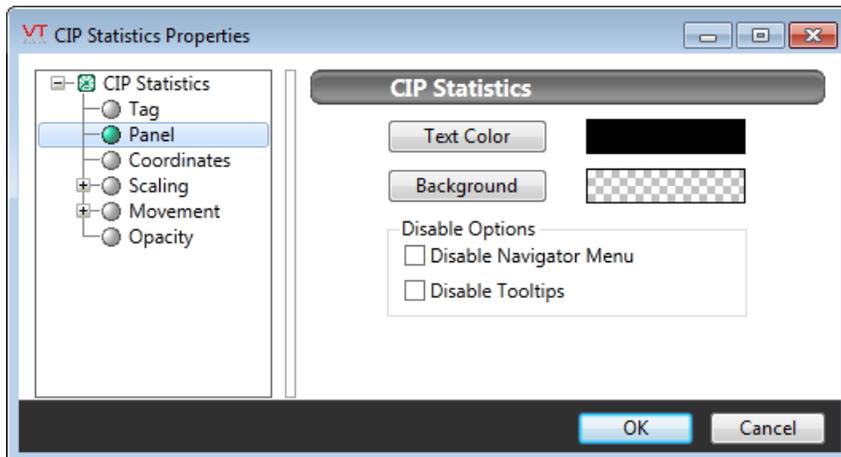
Text Color

Sets the color in which to display the text labels associated
with this widget.

Background

Sets the background color for the completed object. Note: The background color you choose should provide a good contrast for the text color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Communications\
Analytics\Diagnostics\

CIP Statistics Widget

Used by: CIP Driver.
* Does not use the Stye Settings tag.
The CIP Stats tag widget is used to create a beveled display, listing the statistics about the communications between VTScada and the associated hardware.
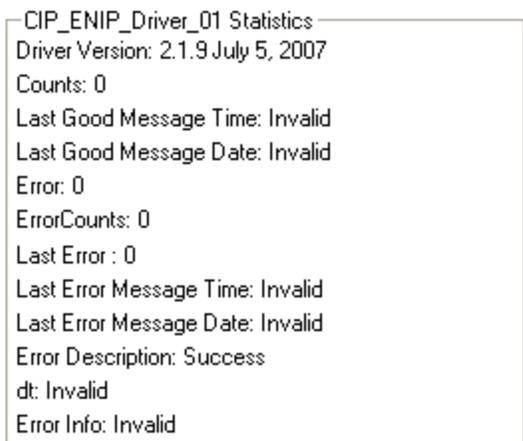
The CIP Stats properties dialog:
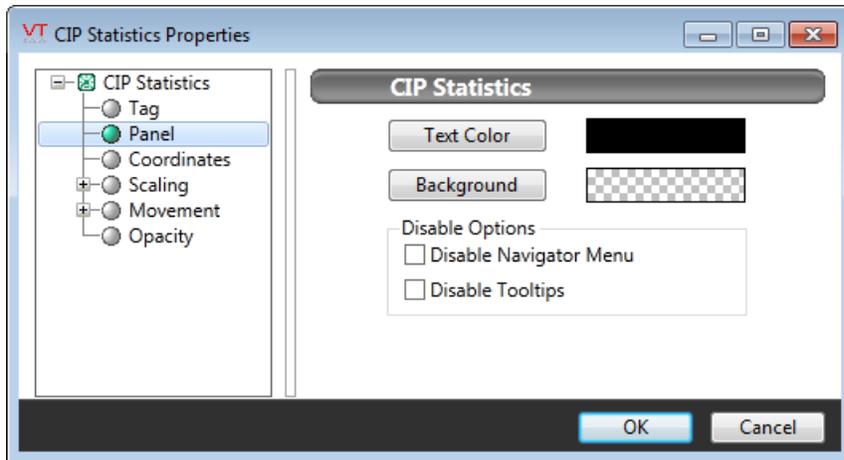
Text Color

Sets the color in which to display the text labels associated with this widget.

Background

Sets the background color for the completed object. Note: The background color you choose should provide a good contrast for the text color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Communications\
Analytics\Diagnostics\

Classic Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Classic Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and

unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Classic Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:
...Choices for Operator Interaction

Cockpit Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.

* Does not use the Stye Settings tag.

The Cockpit Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Cockpit Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer

line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Color Blink Widget

Used by: Alarm, Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Counter, Deadband Control, Digital Control, Digital Input, Digital Output, Digital Status, Network Status, Pump Status, Rate of Change, Selector Switch, Totalizer, SQL Logger, Totalizer, Trigger, .
* Does not use the Stye Settings tag.
The Color Blink tag widget is used to create a color box that blinks to indicate the state of the underlying tag. Blinking is an extremely eye-catching visual feature and should only be used in situations where it is of utmost importance for an operator to notice the state of a tag. Do not use this widget casually.

Shown within a rectangle
for clarity.

You can configure two colors for each state: the status color, and the blink color. The widget will alternate between these colors according to the On Time and Off Time configured for each state.
If one of the colors is left unconfigured, or set to transparent, the object will seem to appear and disappear. If the colors are identical, nobody will notice the blinking. In general, it is best to use either a light and dark shade of the same hue, or to leave one of the colors transparent or unconfigured.

Note: If a page is intended to be viewed on a VTScada Internet Client, it is recommended that Color Blink objects and Animated Bitmap objects not be used. This is due to possible latency and bandwidth issues. If you must use Color Blink objects, please ensure that you set slow blink rates.

An example of a configured properties dialog for a Color Blink.
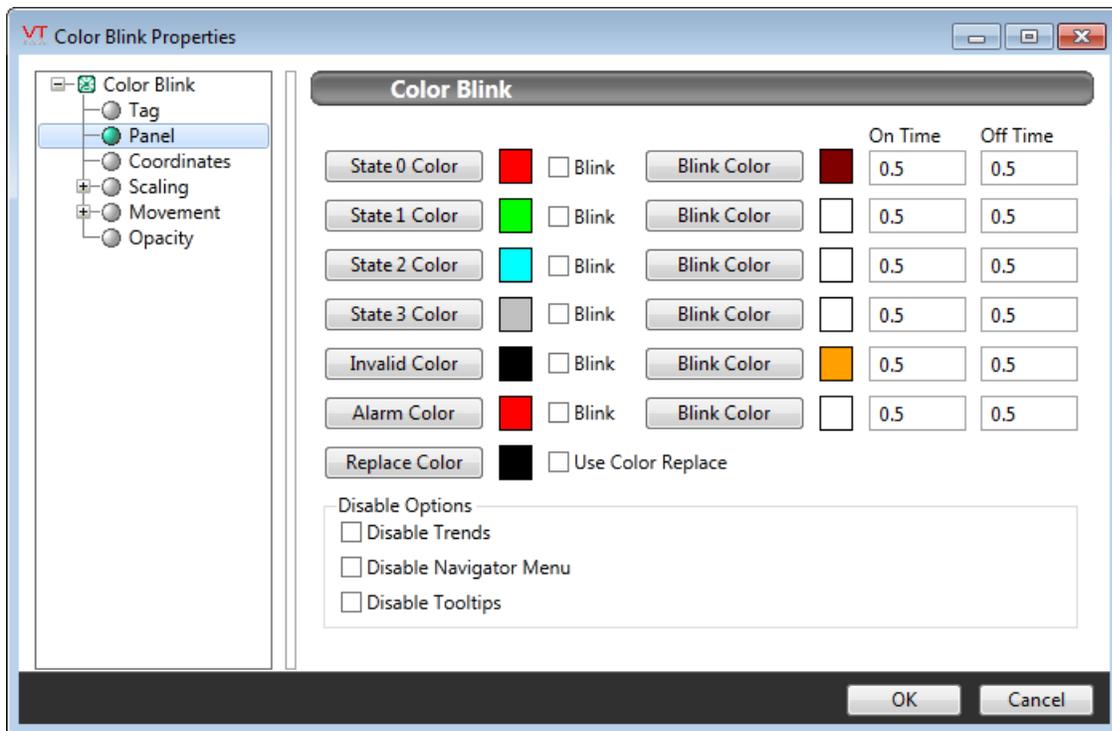
The color blink widget is used most effectively with digital tags, or tags that have binary states such as alarms. If used with an analog tag, the tag's value will be rounded to obtain the closest value of 0, 1, 2 or 3 (repeating).

Note: "2 state" versus "4 state". Not all tags will have all the states shown in these examples. Alarms and single-bit digitals have only state 0 and state 1 while 2-bit digitals also have states 2 and 3. The alarm state applies both to tags that have built-in alarms and to tags that have alarms attached to them. The alarm state will be active as long as the corresponding alarm is unacknowledged, regardless of whether the alarm is active.

State x Color

Opens the Select Color Dialog from which you can choose a color to be displayed when the tag's value is in the corresponding state.

Blink

This check box must be selected if you wish the completed Color Blink object to alternate (blink) between the State X Color and the corresponding Blink Color.

Blink Color

Opens the Select Color Dialog from which you can choose a blink color to be displayed when the tag's value is in the corresponding state.

On Time

Sets the amount of time (in seconds or fractions of a second) that the Blink Color should be displayed.

Off Time

Sets the amount of time (in seconds or fractions of a second) that the State X Color should be displayed.

Alarm Color

Opens the Select Color Dialog from which you can choose a blink color to be displayed when the tag's value is in an alarm state.

Replace Color

Opens the Select Color Dialog from which you can select the background color upon which the completed object will be placed. This object will only be visible when placed over the selected color.

Use Color Replace

Select to indicate that you wish to use the configured Replace

Color (see above). If the Use Color Replace check box is not selected, then the object will always be visible over any back- ground color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Color Indications\

Related Information:

...Choices for Operator Interaction

...Animated Image Widget

Color Box Widget

Used by: Alarm, Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Counter, Deadband Control, Digital Control, Digital Input, Digital Output, Digital Status, Function, History Statistics, Modem, Network Status, Pump Status, Rate of Change, Roster, Selector Switch, SMS Appliance, SQL Logger, Totalizer, Trigger.
* Does not use the Stye Settings tag.
The Color Box widget is used to represent the associated tag as a box that changes color based on the tag's value. It is primarily used with that have discrete states such as digitals and alarms, but may also be used with analogs. For an analog tag, the value will be rounded to find the closest match to one of the states.

Note: "2 state" versus "4 state". Not all tags will have all the states shown in these examples. Alarms and single-bit digitals have only state 0 and state 1 while 2-bit digitals also have states 2 and 3. The alarm state applies both to tags that have built-in alarms and to tags that have alarms attached to them. The alarm state will be active as long as

The properties dialog for a Color Box:

State X Color

Opens the Select Color Dialog from which you can choose a color to be displayed when the tag's value is in the corresponding state.

Invalid Color

Opens the Select Color Dialog from which you can choose a color to be displayed when the tag's value is invalid.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Color Indications\

Related Information:

...Choices for Operator Interaction

Color Fill Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband Control, Function, Rate of Change, Script, Selector Switch, Totalizer, SQL Logger.
* Does not use the Stye Settings tag.
The Color Fill tag widget is used to draw a tag as an animated bar (Bar Color) that graphically illustrates the value of the associated tag against a background color (Unfilled Color).
If the Changed Color is set, the object must be placed over a region of that color in order to be visible.

Color Fill
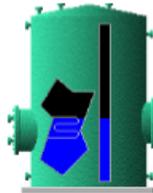
The same Color Fill used behind a tank bitmap

An example of the Color Fill properties dialog. There must be a Filled Color, but you are free to leave the Changed Color and Unfilled color properties unselected.
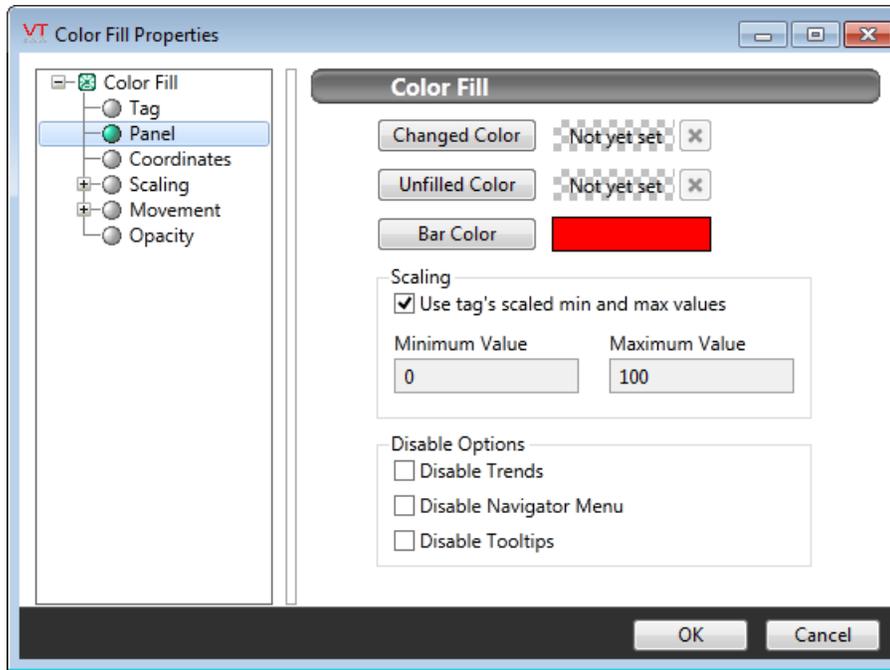
Changed Color

If set, the object will only be visible when placed above an area of this color. The button opens the Select Color Dialog from which you can choose a color. Click the X button to remove the color if you change your mind.

Unfilled Color

Opens the Select Color Dialog from which you can choose a color to be displayed as the background for the bar color.

Bar Color

Opens the Select Color Dialog from which you can choose a color to that will represent the tag's value.

Use Tag's Scaled Min & Max Values

In most cases, it is best to select this option. When the tag is at its minimum scaled output value, the bar will not be visible

and when the tag is at its maximum scaled output value, the bar will fill the area of the widget.

Choosing not to use the tag's scaled min and max values enables the Minimum Value and Maximum Value controls which you can use to scale the bar length relative to the tag's output scaled values. The scaling you set will apply only to this one instance of the tag widget. This option is normally used when the object represents tags such as Calculation and Function that do not have their own scaled values.

Unlike widgets such as the Top Bar, the bar for the color fill will be visible only within the area originally drawn for this object, regardless of the scaling you define.

Minimum Value and Maximum Value

The Minimum Value and Maximum Value fields are used to scale the color fill bar relative to the tag's scaled output values. For example, given a tag with scaled output ranging from 0 to 100:

Setting the widget's scale range from 0 to 200 results in a bar that reaches only half the length drawn when the tag is at its maximum value.

Setting the widget's scale range from 0 to 50 results in a bar that reaches its full length when the tag is still only half way to its maximum value.

Unlike widgets such as the Top Bar, the bar for the color fill will be visible only within the area originally drawn for this object, regardless of the scaling you define.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Bars\

Basic Components\Color Indicators\

Color Line Widget

Used by: Alarm, Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Alarm, Calculation, Counter, Digital Control, Digital Input, Digital Output, Digital Status, Function, History Statistics, Pump Status, Rate of Change, Selector Switch, Serial Port, SQL Logger, Totalizer, Trigger, UDP/IP Port.
* Does not use the Stye Settings tag.
The Color Line tag widget is used to draw a line that changes any combination of color, width and style to indicate the state of the associated tag.

The properties dialog for the Color Line:
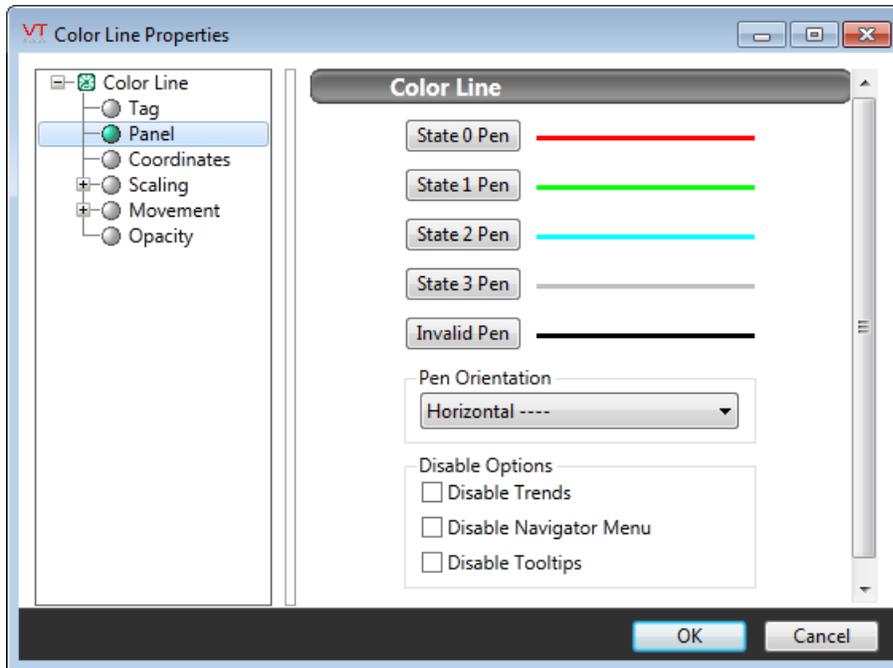
State X Pen and Invalid Pen

Opens the Select Pen dialog. From this dialog, you can select the color, line style and width that the Color Line object will use to show each state of the underlying tag.

Pen Orientation

Use the Pen Orientation drop-down list to select one of four orientations for the completed Color Line object (horizontal, vertical, left 45 degrees, or right 45 degrees). The orientation remains the same for all tag states.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Indicators\Basic Indicators\

Basic Components\Color Indications\

Related Information:

...Choices for Operator Interaction

Comm Indicator Widget

Used by: Data Flow RTU, Polling Driver, Allen Bradley Driver, CIP Driver, CalAmp Diagnostic Driver, DDE Client, DNP 3.0 Driver, Driver Multiplexer, MDS Diagnostic Driver, Modbus Compatible Device , Omron Host Link Driver, OPC Client Driver, Siemens S7 Driver, TCP/IP Port, Workstation Driver.
* Does not use the Stye Settings tag.
The Comm Indicator tag widget is used to draw an indicator square, the color of which indicates the current status of the associated I/O device.

The properties dialog for the Comm Indicator:
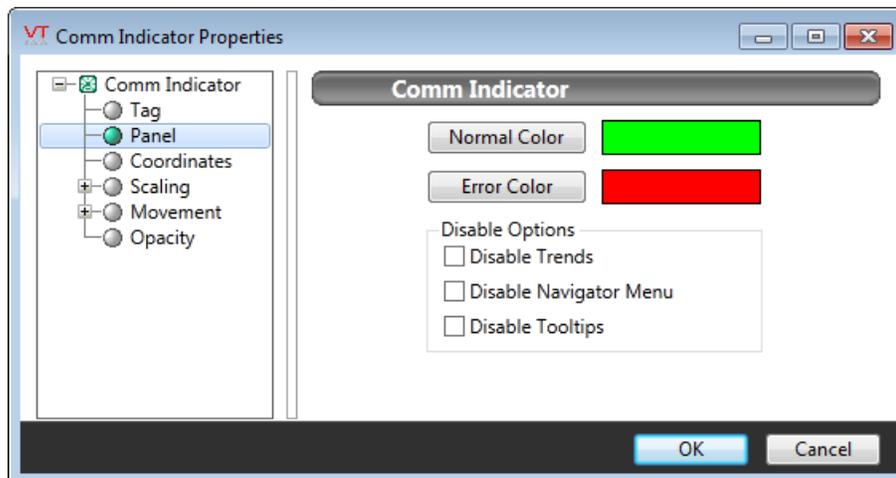
Normal Color

> The color to display when the associated I/O device is oper-
> ating under normal conditions

Error Color

> The color to display when the I/O device is in an error state.

Disable Options

Disable selected operator-interaction features of this widget.

Note: when used with the Data Flow RTU driver, the context menu for this widget will have an extra option: "Active".
Data Flow RTU drivers have their own polling feature which can activated or not through their configuration panel. Operators who do not have con-figuration privileges, but do have either "tag modify" or " manual data" security privileges can toggle the active polling state on or off by right-clicking on this widget and selecting the Active status.
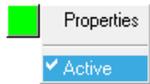
Widget Folders:
Analytics\Communications\Connection Status\
Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Comm Line Widget

Used by: Serial Port, TCP/IP Port, UDP/IP Port.
* Does not use the Stye Settings tag.
The Comm Line tag widget is used to draw a line that changes any com-bination of color, width and style to indicate the state of the associated tag.

The properties dialog for the Comm Line is as shown:
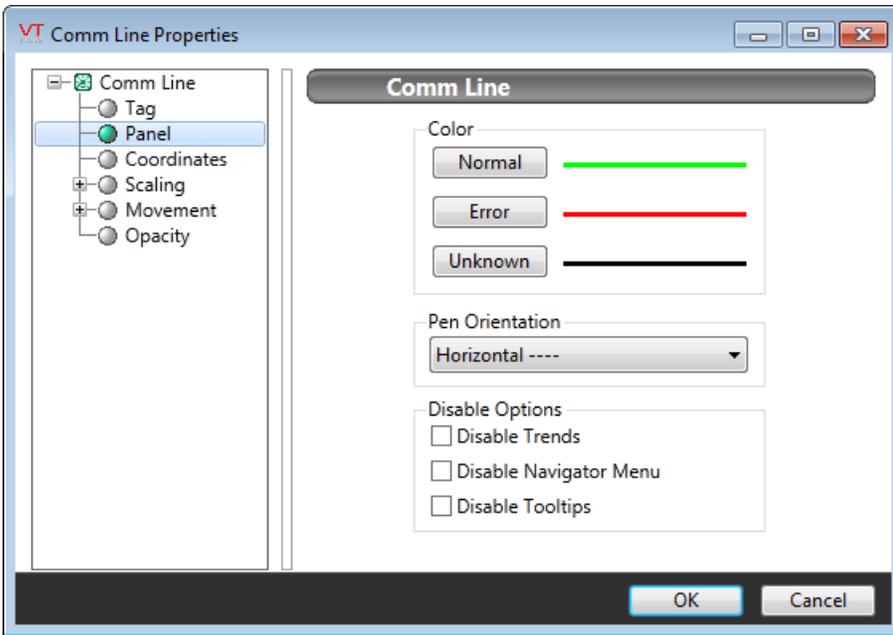
Color

Opens the Select Pen dialog. From this dialog, you can select the color, line style and width that the Comm Line object will use to show each state of the underlying tag.

Pen Orientation

Use the Pen Orientation drop-down list to select one of four orientations for the completed Comm Line object (horizontal, vertical, left 45 degrees, or right 45 degrees). The orientation remains the same for all tag states.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Connection Status\
Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Comm Messages Button Widget

The Comm Messages Button tag widget is used to draw a button that can be clicked to display communications data (transmit and receive communication message strings) for an associated communication driver tag. * Does not use the Stye Settings tag.

Buttons:
Hold Data The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.
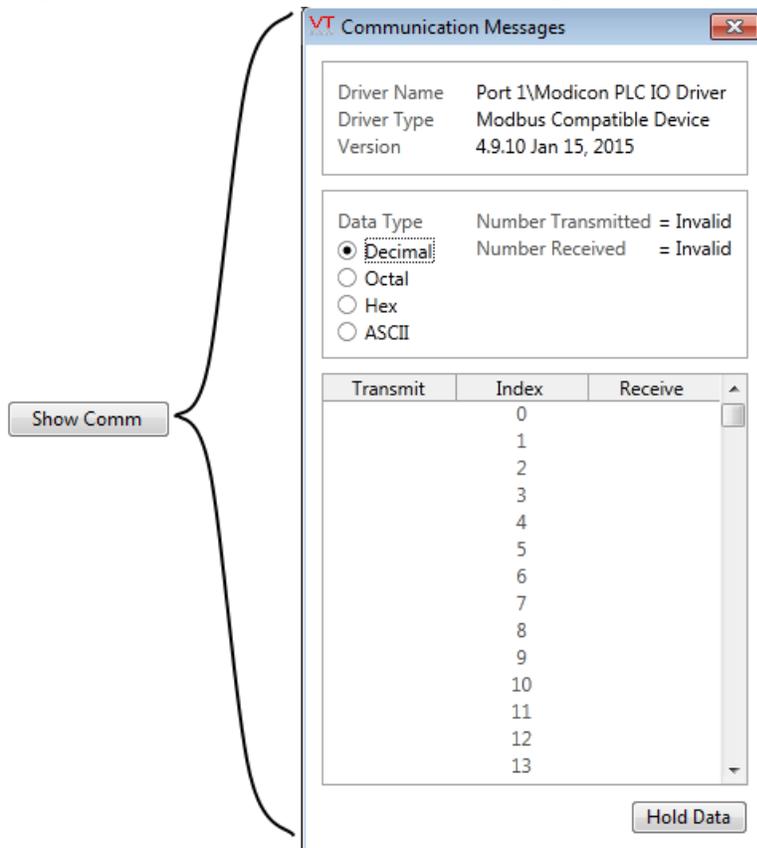
The properties dialog for this widget appears as follows:
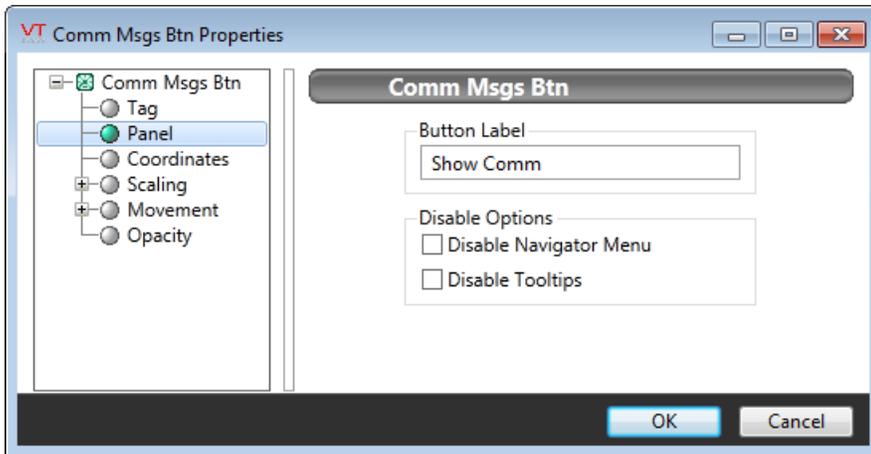
Button Label

Provide the text (if any) that will be displayed on the button.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Connection Status\
Analytics\Diagnostics\

Related Information:

Each driver has its own version of the communication messages dialog, as described in the following topics.

...Allen-Bradley Communication Messages Dialog

...CalAmp Diagnostic Driver Communication Messages Dialog

...Data Flow Communication Messages Dialog

...DNP3 Driver Communication Messages Dialog

...MDS Diagnostic Driver Communication Messages Dialog

...Modbus Compatible Communication Messages Dialog

...Omron Communication Messages Dialog

...Polling Driver Messages Dialog

...Siemens S7 Communication Messages Dialog

Allen-Bradley Communication Messages Dialog

The Comm Messages button will display the following information about an Allen-Bradley driver tag.
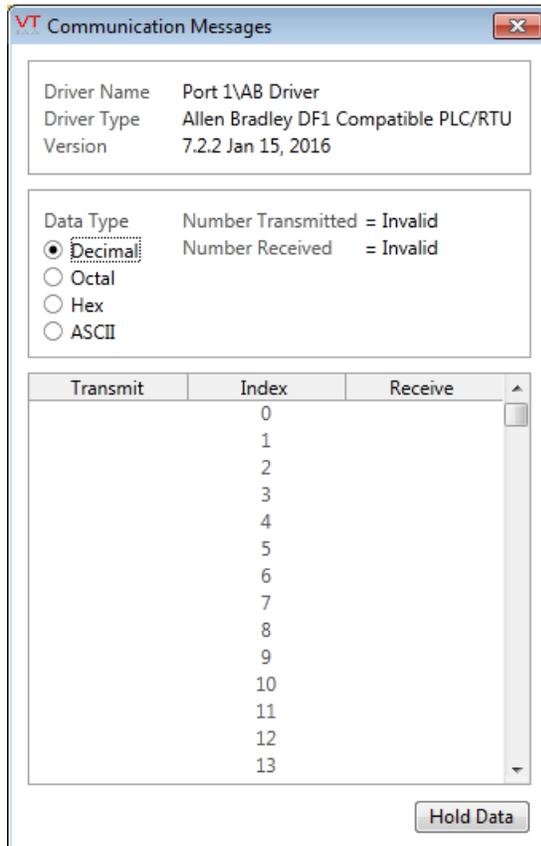
Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If you have the protocol information for the Allen-Bradley device, you can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.
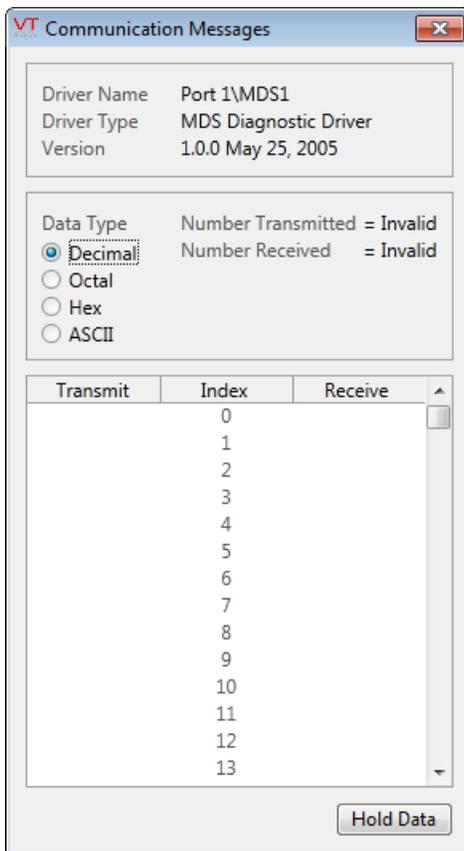
If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the device for data (sending a read request), but the device is not responding. In this case it is wise to check the properties of the Allen-Bradley driver tag (station address, comm. channel, etc.) to make sure all settings are correct.

Related Information:

Allen-Bradley Driver Tags
Allen-Bradley Statistics Dialog

CalAmp Diagnostic Driver Communication Messages Dialog

The Comm Messages button will display the following information about a CalAmp diagnostic driver tag.

Displays the full driver name and version.
Data may be displayed in any of the 4 types shown.
Number Transmitted and Received provides a running count of messages in each direction.
Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.
The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.
If one has the protocol information for the CalAmp hardware, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.
If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties

of the CalAmp diagnostic driver tag (station address, comm. channel, etc.) to make sure all settings are correct.
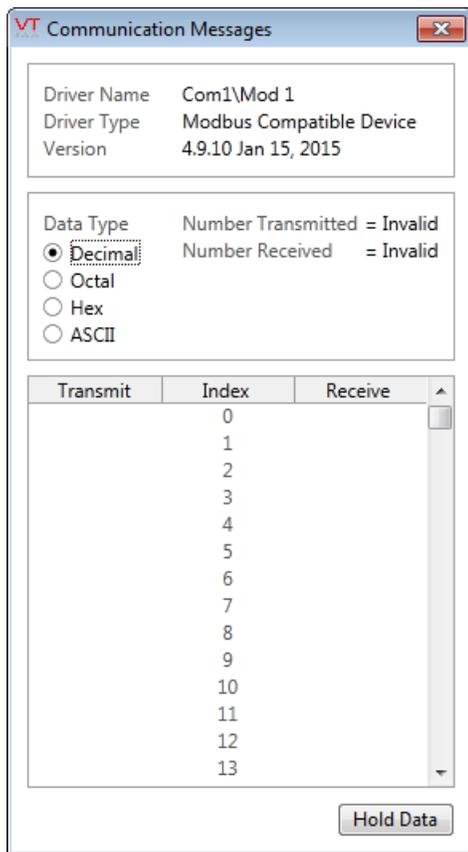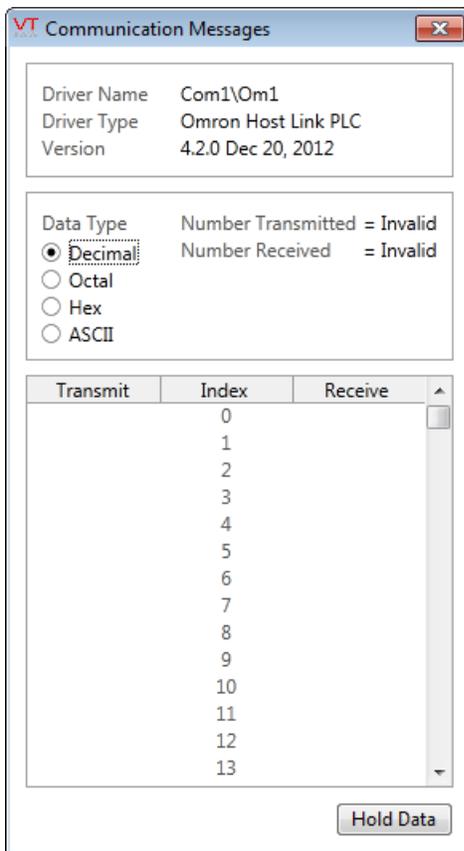
Related Information:
CalAmp Diagnostic Driver Tags

Data Flow Communication Messages Dialog

The Comm Messages button will display the following information about a Data Flow RTU driver tag.

Displays the full driver name and version. Data may be displayed in any of the 4 types shown.
Number Transmitted and Received provides a running count of messages in each direction.
Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.
The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.
If one has the protocol information for the Data Flow RTU, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal so long as bytes appear in both the Transmit and Receive columns.
If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the RTU for data (sending a read request) but the RTU is not responding (as can be seen in the example shown above). In this case it is wise to check the properties of the Data Flow RTU driver tag to make sure all settings are correct.

Related Information:
Data Flow RTU Driver Tags
Data Flow Statistics Dialog

DNP3 Driver Communication Messages Dialog

The Comm Messages button will display the following information about a DNP3 diagnostic driver tag.

Log Data (Unique to the DNP3 driver.) Data will be logged to a text file in the application's Data\TraceFiles folder while the LogData button is pressed.
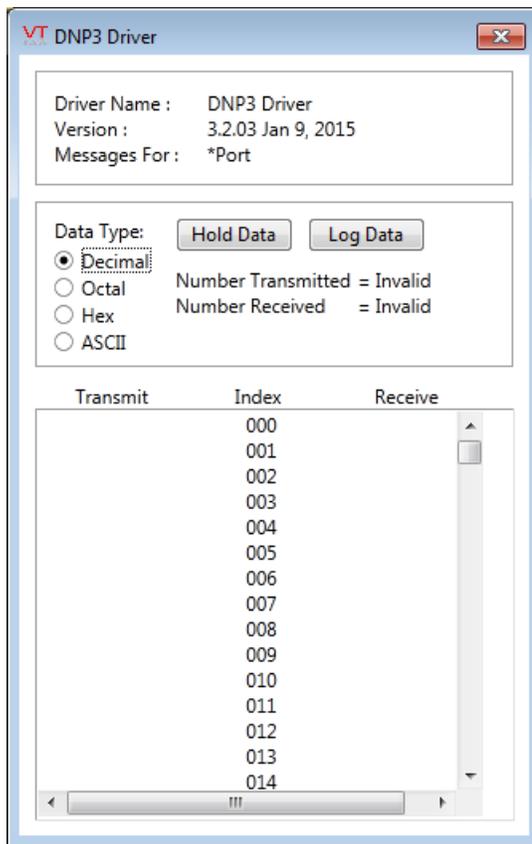
Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

Data will be logged to a text file in the application's Data\TraceFiles folder while the LogData button is pressed.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

If one has the protocol information for the DNP3 hardware, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties of the DNP3 driver tag (station address, comm. channel, etc.) to make sure all settings are correct.
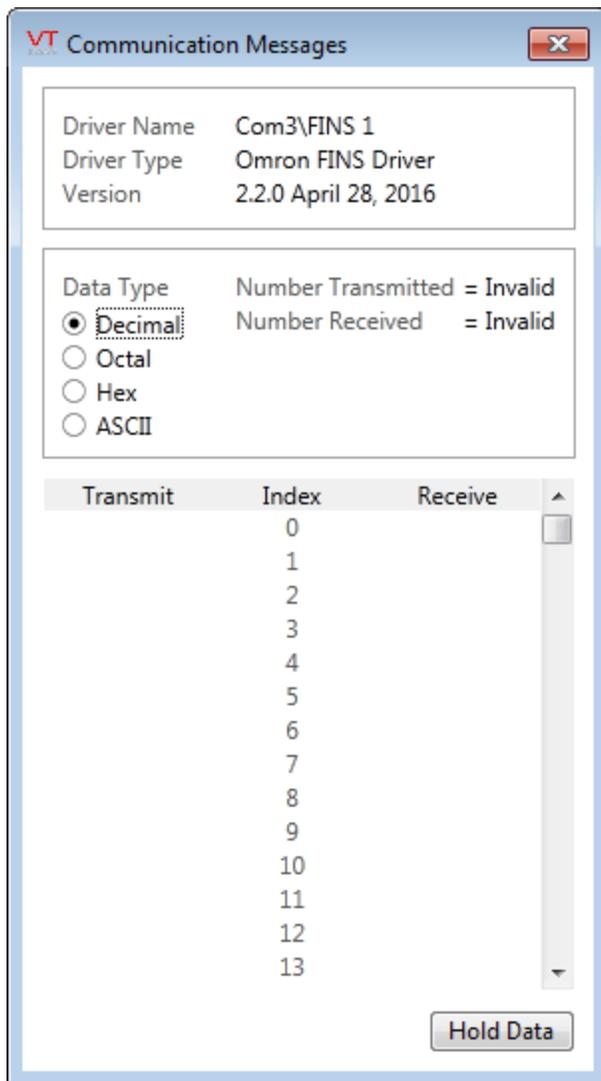
Related Information:

DNP3 Driver Tags

DNP3 Driver Statistics Dialog

Enron Modbus Driver Communication Messages Dialog

The Comm Messages button will display the following information about an Enron Modbus driver tag in response to a left-click. Note that a right-click on the Show Comm button will open a menu, from which you can view associated alarms or a trend window.

Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If you have the protocol information for the device, you can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties of the tag to make sure all settings are correct.
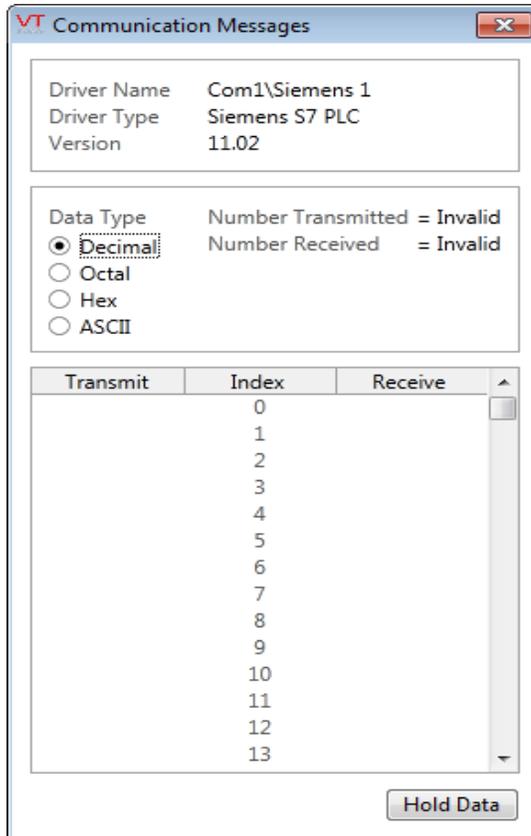
Related Information:

Enron Modbus Driver Tags

Fisher ROC Driver Communication Messages Dialog

The Comm Messages button will display the following information about a Fisher ROC driver tag in response to a left-click. Note that a right-click on the Show Comm button will open a menu, from which you can view associated alarms or a trend window.

Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If you have the protocol information for the device, you can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but

the PLC is not responding. In this case it is wise to check the properties of the tag to make sure all settings are correct.

Related Information:
Fisher ROC Driver Tags

GE Series 90 Driver Communication Messages Dialog

The Comm Messages button will display the following information about a GE Series 90 driver tag in response to a left-click.

Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If you have the protocol information for the device, you can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties of the tag to make sure all settings are correct.

Related Information:

GE Series 90 Driver Tags

MDS Diagnostic Driver Communication Messages Dialog

The Comm Messages button will display the following information about a MDS diagnostic driver tag.

Displays the full driver name and version. Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted. If one has the protocol information for the MDS hardware, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties

of the MDS diagnostic driver tag (station address, comm. channel, etc.) to make sure all settings are correct.

Related Information:
MDS Diagnostic Driver Tags
MDS Diagnostic Driver Statistics Dialog

Modbus Compatible Communication Messages Dialog

The Comm Messages button will display the following information about a Modbus Compatible Device tag.

Displays the full driver name and version. Data may be displayed in any of the 4 types shown.
Number Transmitted and Received provides a running count of messages in each direction.
Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.
The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.
If one has the protocol information for the Modicon PLC, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.
If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties

of the Modbus Compatible Device tag (station address, comm. channel, etc.) to make sure all settings are correct.

Related Information:
Modbus Compatible Device Driver Tags
Modbus Compatible Device Statistics Dialog

Omron Communication Messages Dialog

The Comm Messages button will display the following information about an Omron driver tag.

Displays the full driver name and version. Data may be displayed in any of the 4 types shown.
Number Transmitted and Received provides a running count of messages in each direction.
Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.
The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.
If one has the protocol information for the Omron PLC, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.
If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties

of the Omron driver tag (station address, comm. channel, etc.) to make sure all settings are correct.

Related Information:
Omron Host Link Driver Tags
Omron Statistics Dialog

Omron FINS Driver Communication Messages Dialog

The Comm Messages button will display the following information about an Omron FINS driver tag in response to a left-click.

Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If you have the protocol information for the device, you can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is

that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties of the tag to make sure all settings are correct.

Related Information:

Omron FINS Driver Tags

Polling Driver Messages Dialog

The Communication Messages dialog for a Polling driver will show the communication messages from the driver tag that this Polling driver is subordinate to.

Siemens S7 Communication Messages Dialog

The Comm Messages button will display the following information about a Siemens S7 driver tag.

Displays the full driver name and version.

Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If one has the protocol information for the Siemens S7 PLC, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

If bytes appear in the Transmit column but not in the Receive column, VTScada is trying to poll the PLC for data (sending a read request) but the PLC is not responding. In this case it is wise to check the properties of the Siemens S7 driver tag (station address, comm. channel, etc.) to make sure all settings are correct.

Related Information:

Siemens S7 Driver Tags

Siemens S7 Driver Statistics Dialog

SNMP Communication Messages Dialog

The Comm Messages button will display the following information about a SNMP driver tag.

Displays the full driver name and version. Data may be displayed in any of the 4 types shown.

Number Transmitted and Received provides a running count of messages in each direction.

Transmit & Receive display communications data using the data type selected above. The Index helps locate each byte within a message.

The Hold Data button will freeze the display. Otherwise the dialog updates continuously as messages are transmitted.

If one has the protocol information for the SNMP device, one can interpret each of the transmitted and received bytes to determine their meaning. The basic information that can be obtained from the Communication Messages window is that communications are proceeding as normal as long as bytes appear in both the Transmit and Receive columns.

Related Information:

SNMP Driver Tags

SNMP Driver Statistics Dialog

Show Statistics Button Widget

The Comm Statistics Button tag widget is used to draw a button that can be clicked to display communication statistics for either the device associated with the button, or all devices in your system.

* Does not use the Stye Settings tag.

(Contents will vary by driver)
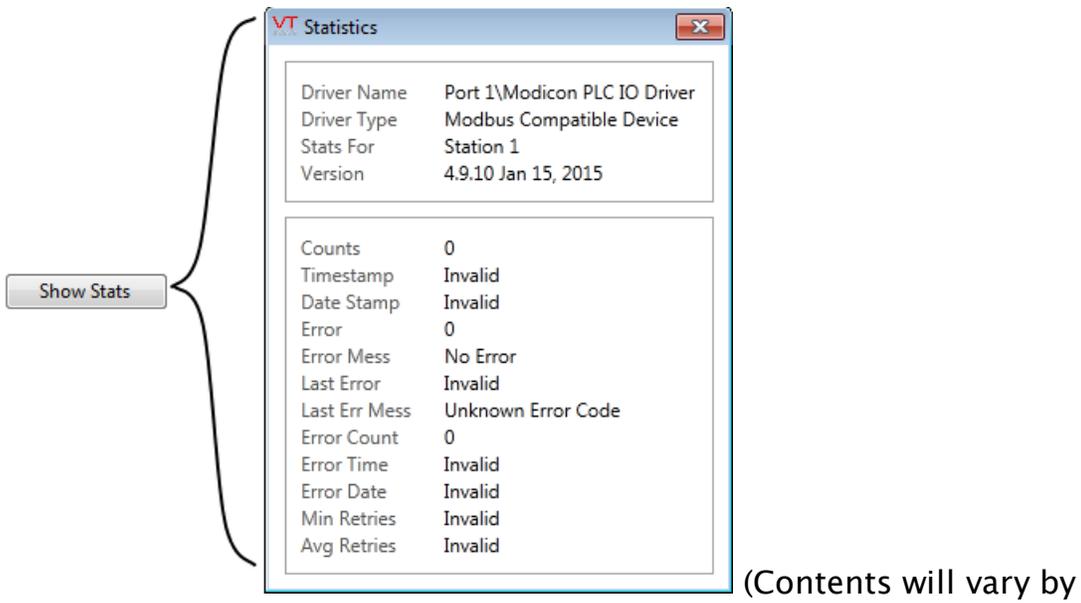
The properties dialog for this widget:
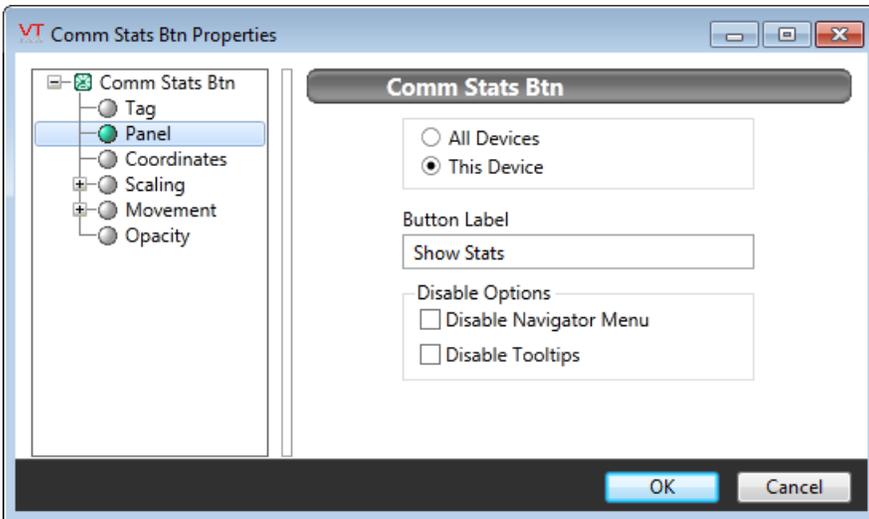
You can choose to display statistics for just the attached driver, or for all drivers.

Button Label

Provide the text (if any) that will be displayed on the button.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Connection Status\

Analytics\Statistics\

Analytics\Diagnostics\

Each driver has its own version of the communication statistics dialog, as described in the following topics.

Related Information:

...Allen-Bradley Statistics Dialog

...CalAmp Diagnostic Driver Statistics Dialog

...CIP Driver Statistics Dialog

...Data Flow Statistics Dialog

...DNP3 Driver Statistics Dialog

...DriverMUX Statistics Dialog

...Enron Modbus Device Statistics Dialog

...Fisher ROC Driver Statistics Dialog

...GE Series 90 Driver Statistics Dialog

...MDS Diagnostic Driver Statistics Dialog

...Modbus Compatible Device Statistics Dialog

...Motorola ACE Statistics Dialog

...Omron Statistics Dialog

...Omron FINS Driver Statistics Dialog

...OPC Client Driver Statistics Dialog

...Siemens S7 Driver Statistics Dialog

...SNMP Driver Statistics Dialog

Allen-Bradley Statistics Dialog

The Show Stats button will display the following information about an DriverMUX tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Stats For The name of the station for which the statistics are displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Error Owner The address at which the last error occurred.

Error Station The address of the station at which the last error occurred.

Error Mem Addr The PLC address at which the last error occurred.

Serial Error The error number (code) for the serial error (if one has occurred).

Serial ID The ID of the serial error.

SD ID The ID of the KT/SD board, if one has been installed for this I/O device.

CalAmp Diagnostic Driver Statistics Dialog

The Show Stats button will display the following information about a CalAmp diagnostic driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Error Info Available description about the error.

Related Information:

CalAmp Diagnostic Driver Tags

CalAmp Diagnostic Driver Communication Messages Dialog

CIP Driver Statistics Dialog

The Show Stats button will display the following information about a CIP driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess The message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time Time when the last error message was recorded.

Error Date Date when the last error message was recorded.

Error Info Further details regarding the message.

Related Information:

CIP Driver Tags

Data Flow Statistics Dialog

The Show Stats button will display the following information about a Data Flow RTU driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Stats For The name of the station for which the statistics are displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that is occurring.

Last Error The error number (code) for the last error that last occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Related Information:

Data Flow RTU Driver Tags

Data Flow Communication Messages Dialog

DNP3 Driver Statistics Dialog

The Show Stats button will display the following information about a DNP3 driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Stats For The name of the station for which the statistics are displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

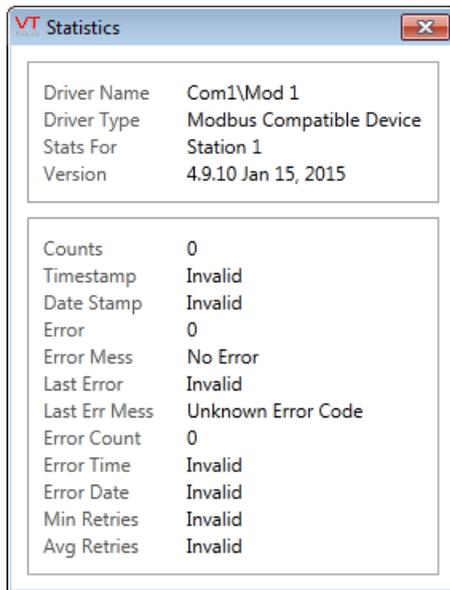**Error Date** The date of the last reported error.

Related Information:

DNP3 Driver Tags

DNP3 Driver Communication Messages Dialog

DriverMUX Statistics Dialog

The contents of this dialog will depend on the mode in which the Driver-MUX is operating. If set to Alternating or Parallel, the contents will show statistics for the Driver Multiplexer, as follows.

In all other modes, the statistics will be for the selected subordinate driver.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the current error message for this driver.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Last Error Mode Identifies the mode in which the driver was operating at the time of the last error.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Related Information:

Driver Multiplexer Tags

Enron Modbus Device Statistics Dialog

The Show Stats button will display the following information about an Enron Modbus driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Min Retries The minimum number of attempts at communication (retries) once an error has occurred.

Avg Retries The average number of attempts at communication (retries) once an error has occurred.

Related Information:

Enron Modbus Driver Tags

Fisher ROC Driver Statistics Dialog

The Show Stats button will display the following information about a Fisher ROC driver tag.

VT Statistics	
Driver Name	Port 1\ROC 1
Driver Type	Fisher ROC Driver
Version	2.0 Mar 28, 2016
Counts	0
Timestamp	Invalid
Datestamp	Invalid
Error	42
Error Mess	Communication channel not available
Last Error	42
Last Err Mess	Communication channel not available
Error Count	57
Error Address	3:3:2
Error Time	08:43:15
Error Date	April 6, 2016
Min Retries	Invalid
Max Retries	Invalid
Avg Retries	Invalid

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Address The address at which the error occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

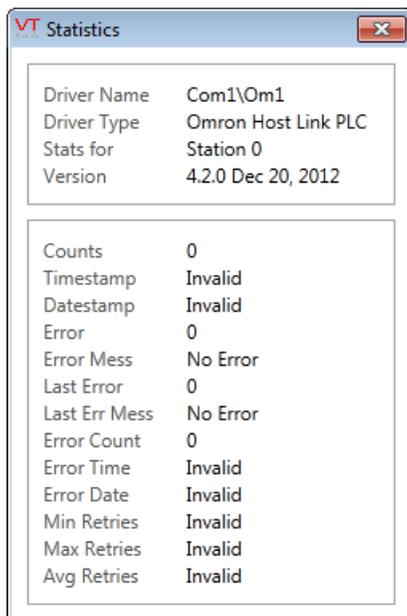**Error Owner** The address at which the last error occurred.

Error Station The address of the station at which the last error occurred.

Related Information:

Fisher ROC Driver Tags

GE Series 90 Driver Statistics Dialog

The Show Stats button will display the following information about a GE Series 90 driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

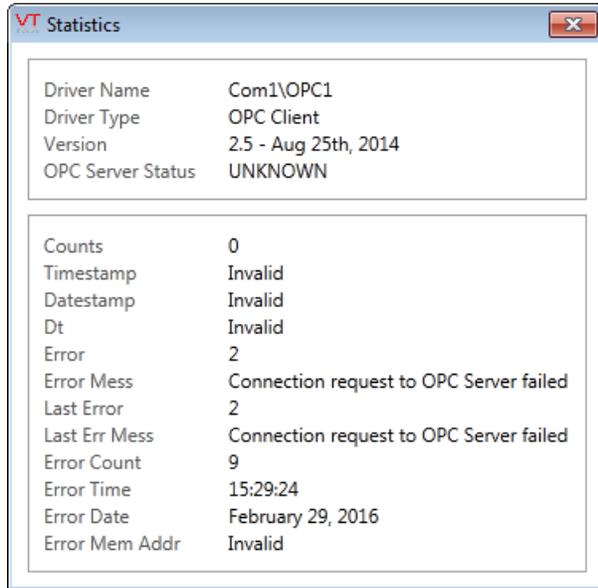**Error Date** The date of the last reported error.

Error Owner The address at which the last error occurred.

Error Station The address of the station at which the last error occurred.

Error Mem Addr

Related Information:

GE Series 90 Driver Tags

MDS Diagnostic Driver Statistics Dialog

The Show Stats button will display the following information about a MDS diagnostic driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Related Information:

MDS Diagnostic Driver Tags

MDS Diagnostic Driver Communication Messages Dialog

Modbus Compatible Device Statistics Dialog

The Show Stats button will display the following information about a Modbus Compatible Device tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Stats For The name of the station for which the statistics are displayed.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Min Retries The minimum number of attempts at communication (retries) once an error has occurred.

Avg Retries The average number of attempts at communication (retries) once an error has occurred.

Related Information:

Modbus Compatible Device Driver Tags

Modbus Compatible Communication Messages Dialog

Motorola ACE Statistics Dialog

The Show Stats button will display the following information about a Motorola ACE Driver tag.
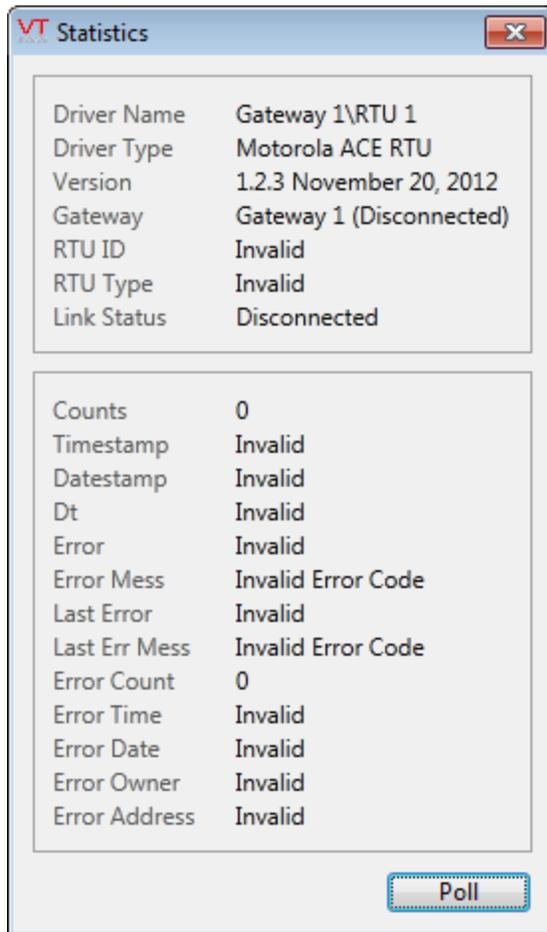
Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

Gateway The name of the IP Gateway tag used by this driver.

RTU ID The name of the RTU as specified in the driver configuration.

RTU Type The name of type tables as specified in the configuration file.

Link Status Shows whether or not the link is connected.

Counts The number of successful communications since the system started.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Error Owner The name of the module, in which the error occurred.

Error Address May be used to identify which instance of the module, named in Error Owner, caused the error

Related Information:

Motorola ACE Driver Tags

Omron Statistics Dialog

 The Show Stats button will display the following information about an Omron driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Stats For The name of the station for which the statistics are displayed.

Version The version number and date that this driver was created.

Counts The number of successful com-munications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

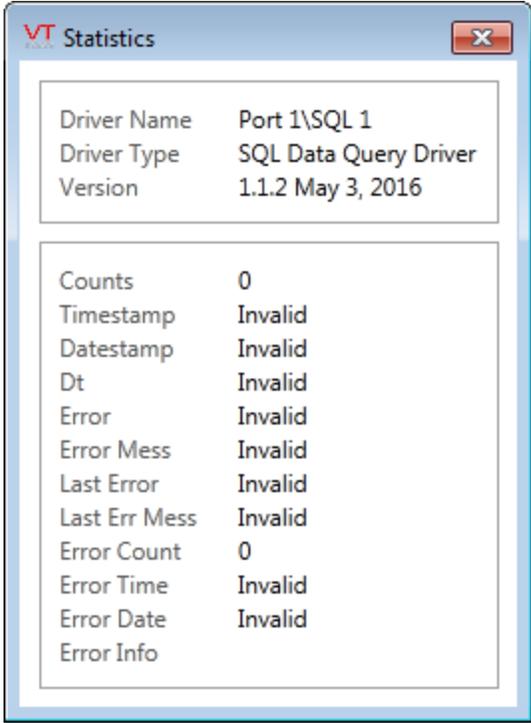**Last Error Mess** A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Min Retries The minimum number of attempts at communication (retries) once an error has occurred.

Max Retries The maximum number of attempts at communication (retries) once an error has occurred.

Avg Retries The average number of attempts at communication (retries) once an error has occurred.

Related Information:

Omron Host Link Driver Tags
Omron Communication Messages Dialog

Omron FINS Driver Statistics Dialog

The Show Stats button will display the following information about an Omron FINS driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Error Owner The address at which the last error occurred.

Related Information:

Omron FINS Driver Tags

OPC Client Driver Statistics Dialog

The Show Stats button will display the following information about an OPC Client driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

OPC Server Status Information about the attached OPC server.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Error Mem Addr The memory address on the OPC server, related to the error.

Note: For OPC, the Response time value will vary, since the OPC Client driver does not make requests for data; rather, data is sent to it when changes occur.

Related Information:
OPC Client Driver Tags

Siemens S7 Driver Statistics Dialog

The Show Stats button will display the following information about a Siemens S7 driver tag.
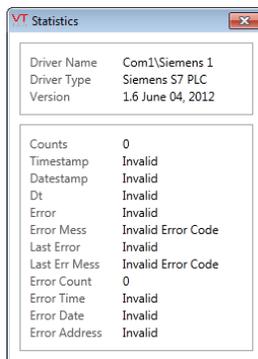
Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Version The version number and date that this driver was created.

Stats For The name of the station for which the statistics are displayed.

Counts The number of successful communications since the system started.

Success Count Displays a count of successfully transmitted messages.

Timestamp Displays the time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Response Time The time it takes the PLC/RTU to receive a command, process it and send a response.

Error The error number (code) for the current error.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Time The time of the last reported error.

Error Date The date of the last

Error Mess Describes the type of com- munications error that has occurred.
Failed Count Incremented on each error.

reported error.
Error Address If an address was associated with the error, it will be recorded here.

SNMP Driver Statistics Dialog

The Show Stats button will display the following information about a SNMP driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.
Version The version number and date that this driver was created.
Stats For The name of the station for which the statistics are displayed.
Counts The number of successful communications since the system started.
Timestamp Displays the time of the last successful com- munication.

Date Stamp The date of the last successful communication.
Error The error number (code) for the current error.
Error Mess Describes the type of communications error that has occurred.
Err OID Gives the object ID where the error occurred.
Error Count Incremented on each error.

Last Error The number (code) for the error that last occurred.
Last Error Mess A message describing the type of communications error that last occurred.
Last Err OID A message giving the object ID where the error last occurred.
Error Time The time of the last reported error.
Error Date The date of the last reported error.

SQL Data Driver Statistics Dialog

The Show Stats button will display the following information about an SQL Data driver tag.

Driver Name The name of the driver (within the driver's code) for which statistics are being displayed.

Driver Type Identifies the driver.

Version The version number and date that this driver was created.

Counts The number of successful communications since the system started.

Timestamp The time of the last successful communication.

Date Stamp The date of the last successful communication.

Dt The Response time – that is, the time between a request for information and the message being sent.

Error The error number (code) for the current error.

Error Mess A message describing the type of communications error that has occurred.

Last Error The error number (code) for the last error that occurred.

Last Error Mess A message describing the type of communications error that last occurred.

Error Count The number of errors that have occurred since the system started.

Error Time The time of the last reported error.

Error Date The date of the last reported error.

Related Information:
SQL Data Query Driver Tag

Compass 1 Widget

Used by: Analog Input, Analog Output, Analog Status, Analog Control, Calculation.
* Does not use the Stye Settings tag.
Displays a compass having a needle that rotates according to the tag's value. The compass needle will point to North when the tag's value is 0. Direction of rotation is clockwise for positive values. The full range of the underlying tag's scaled values will cause one rotation.

The properties dialog for the Compass 1 widget:
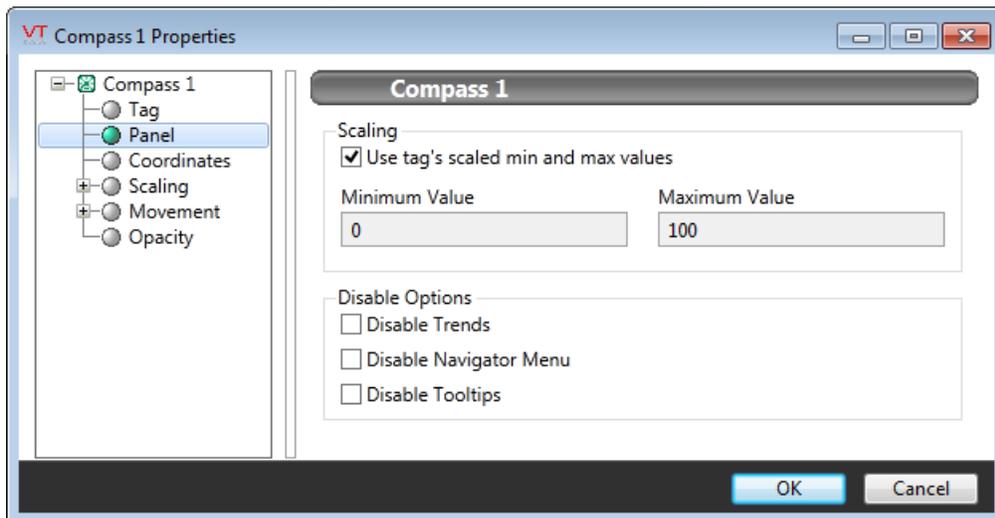
Use Tag's Scaled Min & Max Values

> In most cases, it is best to select this option. The tag's min-
> imum scaled value corresponds to North (0 degrees on the
> compass) and the maximum scaled output value as well (360
> degrees).

If the compass is to represent tags such as Calculation and Function that do not have their own scaled values, then deselect this box and enter the minimum and maximum values that can be expected from the tag.

Minimum Value and Maximum Value

The Minimum Value and Maximum Value fields are used to scale the amount of rotation of the needle, relative to the tag's scaled output values.
For example, given a tag with scaled output ranging from 0 to 100:
Setting the widget's scale range from 0 to 200 results in a needle that will rotate by a maximum of 180 degrees.
Setting the widget's scale range from 0 to 50 results in a needle that will complete two full rotations.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Compass\

Related Information:

...Choices for Operator Interaction

Compass 2 Widget

Used by: Analog Input, Analog Output, Analog Status, Analog Control, Calculation.
* Does not use the Stye Settings tag.
Displays a compass having a numeric display that rotates according to the tag's value. The 0-value will point to North when the tag's value is 0. Direction of rotation is counter clockwise for positive values. The full range of the underlying tag's scaled values will cause one rotation.

The properties dialog for the Compass 2 widget:
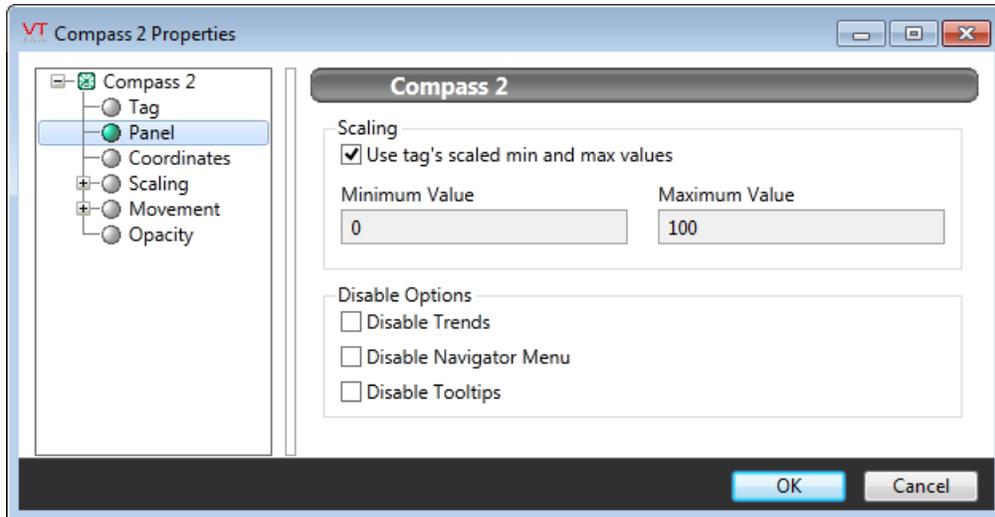
Use Tag's Scaled Min & Max Values

In most cases, it is best to select this option. The tag's minimum scaled value corresponds to North (0 degrees on the compass) and the maximum scaled output value as well (360 degrees).
If the compass is to represent tags such as Calculation and Function that do not have their own scaled values, then deselect this box and enter the minimum and maximum values that can be expected from the tag.

Minimum Value and Maximum Value

The Minimum Value and Maximum Value fields are used to scale the amount of rotation of the dial, relative to the tag's scaled output values.

For example, given a tag with scaled output ranging from 0 to 100:

Setting the widget's scale range from 0 to 200 results in a dial that will rotate by a maximum of 180 degrees.

Setting the widget's scale range from 0 to 50 results in a dial that will complete two full rotations.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Compass\

Related Information:

...Choices for Operator Interaction

Connection Status Indicator Widget

Used by: Serial Port
Enabled for TCP/IP Port and UDP/IP Port, but should not be used for those tags.
* Does not use the Stye Settings tag.
The Connection Status Indicator is used to notify operators when the port being monitored is in any of the following states: Disconnected, Connecting, Connected, or Failed.
You may configure the indicator to use any color for each state, and to use any image (or none) to display the status. The default configuration is a color-coded rectangle. Note that the rectangle is solid – it will not interact with active orange as is the case with an Image Change widget.
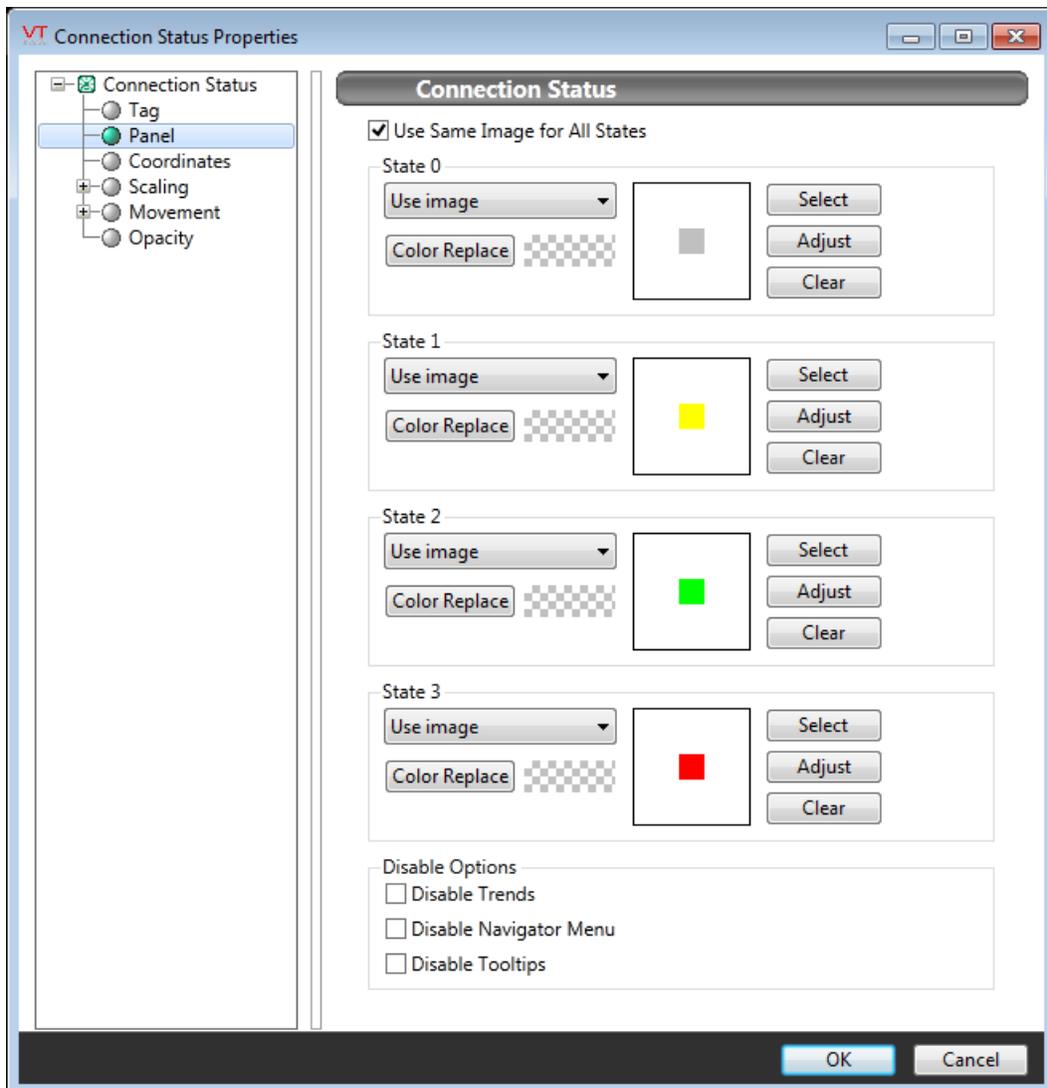
Use Same Image for All States

Check this box if you wish to use the image selected for State 0 (Disconnected), for all states of the tag.
This option is often used when the selected image's indicator area (the small rectangle at its base) is to be replaced with a new color according to the state of the tag.

Use Image / No Image

Select whether or not to include an image for the Connector

Status.

If not using an image, you should select Color Replace, then position the Connector Status over an area of Active Orange. Everything that is Active Orange below the Connector Status will be replaced with the matching state's color.

Color Replace

Use the Color Replace option if the image includes a color indicator patch, or if you have selected the No Image option. (Color indicators within images are areas of Active Orange.)

Select (Image)

Opens the Select Color Dialog, from which you can choose a color, either to colorize the image, or to replace Active Orange color patches.

Adjust (Image)

Opens the Adjust Image dialog, with its full range of controls over image appearance.

Clear (Image)

Remove the image. Has the same effect as selecting the No Image option.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Color Indications\
Basic Components\Images\
Analytics\Communications\Connection Status

Related Information:

...Adjust Image Dialog – Tools within the Adjust Image dialog.

...Choices for Operator Interaction

Data Age Widget

Used by: Data Flow RTU, Polling Driver.
* Does not use the Stye Settings tag.
The Data Age tag widget is used to display the time of the most recent poll of an RTU in hours, minutes, and seconds.

00:00:25

An example of a configured properties dialog for the Data Age widget:

Font

    Select one of the font tags (or parameter linked to a font tag) that you with to use to display the text. The display will use the default system font if you do not select a font tag.
    The parameters button is enabled only if the tag is drawn on a page that includes a font parameter or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Type of Border

Select the type of border you would like to display the data age within. This can be one of three choices, as illustrated:

Text

Opens the Select Color Dialog from which you can choose the color of the text.

Background

Opens the Select Color Dialog from which you can choose the color of the background.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Polling\

Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Data Flow Station Draw Widget

Used by: Data Flow RTU.
* Does not use the Stye Settings tag.
The Data Flow Station Draw tag widget is used to create a button that an
operator may click in order to view information about the modules that
have been installed at the physical RTU being represented by the asso-
ciated tag.
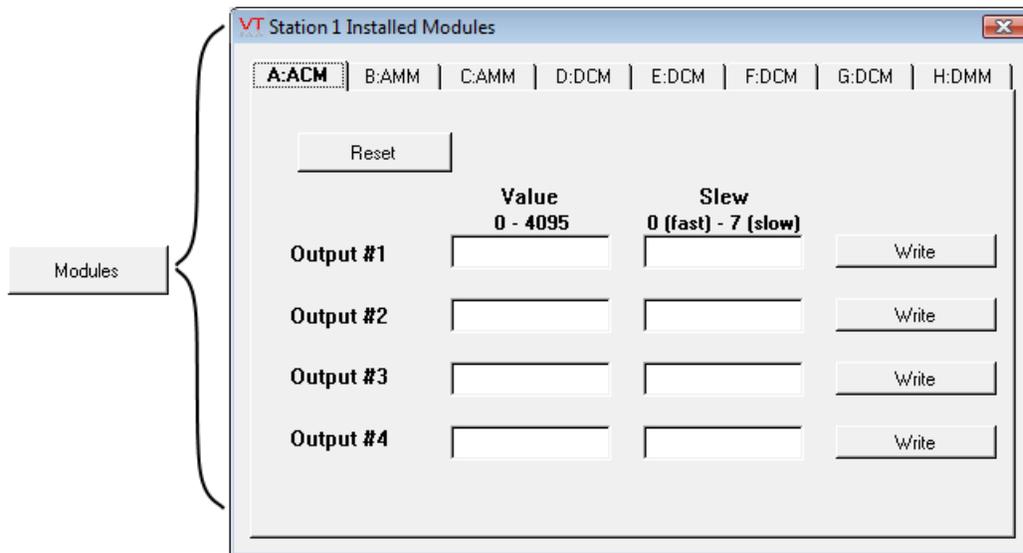
Each lettered module (A to H shown here) represents represents one of
the physical I/O cards installed in the DFS RTU. The three-letter abbre-
viation following the colon represents the type of I/O card and may be
one of the following:

- ACM – Analog Control Module : four analog outputs
- AMM – Analog Monitor Module : four analog inputs
- DCM – Digital Control Module : (4 digital outputs + 8 digital inputs) or (8
  digital outputs + 4 digital inputs) depending on the model
- DMM – Digital Monitor Module : 12 digital inputs

For the Analog Control Module, the value in this dialog represents the
raw (unscaled) value to write to the I/O card that controls the analog out-
put signal on the card. The Slew Rate controls how fast the value changes
from its current value. A slow slew rate would mean that the value
changes gradually from the current value to the new value.

The properties dialog for this button provides only the Disable Options.
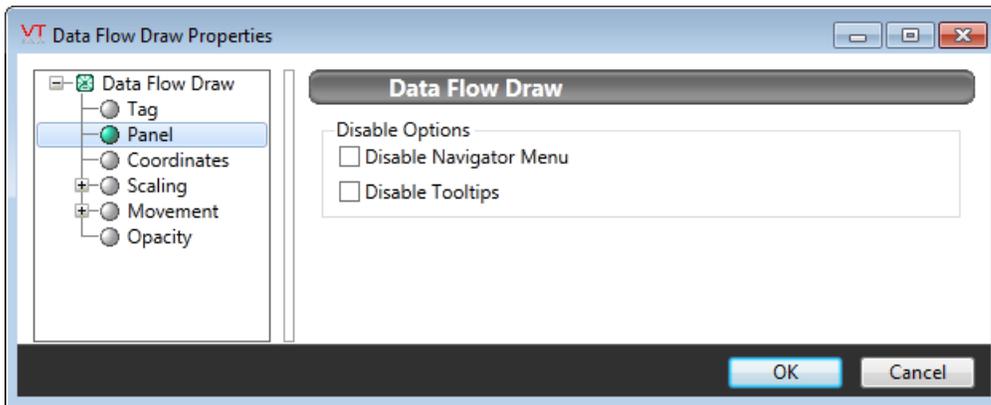
Widget Folders:

Equipment\Lift Stations\Data Flow Systems\

Related Information:

...Choices for Operator Interaction

DFS Station Draw

Used by: DFS Station RTU tag

The DFS Station Draw widget is displayed as a button that operators can use to open a pop-up window showing the installed modules for this station.
The contents of this window will vary according to the station's installed modules.
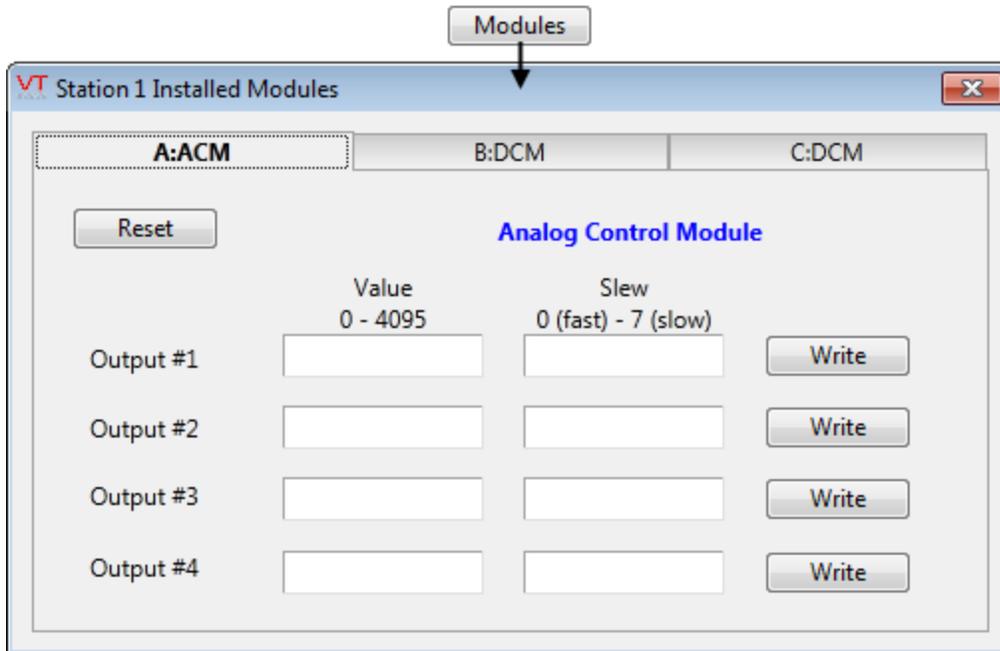
The properties dialog for the x widget. As shown, the only configurable options are to disable tooltips and the navigator menu.
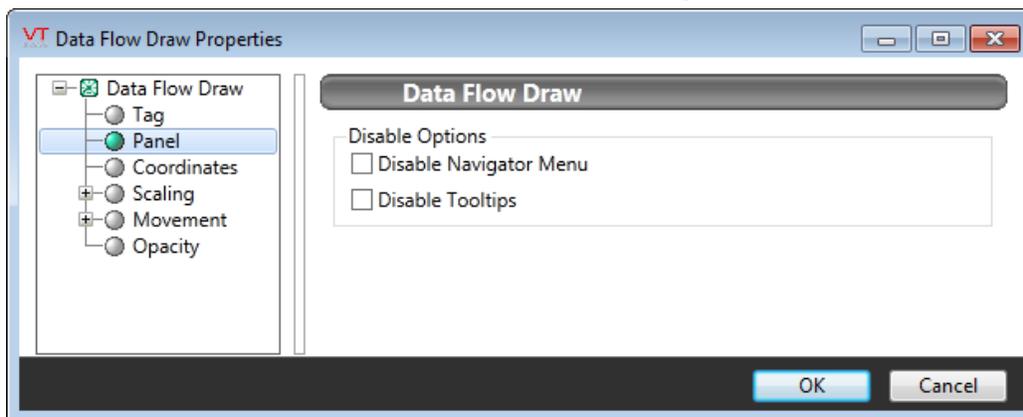
Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

Choices for Operator Interaction

Draw Widget

Used by: Data Flow RTU, Analog Status, Digital Status, Pump Status, Pulse Input.

* Does not use the Stye Settings tag.

The Draw tag widget is used to display a bordered field containing the description of the associated tag on the left, and its value to the right. The description and the border are optional.

If the associated tag type includes an alarm setup then orange dots will appear in the object to indicate that one or both alarms are either not configured or not enabled. For tag types that have both high and low alarms as part of their definition, the upper and lower squares refer to the high and low alarms in that order.

Analog Status 1	0 %
Digital Status 1	Off
Pulse Input 1	0 %
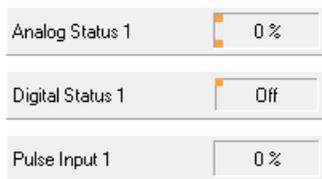
The properties dialog for the Draw widget is as shown:
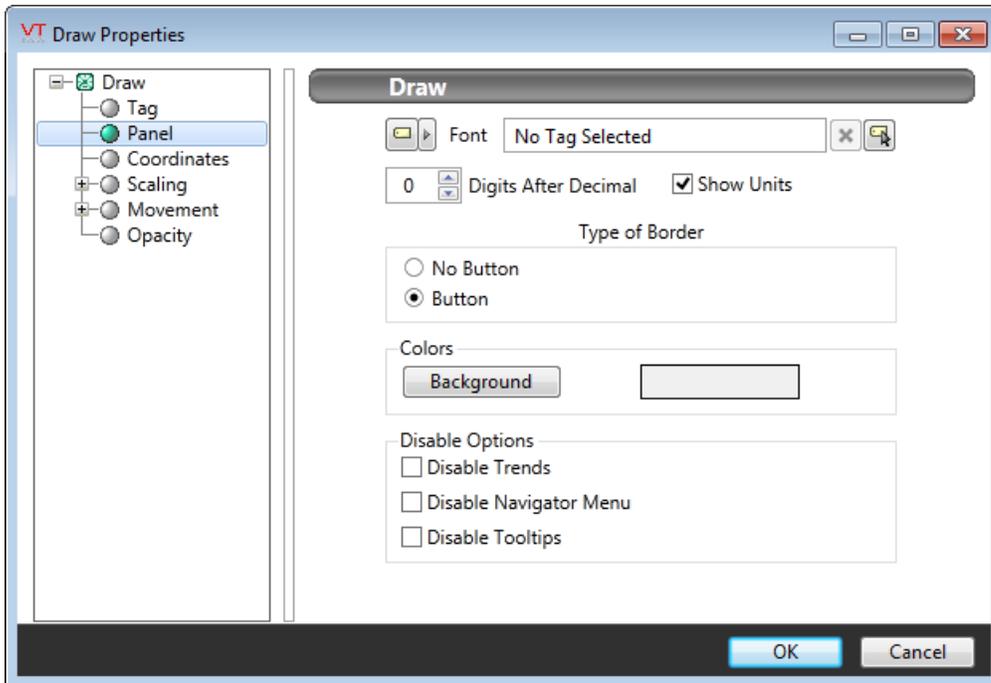
Font

Select one of the font tags (or parameter linked to a font tag) that you with to use to display the text. The display will use the default system font if you do not select a font tag.
The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

Digits After Decimal

Controls how many decimal places will be shown in the display. Values will be rounded to the number set here.

Show Units

Controls whether or not the tag's engineering units (if set) will be shown as part of the display.

Type of Border

Select whether to display the tag name as well as its value (the Button option) or simply the tag's value (the No Button option).

Colors (Background)

Opens the Select Color Dialog from which you can choose a color to be displayed as the background to the tag's value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tag Types\Inputs & Outputs\Digital Status
Tag Types\Inputs & Outputs\Analog Status
Tag Types\Inputs & Outputs\Pump Status

Related Information:

Draw HDV Widget

* Does not use the Stye Settings tag.

Add a scalable view of the Historical Data Viewer to any page as a widget. This widget is designed to be linked to a parent tag, which may be either a Context, or a user-defined type that was created from a Context tag. Alternatively, you may select an existing pen group for it to display.

If linking to a parent tag (Context or user-created type), use the widget's properties dialog to select the child tags that should be included in the plot. You can also configure the appearance of the HDV, including title bar, time selector (footer bar), colors and more. A useful technique is to add the DrawHDV widget to a parameterized page, linking the widget to a page parameter of type, tag. Menu entries or page hotboxes are then configured to use different instances of the user-created tag type when opening the page.

Components of the HDV including the toolbar, legend, and scales may be configured to be hidden or displayed as required.

> Note: If running a legacy application (created with 11.0 or earlier), user-created tag types will not link to a Draw HDV widget without additional code. Refer to Moving to the Current Version in the VTScada Admin Guide.

Procedure for use:

1. Add the Draw HDV Widget to a page.
2. Link the widget to a context tag or user-defined type, containing the tags to be plotted.
3. Open the Properties dialog of the Draw HDV Widget.
4. Select the child tags that will be included in the plot.
5. Adjust other display options as required.

> Note: The legend grid and the divider between the legend and the graph can be adjusted only in the live view, not in the Idea Studio.

		Pen	Name	Description	Value	Maximum	Minimum	Average
👁 ⚙	⌒		...\Tank Level	Monitor tank level	48.6 %	17.4	76.7	44.3
👁 ⚙	⌄		...\Tank Level	Monitor tank level	10.1 %	9.7	91.1	47.7
👁 ⚙	⌒		...\Tank Level	Monitor tank level	24 %	17.5	82.1	47.2

Plot / Grid / Notes
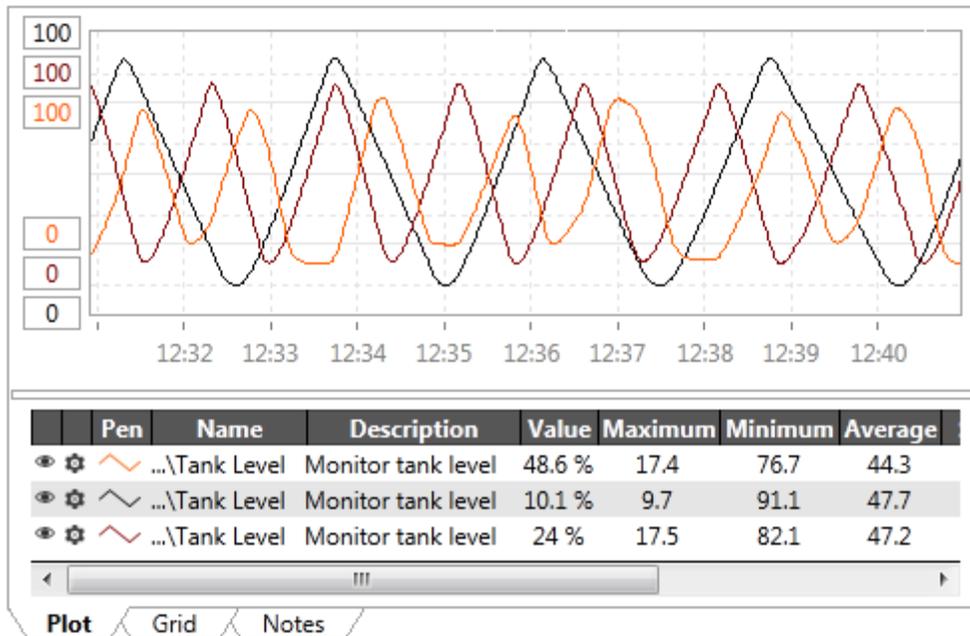
- You can left-click on the graph to add value labels.
- You can open a properties menu for each pen.
- The full controls of the HDV window are not shown by default, but may be added using options in the widget's properties dialog.

The properties dialog for the Draw HDV will vary, depending on whether it is linked to a type or using a named pen group.

Draw HDV Properties

Tag

Tag: Stn1 TCP\Modbus1\Pump 1

Draw HDV

- Draw HDV
  - Tag
  - Panel
  - Coordinates
  - Scaling
  - Movement
  - Opacity

Child Tags (2)

PumpStart
PumpStop

Selected Tags (1)

PumpStatus

Time Span: 10 Minutes

☑ Show Toolbar

Background

Tab Text Color

Views	Enabled	Default
Plot	☑	⦿
Grid	☑	○
Notes	☑	○

Plot | Grid | Notes

☑ Show Legend
☑ Show Tag Scales
☑ Show Time Scales
☑ Show Time Selection
☑ Show Time Cursor
☐ Stacked Plot
☑ Simplified Legend

Plot Background

Grid Color

Text Color

OK   Cancel

Linked to a Type

Using a Pen Group

Tag (If linked to a tag)

The selected context or type, if not using a pen group.

Child Tags (If linked to a tag)

If using a linked tag rather than a named group. The list of child tags of the selected parent that could be plotted, but have not yet been selected.

Selected Tags (If linked to a tag)

If using a linked tag rather than a named group. The list of tags that have been selected to be shown in the HDV widget.

Group (If not linked to a tag)

Select an existing pen group to be displayed.

Time Span

Select the length of time to be displayed in the HDV. Options match those in the full HDV page, ranging from one second to five years.

Show Toolbar

When checked, the full set of HDV tools will be displayed across the top of the widget. The toolbar does not scale, there-fore if the widget is not wide enough to show all the buttons, those on the right will be lost.

Show Time Selection

When checked, the HDV's time scroll bar will be added between the graph and the legend.
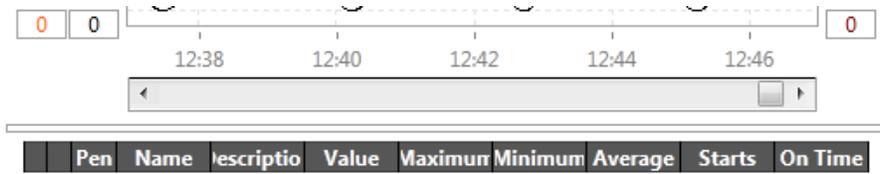
Views: Plot View, Grid View & Notes

Control whether the widget can display the Plot View, the Grid View, and the Notes view. If more than one is selected, choose which view should be shown by default when the page is opened.

Background Color and Tab Text Color

Sets the background and tab-text colors of the overall widget. (Plot, grid and other text colors are set in the following section.)
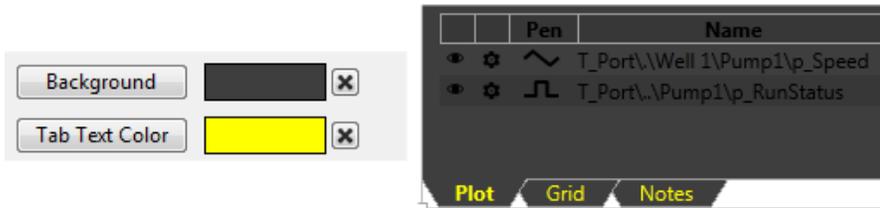
Plot View Options

Options as shown to control what will be included in the plot view of the HDV display, including:

- Choice of color for the background, grid lines and numeric scale. The "X" beside the color choice will reset each value to the default color.
- Options to display the legend, tag scales, time scales, stacked plots.
- Option to display a full legend including statistics or simplified legend of tag identification and values only.
- Option to show the time cursor (vertical line following the pointer).

Grid View Options

Options to control what what will be visible, and what the default settings will be when an operator views the grid. If the toolbar option is not provided, then the default settings will be the only view possible in the grid.

- Option to sort the grid in ascending order by time stamp (sorts in descending order by default).
- Select between data retrieval modes, raw values, averages, minimums or maximums.

- For all data retrieval modes other than raw values, set the time per record (row) for each average, minimum or maximum.
- Choose colors for the cell background and the grid lines.

Notes View Options

Choose from the following options of what will be included in the notes tab and how it will be shown.

- Sort by either the notes's original time stamp or date of most recent comment (update) added to the note.
- Option to display either the date and time when the notes were created, or the age of the note.
- Choose colors for the text and background.

Widget Folders:

Analytics\Trends & Historical Data
Tools\Standard Library
Tag Types\Organization/Containers\Context

Related Information:

...Plot Trends and Graphs – Instructions are provided in the Operator's Guide.

...Operator Notes – General information is provided in the Operator's Guide.

...HDV Button Widget – Draw a button that will open a customized HDV window.

...HDV Hotbox Widget – Draw a hotbox that will open a customized HDV window.

...Plot Data Widget – Draw an HDV widget that is linked to a single tag.

...See: "File Management Tools" in the VTScada Admin Guide

Draw Text Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband

Control, History Statistics, Linear Legend, Rate of Change, Selector
Switch, Totalizer, SQL Logger.
* Does not use the Stye Settings tag.
The Draw Text widget is used to display the current value of any of a tag
as text. Units (if any) will not be displayed and numeric values will display
either 0 or 8 decimal points with no option for user-configuration. If you
require just one decimal points of accuracy, or if you would like to dis-
play the engineering units that the value represents, then the Numeric
Value Widget will be a better choice.
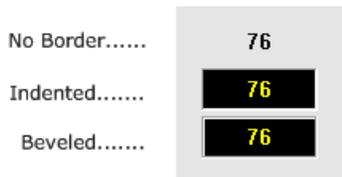
The properties dialog for the Draw Text object:

Font

> Select one of the font tags (or parameter linked to a font tag)
> that you wish to use to display the text. The display will use
> the default system font if you do not select a font tag.

The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Horizontal Alignment

Sets the position of the text within the display area. You may select left-aligned, right-aligned or centered. To align the display area itself with other elements on the page, see: Alignment and Positioning Tools.

Type of Border

Examples of the three border types are shown at the beginning of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the color of the text.
You can also set the background color against which the text will be displayed.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Text\

Related Information:

...Choices for Operator Interaction

DriverSelect Widget

Used by: DriverMux (driver multi-plexer) tag.
* Does not use the Stye Settings tag.
The Driver Select widget is used to provide operators with a means to select one of the configured drivers to use exclusively, or to allow the DriverMUX tag to work automatically.
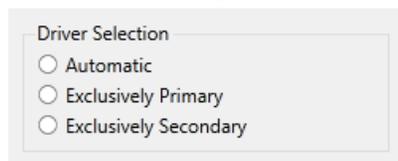
The properties dialog to configure the Driver Select widget is as follows:
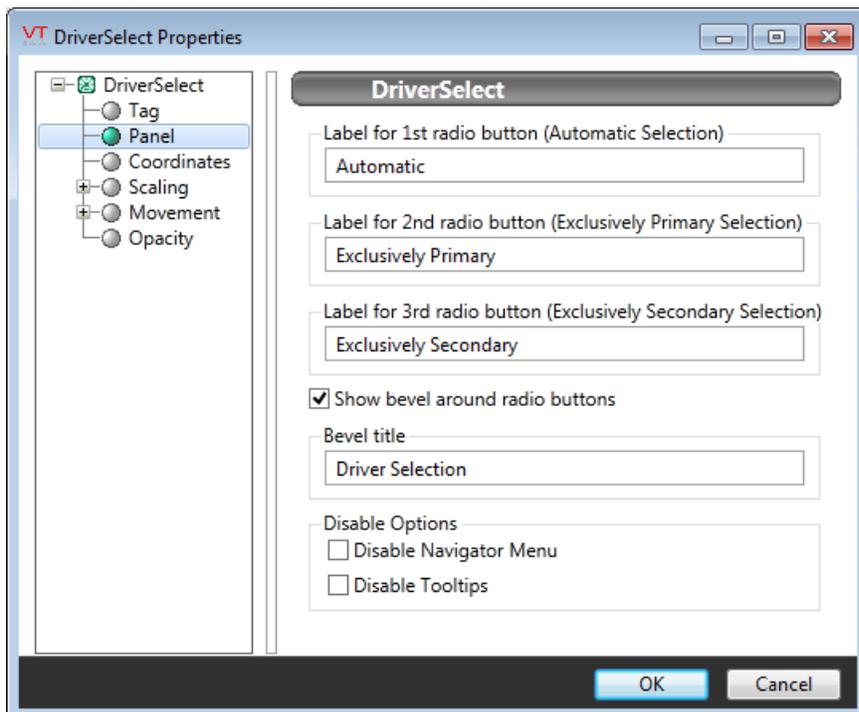
Label for 1st, 2nd and 3rd radio buttons

The function of each radio button is defined within the widget. The default labels describe these functions well, but you may choose to use your own labels for each.

Show bevel around radio buttons

Adds a decorative edge to the cluster of three buttons. When the bevel is shown, you may also add a title.

Bevel title

Available only when the bevel is shown around the radio button cluster. If you have more than one DriverMux tag in your application, it would be appropriate to make the title describe which is being affected by the operator's choices.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Driver Multiplexer\
Tag Types\Communications\Link Management\Driver Multiplexer\

Related Information:

...Choices for Operator Interaction

Droplist Control Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control, Selector Switch.
* Does not use the Stye Settings tag.
The Droplist Control widget creates a drop-down list from which operators can select discrete values to output. (0 to 31)
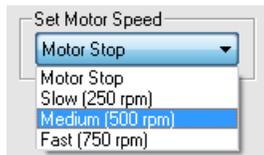
Regardless of the text used for each label, output values are integers beginning with 0 and increasing to the largest configured value, up to 31. The example shown here assumes that the underlying control will set the motor speed to 250 when a value of 1 is sent, 500 when a value of 2 is sent and so on.

The properties dialog for the control will appear similar to the following. Note that the control will always output values up to the largest text value configured. Should you skip one of the Text To Display boxes, that value will still be included in the drop list, but will display nothing to tell operators what it is.

The source of the text for each entry in the drop list may be any of: text, a tag value, the value of a selected parameter, or an expression.

Tip: To write any value you want, instead of sequential values from 1 to 31, then in many cases it is possible to use a String I/O tag with a String Droplist widget.
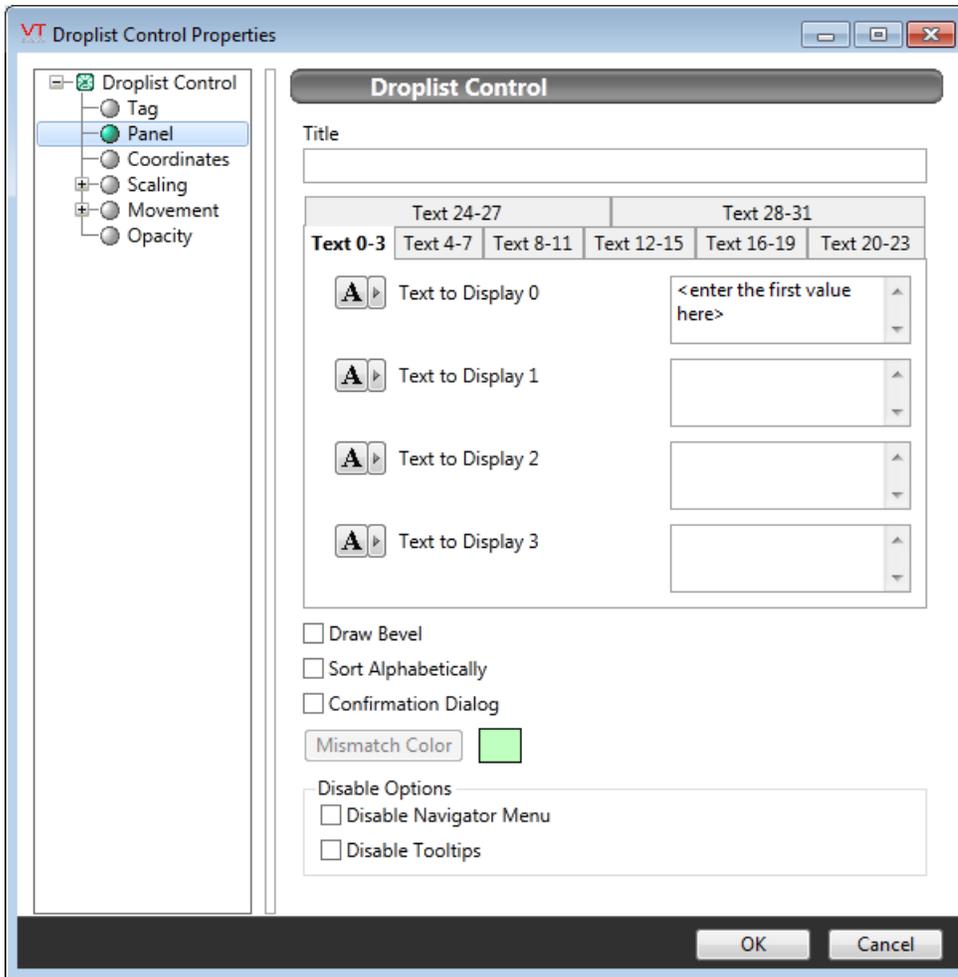
Title

Sets the title, should you wish to include one above the drop-down list. Enabled whether the bevel is drawn or not.

Draw Bevel

Encloses the drop down in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Sort Displayed List

Sort the list alphabetically by label. Output values stay with

their title and therefore, are likely to be in a scrambled order after the list is sorted.

Text to Display n

Sets the text to be displayed for each output value, n. The number of items in the list is controlled by the largest value for which the Text to Display field has been configured.

Sort Alphabetically

When checked, the droplist will be reorganized into alphabetic order by text for the output values, rather than being ordered by output value.

Confirmation Dialog

When checked, the operator will be prompted to confirm the control action.

Mismatch Color

Used only when the associated tag is a Selector Switch, or an Analog Input or Analog Status type, configured to write values. When the last value written does not match the value being read, the widget will display the configured mismatch color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analog controls\

Related Information:

...Choices for Operator Interaction

...String Droplist Widget

Duplexes Widget

Deprecated. Available only in applications that use the legacy VTScada layer.
Used by: DataFlow RTU.
* Does not use the Stye Settings tag.
The Duplexes tag widget is used to configure a display of the status of all equipment associated with a Data Flow RTU. The completed Duplexes widget shows an image of two pumps in a well, along with a series of data fields, control buttons and an alarm panel that shows critical inform-ation about the duplex station.
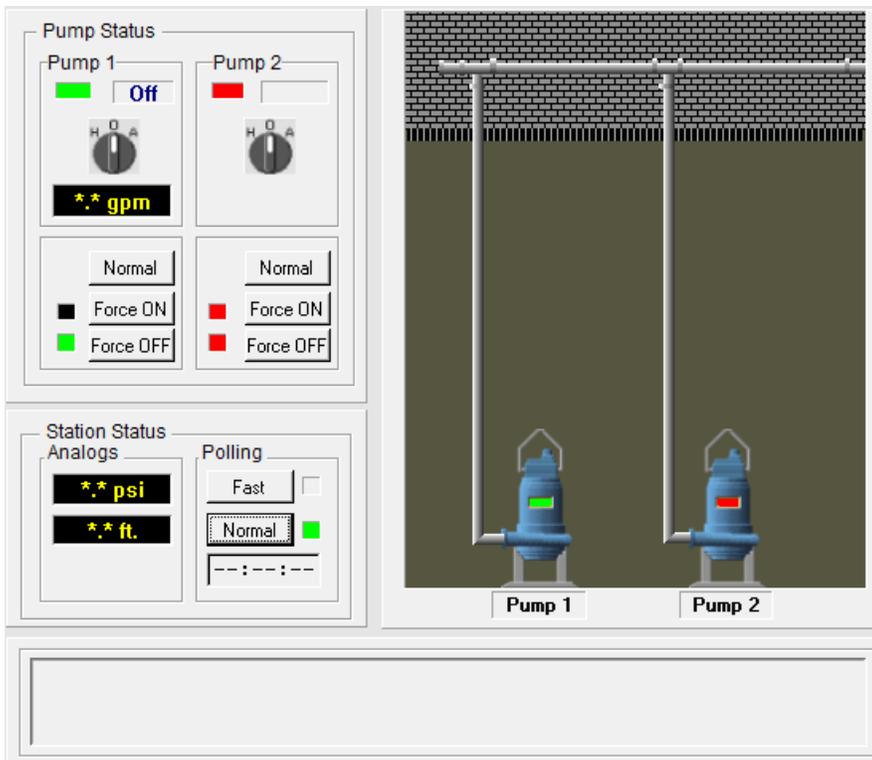
The properties dialog for a Duplexes widget:
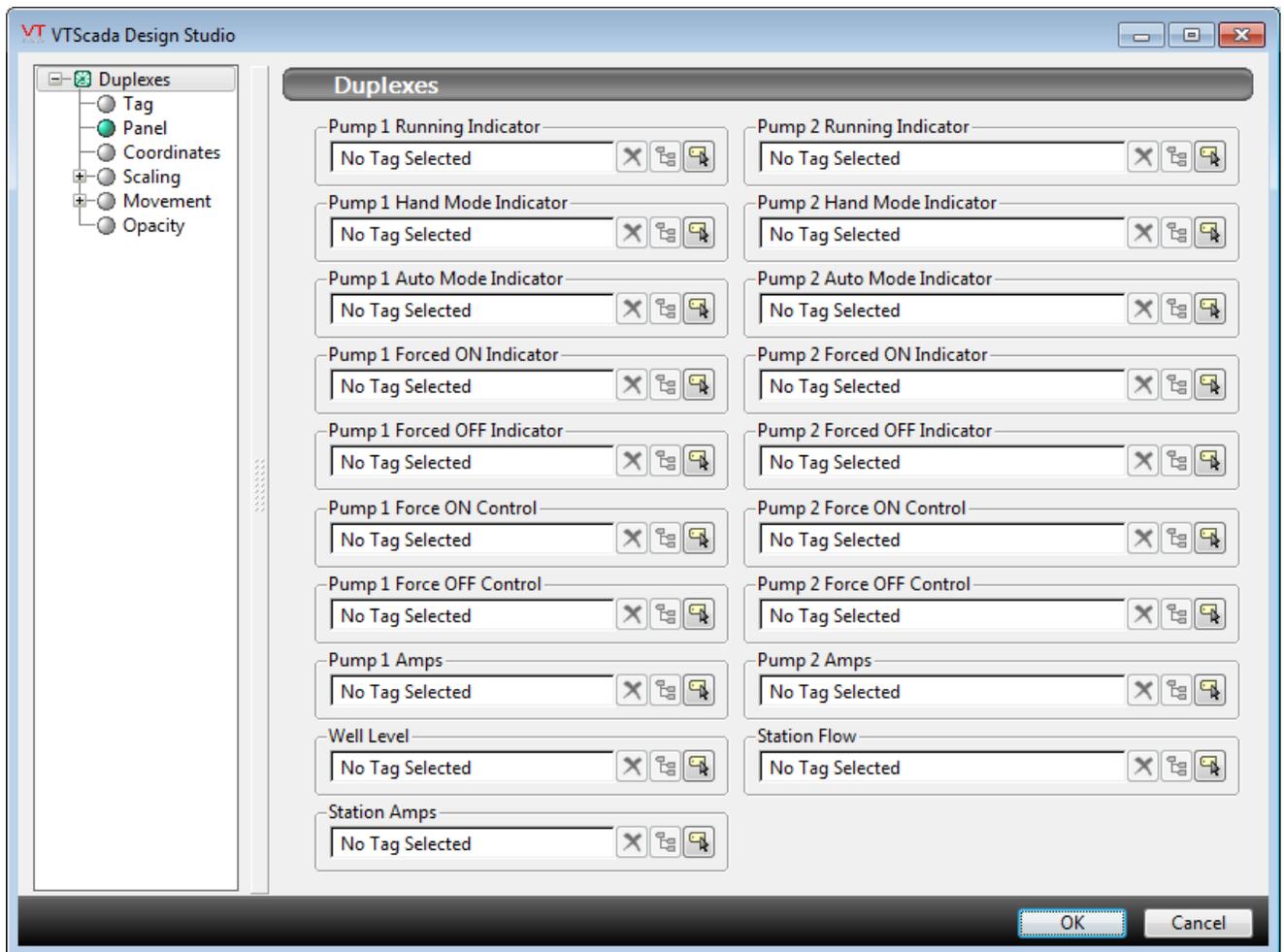
Pump 1 Running Indicator

Used to associate with this Data Flow RTU tag with the Pump Status tag responsible for reading the status of the first pump belonging to this station.

Pump 1 Hand Mode Indicator

Used to select the Digital Status tag responsible for reading the manual mode (hand) of the first pump belonging to this station.. This feature pre-dates the VTScada selector switch. Each

mode of the H–O–A switch in the Duplexes widget is represented by a separate Digital Status tag, all of which must be configured such that only one may be TRUE at a time.

Pump 1 Auto Mode Indicator

Used to select the Digital Status tag responsible for reading the auto mode of the first pump belonging to this station.

Pump 1 Forced On Indicator

Used to select the Digital Status tag responsible for reading the forced on mode of the first pump belonging to this station.

Pump 1 Forced Off Indicator

Used to select the Digital Status tag responsible for reading the forced off mode of the first pump belonging to this station.

Pump 1 Force On Control

Used to select the Digital Status tag responsible for setting the forced on mode of the first pump belonging to this station.

Pump 1 Force Off Control

Used to select the Digital Status tag responsible for setting the forced off mode of the first pump belonging to this station.

Pump 1 Amps

Select the Analog Status tag responsible for reading the Amps for this station.

Well Level

Select the Analog Status tag responsible for reading the level of the well for this station.

Pump 2 Running

Select the Pump Status tag responsible for reading the status of the second pump belonging to this station.

Pump 2 Hand Mode Indicator

Select the Digital Status tag responsible for reading the manual mode of the second pump belonging to this station.

Pump 2 Auto Mode Indicator

Select the Digital Status tag responsible for reading the auto mode of the second pump belonging to this station.

Pump 2 Forced On Indicator

Select the Digital Status tag responsible for reading the forced on mode of the second pump belonging to this station.

Pump 2 Forced Off Indicator

Select the Digital Status tag responsible for reading the forced off mode of the second pump belonging to this station.

Pump 2 Force On Control

Used to select the Digital Status tag responsible for setting the forced on mode of the second pump belonging to this station.

Pump 2 Force Off Control

Used to select the Digital Status tag responsible for setting the forced off mode of the second pump belonging to this station.

Pump 2 Amps

Select the Analog Status tag responsible for reading the Amps for the second pump belonging to this station.

Station Flow

Select the Analog Status tag responsible for reading the total flow for the station.

Widget Folders:

-- none --

Edit Property Checkbox

Not linked to any tag.
* Does not use the Stye Settings tag.
The Edit Property Checkbox widget provides a way for operators to switch a specific property on and off. You choose which property can be edited when configuring the widget. These widgets can allow access to certain properties that need to be updated, without granting operators full access to the Application Properties dialog.
If the property can have values other than zero and one, use the Edit Property Field widget.

☑ Echo Dialed Alarms

The properties dialog for the Edit Property Checkbox widget:
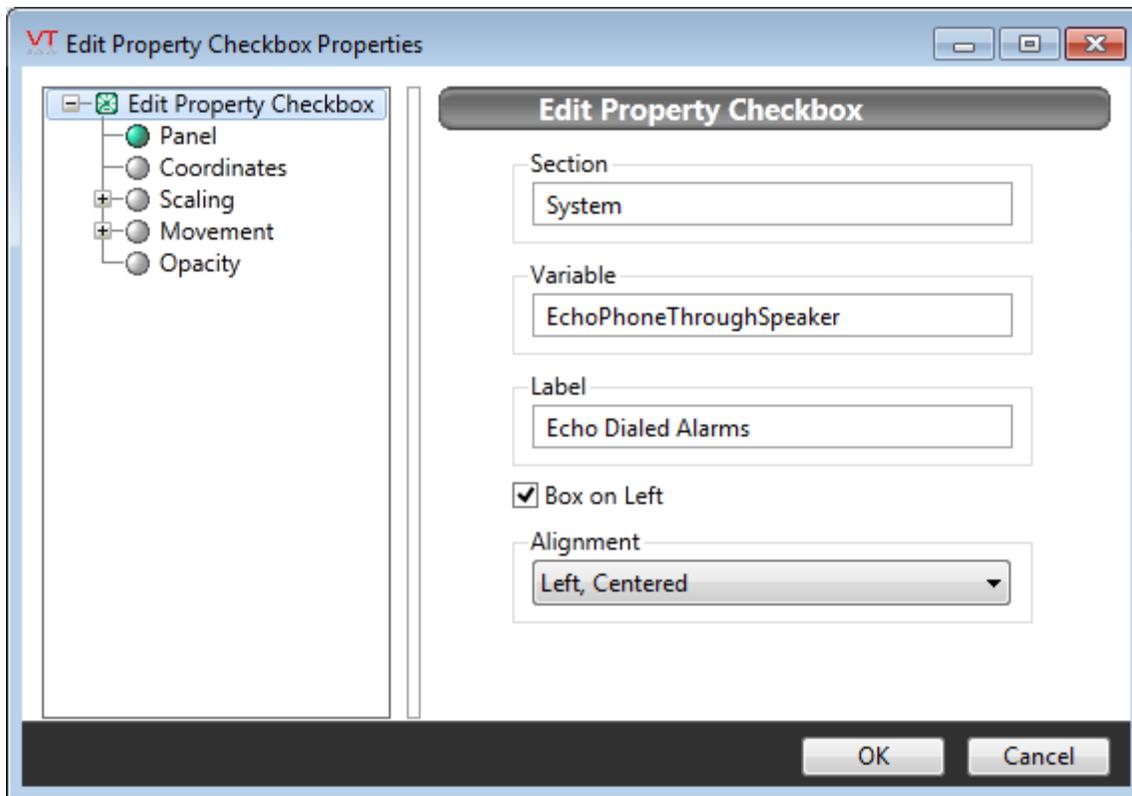
Section

> Application properties are organized by section. Properties that are placed in the wrong section will have no effect. The majority of properties that you are likely to use will be in the System section, but always confirm this before proceeding.

Variable

> The name of the property that can be changed using this field.

Label

> Should describe this edit property field to the operator.

Box on Left and Alignment

> Use these options to obtain the best appearance on a page.

Data Type

Restricts the values that the operator can provide. Must match the value type required by the configured variable.

Widget Folders:

Basic Components\Specialty\Settings Editing\

Tools\Standard Library\

Related Information:

...Edit Property Field

...See: "Application Properties" in the VTScada Admin Guide

Edit Property Field

Not linked to any tag.
* Does not use the Stye Settings tag.
The Edit Property widget is used to allow operators to change an application property value. You choose which property can be edited when configuring the widget. These widgets can allow access to certain properties that need to be updated, without granting operators full access to the Application Properties dialog.
If the property can have values of only zero and one, use the Edit Property Checkbox widget.

The properties dialog for the Edit Property Field widget:
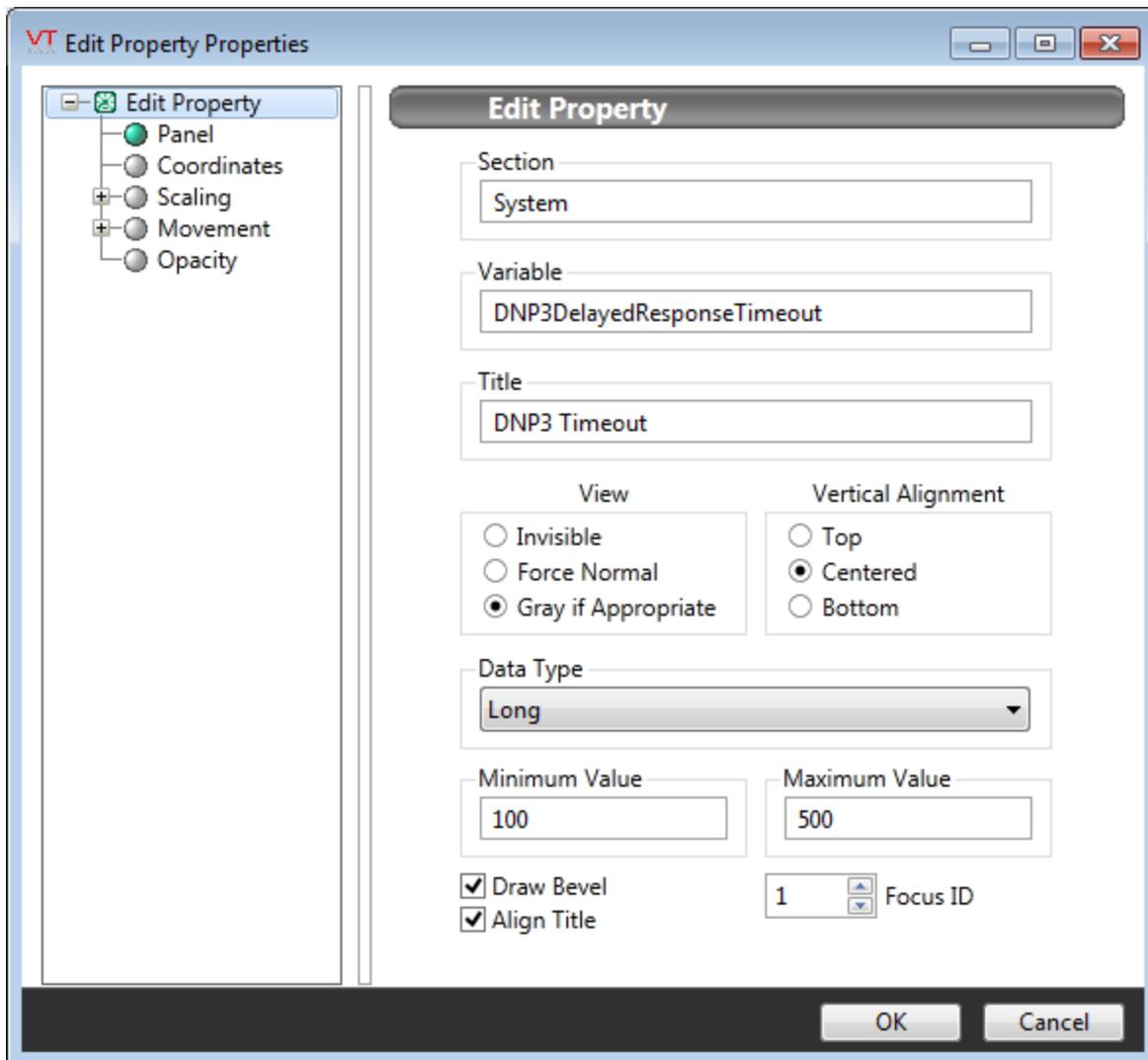
Section

Application properties are organized by section. Properties that are placed in the wrong section will have no effect. The majority of properties that you are likely to use will be in the System section, but always confirm this before proceeding.

Variable

The name of the property that can be changed using this field.

Title

Should describe this edit property field to the operator.

View & Vertical Alignment

In most cases, properties that are not immediately available for editing will be shown as gray. You may override this to either make the field invisible, or to force the property to appear normal even though it cannot be changed.
Use the vertical alignment options to obtain the best appearance on a page.

Data Type

Restricts the values that the operator can provide. Must match the value type required by the configured variable.

Minimum and Maximum Values

If the data type is numeric, these restrict the range of values that the operator can provide.
If the data type is text, these restrict the number of characters that may be typed.

Bevel, Title and Focus ID

Use Bevel and Title to improve the appearance of the field. You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Widget Folders:
Basic Components\Specialty\Settings Editing\
Tools\Standard Library\

Related Information:

...Edit Property Checkbox

...See: "Application Properties" in the VTScada Admin Guide

Elapsed Time Widget

Used by: Totalizer.

* Does not use the Stye Settings tag.

The Elapsed Time widget is used to display the current value of a Totalizer tag using a time format in hours : minutes : seconds.

Note: The "Elapsed Time" widget will only show "time that has elapsed" or "time that the equipment has been running" if that is what the totalizer is counting. To do this, the source for the totalizer must be a tag whose value is 1 when the equipment is on and 0 when the equipment is off.

Note: A Totalizer tag that is accumulating a value such as total flow does not also track its running time. The elapsed time widget cannot show information that is not being collected.

The properties dialog for the Elapsed Time widget will appear as shown:

Font

Select one of the font tags (or parameter linked to a font tag) that you with to use to display the text. The display will use the default system font if you do not select a font tag.

The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Horizontal Alignment

Sets the position of the text within the display area. You may select left-aligned, right-aligned or centered. To align the display area itself with other elements on the page, see: Alignment and Positioning Tools.

Type of Border

Examples of the three border types are shown at the beginning of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the color of the text.
The background color against which the text will be displayed can also be defined.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Text\

Analytics\Statistics\

Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Enable Polling Checkbox Widget

Used by: Polling Driver, DataFlow Systems RTU
* Does not use the Stye Settings tag.
Use this widget to give operators the ability to enable or disable polling
without needing to open the Polling Driver's properties dialog. Operators
must possess either the Modify Tags privilege or the Manual Data priv-
ilege in order to change this setting.

☑ Enable Polling

While polling is disabled, no communication will be sent or received
through the station. To switch from a timed poll to continuous com-
munication, use the Fast Poll option of the Polling Driver.
The properties dialog for the Enable Polling Checkbox object:
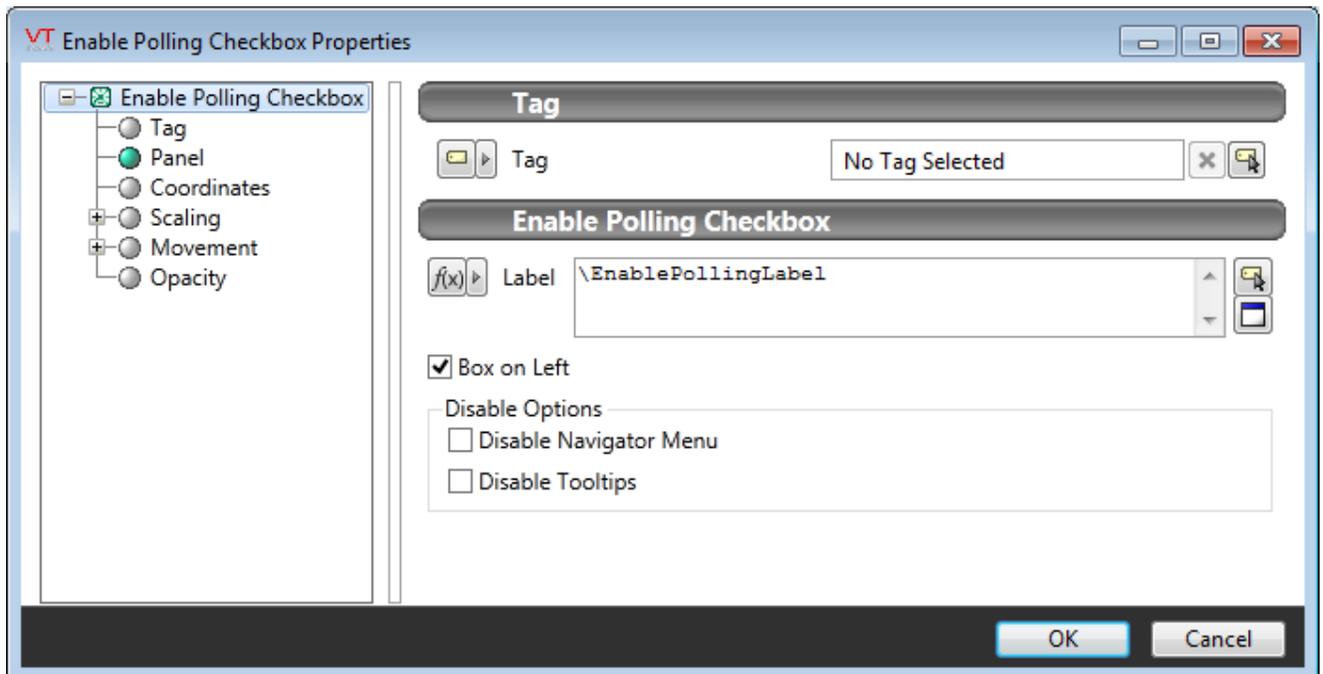
Label

The default value is a link to the application property, "\EnablePollingLabel", which defaults to "Enable Polling". You may provide any text or expression you prefer.

Box on Left

Controls whether the check box will be displayed to the left or the right of the label.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Tag Types\Communications\Link Management\Polling Driver

Related Information:

...Toggle Polling Button Widget

...Choices for Operator Interaction

Equipment / Status Color Indicator

* Can be linked to a Stye Settings tag.
Used by: Alarm, Alarm Status, Analog Control, Analog Input, Analog Output, Analog Status, Calculation, Counter, Deadband, Digital Control, Digital Input, Digital Control, Digital Status, Function, History Statistics, Modem, Network Status, Pump Status, Rate of Change, Roster, Selector Switch, Serial Port, SMS Appliance, SQL Logger, TCP/IP Port, Totalizer, Trigger, UDP/IP Port,
Status color indicators can be found in the Equipment folder of the Widgets palette. These are images of equipment that take their color from a Style Settings tag, based on the tag to which they are linked.

The properties dialog for a Status Color Indicator widget:

Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

Fast Scan Widget

Used by: Data Flow RTU, Polling Driver.
* Does not use the Stye Settings tag.
The Fast Scan widget is used to draw a button that the operator can use to control the poll rate for the associated station. The following example shows a polling driver site draw below the fast scan button. A green outer ring in the site draw indicates normal polling while a blue outer ring indicates that fast scan mode has been enabled.

There are two ways you may choose to use the Fast Scan tag widget:

- Configure the completed Fast Scan button as a toggle button that switches between fast poll mode and normal poll mode each time it is clicked. (As shown in the above example.)
- Draw two instances of the Fast Scan widget: The first as a button that can be clicked to put the station in fast poll mode and the second as a button that can be clicked to put the station in normal poll mode.

An example of the properties dialog for the fast scan widget follows:
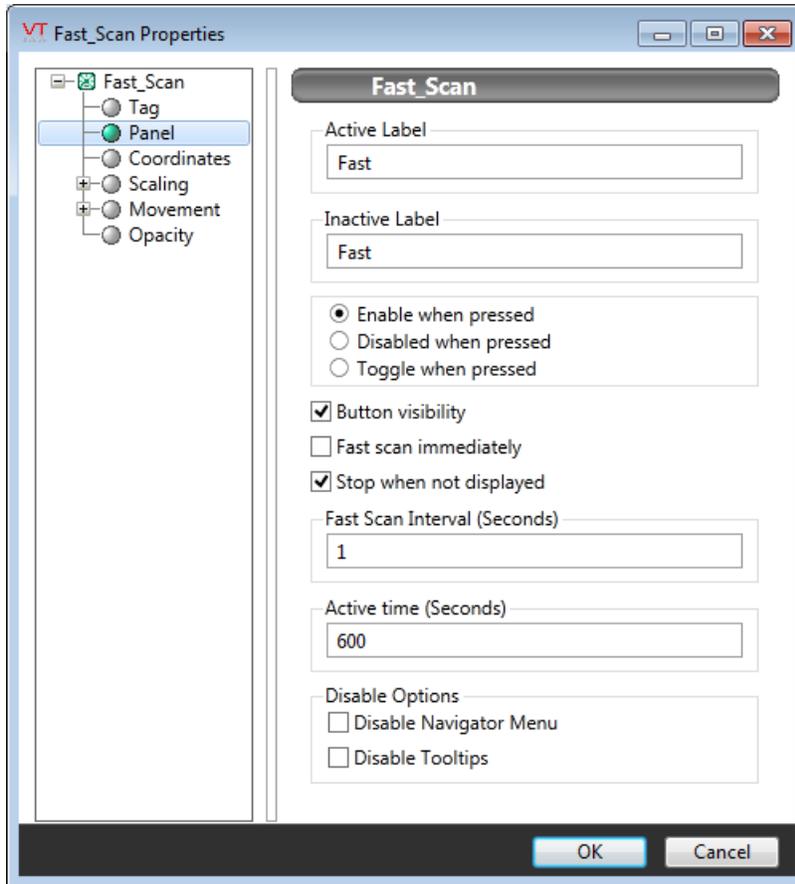
Active Label

Enter the label that should appear on the completed button while the associated station is being fast-polled.
Note that the label will not be displayed if the button visibility option is not checked.

Inactive Label

Enter the label that should appear when the associated station is not being fast polled. This option only applies if the Button Visibility check box has been selected.

Enable / Disable / Toggle When Pressed

This set of options controls how the button will function. If you want to draw two buttons: one to enable fast polling and one to disable it, then select Enable for the first instance and disable for the second. If you would like to draw a single button that toggles fast polling on and off, select the Toggle When Pressed option.

Button Visibility

When selected, the button and label will be visible. When not selected, the area occupied by the widget will be replaced with a hotbox, similar to the page change hotbox, visible only when the cursor is located within it.

Since the hotbox version of this button does not include a descriptive label and is not visible when the mouse is not above, it should be placed relative to a label or other screen object so that the operator can find it and will understand its function.

Fast Scan Immediately

Select this option to indicate that you wish the associated RTU to be polled in fast scan mode immediately when this object is displayed. That is, when the page or dialog upon which this object is placed has been opened.

If the Fast Scan Immediately check box is not selected, the completed object must be clicked in order for the fast poll to occur.

Stop When Not Displayed

Often used in conjunction with the Fast Scan Immediately check box, this option can be selected if the associated RTU should immediately return to normal polling mode when the page or dialog box containing this button is closed.
It can also be used as a fail-safe, regardless of how the Fast Scan Immediately option is set, forcing polling to return to normal when the operator leaves the current page, regardless of whether they remembered to set normal polling first.
If the Stop When Not Displayed check box is not selected, the completed object will continue to be fast polled when the page or dialog upon which it has been drawn is closed. Fast polling will continue for the amount of time indicated in the Active Time (Seconds) field (see following).

Fast Scan Interval

Use this option to set the interval (in seconds) at which the RTU will be polled when placed into fast poll mode by the Fast Scan object.

Active Time

Enter the length of time (in seconds) that the RTU will remain in fast poll mode when placed in fast poll mode by the completed object. If the Stop When Not Displayed check box (see above) has been selected, fast poll mode will stop when the page or dialog that the completed object has been drawn on is closed.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\Basic Controls\
Basic Components\Button Controls\
Analytics\Communications\Polling\

Related Information:

...Choices for Operator Interaction

Glance Meter Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.

The Glance widget represents the associated tag's value as a pointer needle within a bounding arc. There are no scales.

The properties dialog for the Glance widget:
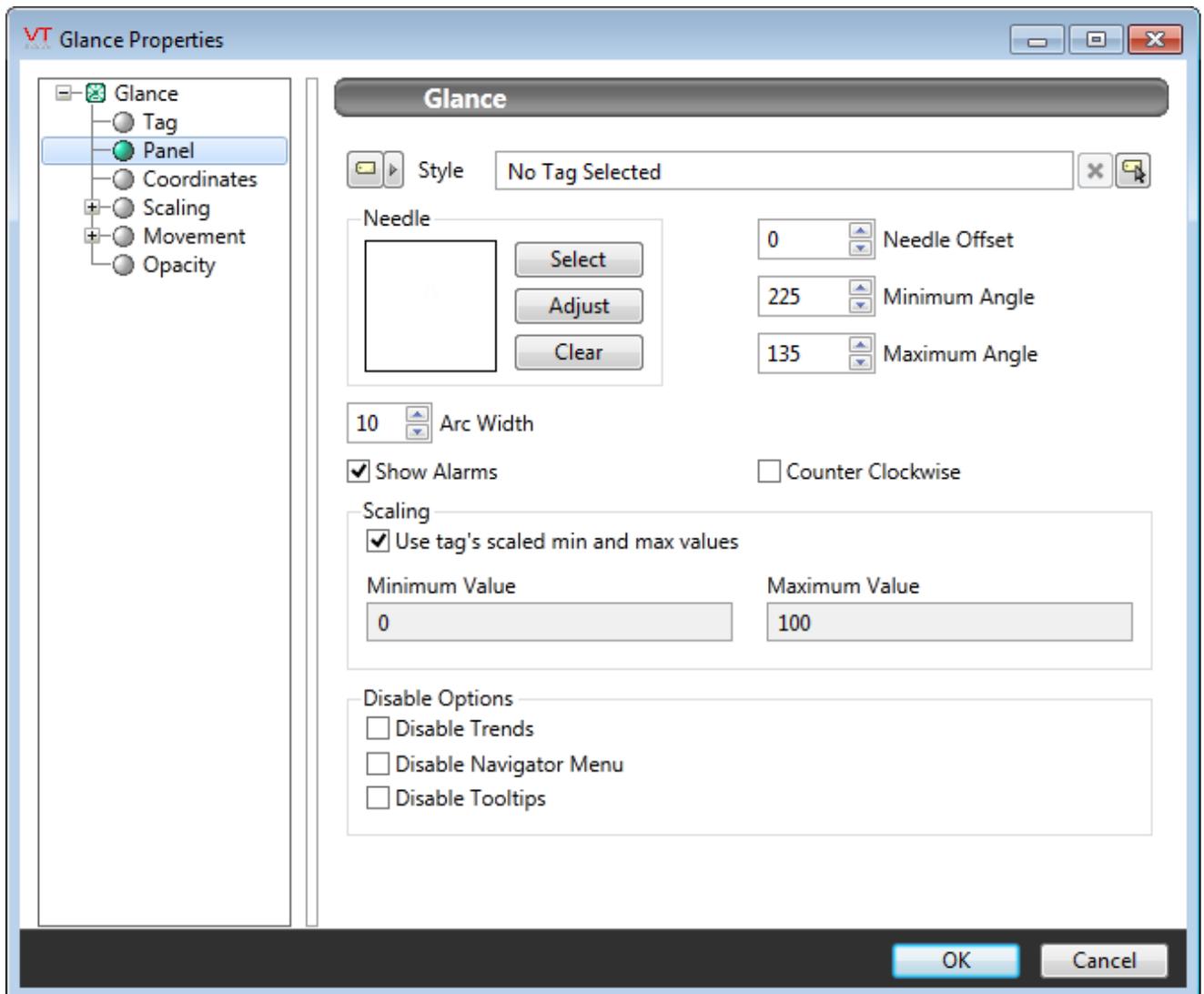
Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

The Tag field is only present when editing a completed widget object.

Style

Optionally, select a Style Settings tag to control the colors

shown by the arc, in response to the linked tag's value and alarm state. Style Settings Hierarchy.

Needle

The default image is ArrowNeedle, found in the image folder: Widget Parts >> Meter Parts >> Indicators >> Radial.

Minimum Angle & Maximum Angle

Sets the starting point and the ending point for the sweep of the arc. Defaults to 270 degrees for the start angle and 90 degrees for the end angle. The angle is measured in degrees, and increases in the clockwise direction.

Arc Width

Use to adjust the width of the arc. The width is measured from the outside edge towards the center.Values larger than 15 are not recommended.
The needle will scale in proportion to the available space within the arc.

Show Alarms

If selected, the arc will indicate the portion of the available range that would result in an alarm state, using the alarm color configured in the associated Style Settings tag.
If the linked tag is in an active alarm state, the needle will blink using the configured alarm color.

Show Alarms

If selected, values that are beyond the alarm setpoint will be shown using the designated color from the Style tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.

Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

Choices for Operator Interaction

Glass Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Glass Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Glass Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a

shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:
...Choices for Operator Interaction

Gradient Color Change Widget

Maintained for backward compatibility.
The Gradient Color widget was used to overlay any graphic object with a wash of color so that when the associated tag's value changes from 0 to 1 (or any value greater than 0.5 in the case of analogs), the selected

color range in the Gradient Color object superimposes itself over the object. Works only on the exact colors in the initial color range.

 Example showing pipes in VTS 10.2. This will not work with pipes created in VTScada 11.0 or later. If the object over which the Gradient Color object is layered is a solid color having an exact match to one of gradient's initial color range shades then the Gradient Color object washes the object with a solid color. If the underlying object is a pipe (created in a legacy application prior to release 11.0), the Gradient Color object will produce a multi-shaded color change.

The initial color range of the gradient color must match the object it is placed above in order for the final color to be displayed. In the case of a pipe, any of the pipe's shades will work for the gradient color's initial range. (Note: although rare, this can present an odd situation in the case where one of the pipe shades matches the background of the page. The result is that the gradient color cannot distinguish between the pipe and the page and so will be displayed in its entirety.)

The properties dialog for the Gradient Color widget:
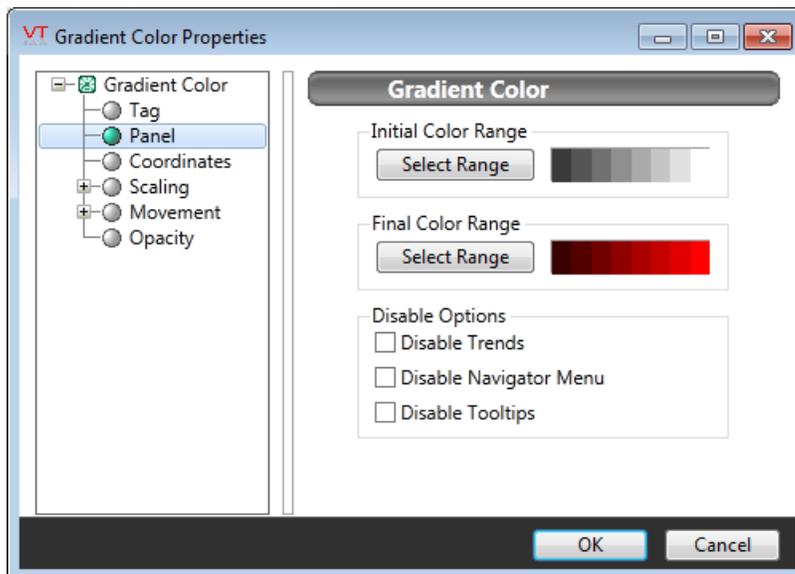
Initial Color Range

Opens the Select Color Dialog, from which you can choose a
color matching the object that the Gradient Color will overlay.
If the color you select matches any of the shades used for a
color range, as used when drawing pipes, then both that color
and the matching full color range will be used.
If the Initial Color does not match the overlaid object, the final
color will not be shown.

Final Color Range

Opens the Select Color Range Dialog, from which you can
choose the color that the overlaid object will turn when the tag
associated with this widget has a value of 1.
If the overlaid object is a solid color, gradient color will also be
a single color. In this case, the lightest color from the range
will be used.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators

Related Information:
Choices for Operator Interaction

HDV Button Widget

Not linked to any tag.
* Does not use the Stye Settings tag.
Create a button that opens an Historical Data Viewer window with a pre-
selected set of tag values, over a preset time range. The HDV page that
opens will include all of the standard tools for changing the tag selec-
tion, legend, time range and scale factor.
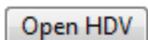
Open HDV

The properties dialog for the HDV Button widget:
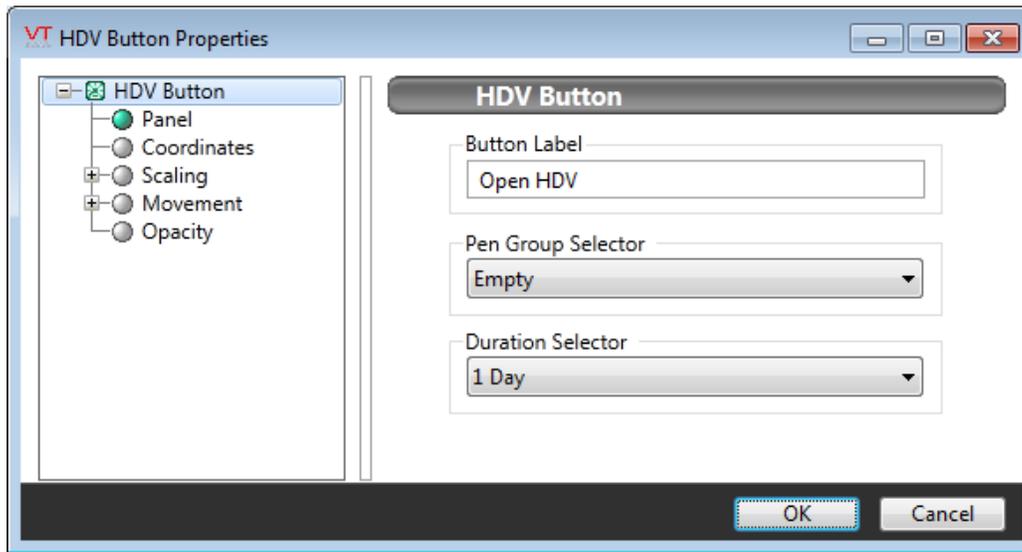
Button Label

The default label is generic. You are advised to modify this to include more information about what will be included in the plot
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Pen Group Selector

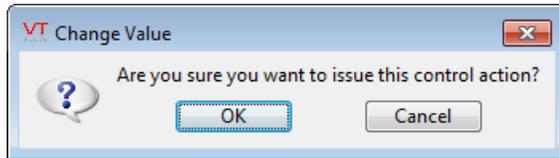An existing pen group must be chosen, otherwise this button provides nothing more than a link to an empty HDV page.
Pen groups are created in the Historical Data Viewer page, using the Tag Selector dialog.

Duration Selector

Choose the length of time that will be shown initially when the HDV page opens.

Widget Folders:
Analytics\Trends & Historical Data\
Tools\Standard Library\

HDV Hotbox Widget

Not linked to any tag.
* Does not use the Stye Settings tag.
Create a hotbox that opens an Historical Data Viewer window with a pre-selected set of tag values, over a preset time range. The HDV page that opens will include all of the standard tools for changing the tag selection, legend, time range and scale factor.
Since hotboxes are not visible until the pointer is over them, you are advised to add a label that will help operators find this control.
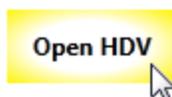
The properties dialog for the HDV Hotbox widget:
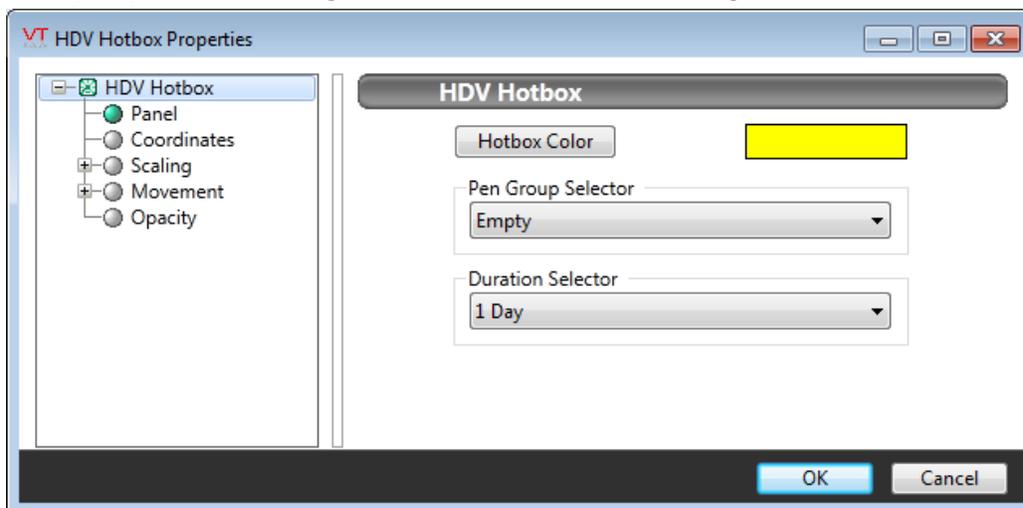
Hotbox Color

The default color is yellow. You may wish to choose a color that distinguishes this link from other types of hotbox.

Pen Group Selector

An existing pen group must be chosen, otherwise this button provides nothing more than a link to an empty HDV page. Pen groups are created in the Historical Data Viewer page, using the Tag Selector dialog.

Duration Selector

Choose the length of time that will be shown initially when the HDV page opens.

Widget Folders:

Analytics\Trends & Historical Data\
Tools\Standard Library\

Related Information:

...Save a Pen Group

...HDV Button Widget

...Plot Data Widget

...Draw HDV Widget

...Plot Trends and Graphs – Operator's Guide to the Historical Data Viewer

Historian Status

* Does not use the Stye Settings tag.
You can monitor the operational status of all of the Historian tags in your application using the Historian Status. This widget can be found in the System Status folder of the Widgets palette.
Note that the information displayed is from the perspective of the local machine, since Historians are both services and tags (much like device drivers).

There are no user-configurable options available in the properties dialog for this widget.

The Historian Status display will be blank while you are working in Idea Studio. Refer to the live page to see the status display.
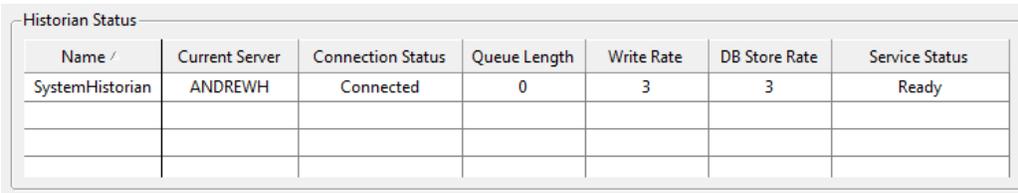
Historian Status						
Name /	Current Server	Connection Status	Queue Length	Write Rate	DB Store Rate	Service Status
SystemHistorian	ANDREWH	Connected	0	3	3	Ready

All Historian tags in your application will be listed. The grid can be sorted on any column and in either ascending or descending order. Click on the title of any column – once to sort in ascending order and twice to sort in descending order.

Current Server and Connection Status:

A Historian is a tag. On any given computer it may running as a primary server, backup server or a client. Current Server shows which machine is considered to be the server for the locally running Historian. The connection status refers to the connection between the Historian and its data store location.

Note: It is possible to monitor the connection to any particular server for a given Historian and therefore to trigger an alarm in the event that the connection is lost – see: Use an Expression to Monitor the Connection.

Queue Length

Shows the number of records in memory, waiting to be written. This number will normally fluctuate slightly with system activity. An ever-increasing value indicates that the hard drive or database cannot keep pace with the data being collected by the Historian and may be taken as a warning of data storage difficulties.

Write Rate

Shows the number of records per second being added to the queue. The DB Store Rate shows the number of records being processed by the database each second. A Write Rate that is consistently higher than the DB Store Rate indicates that the Historian is collecting data faster than the hard drive or database program is able to store it. If this is the case, you should look for ways to improve the performance of the data storage mechanism.

Service Status

The message in the Status column will change depending on the Historian's activity.
"Ready" indicates that, so far as can be found on the current computer, the Historian service is synchronized and is ready to accept records to be written.
"Waiting" indicates that the Historian is waiting for synchronization to finish, or that the service is unavailable.
If the Historian is synchronizing data across servers, the message will indicate the status of that synchronization.

Widget Folders:
Analytics\Trends & Historical Data\
Analytics\Statistics\
Analytics\Diagnostics\
Tools\System Status Tools\

Horizontal Button Widget

Used by: Selector Switch.
* Does not use the Stye Settings tag.
The Horizontal Button widget, one of the native widgets for Selector Switch tags, is a control method used to output one of three possible val-

ues. The values will be those configured for the three positions of the selector switch, 0 to 2 in order from left to right.

> Note: If the underlying selector has only two positions, then only the left and the middle button will be functional. The operator may click the right button, but this will have no effect on the switch position or output. Due to the confusion that this may cause, horizontal buttons should be used only with three-position selector switches.

Color is used to indicate that either a mismatch in state has occurred, or that an alarm is active as shown in the following examples.

Horizontal buttons in a normal operating state:
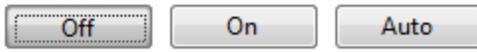
Horizontal buttons showing a state mismatch: (See: Selector Switch Tags, for a discussion of mismatch conditions.)

Horizontal buttons showing an active alarm:
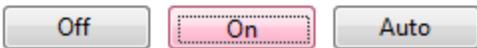
The following image shows the properties dialog where you can set the properties of the object.
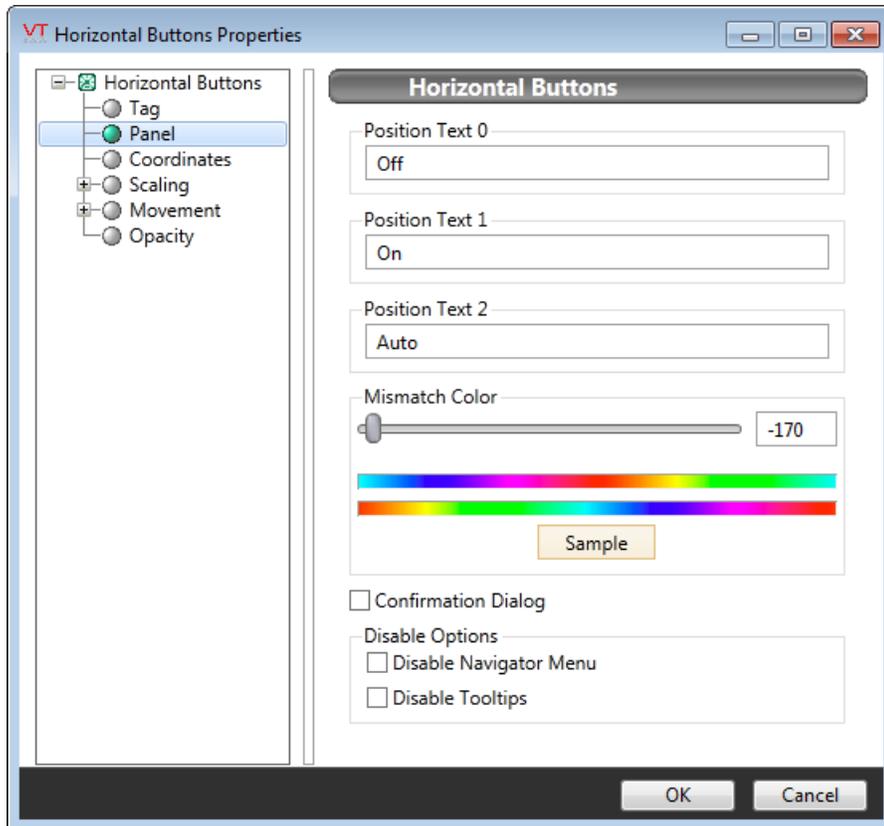
Position Text n

Set the labels that will appear on the buttons. The buttons are drawn from left to right and match the Position 0 through Position 2 settings of the associated selector switch.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Mismatch Color

Use the slider to set the color that will be superimposed on a button in the event of a position mismatch in the selector switch. A sample image displays the result as you adjust the slider.

Confirmation Dialog

If selected, the operator will be prompted to confirm every change made by clicking on the buttons. See also: Con-firmation Prompts for Output Tags.

An example of the confirmation dialog:

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Illuminated Toggle Switch Widget

* Can be linked to a Stye Settings tag.
Used by: Selector Switch.
See also: Toggle Switch Widget
The Illuminated Toggle Switch widget, one of the native widgets for Selector Switch tags, is a control method used to output either of two possible values. The values will be those configured for the first two of the three positions of the selector switch, 0 to 1.

To indicate to the operator that the image is a button, a highlight is drawn when the pointer is over the switch.

In the event that the position feedback from the Selector Switch indicates that the actual position is not the same as the requested position, the border of the image will blink using the mismatch color. You should select a color that stands out clearly on your page background, but red is not advised since it is commonly understood to mean that an alarm has been triggered.

The switch works by toggling between positions 0 and 1 whenever the operator clicks on the button. Should the tag be in an invalid state (which will not be unusual when a selector switch is first created) then it is not clear whether the first click should move the switch to the state 0 or the state 1 position. For this reason, a dialog will prompt the operator for an initial position.

The Properties dialog for an Illuminated Toggle.
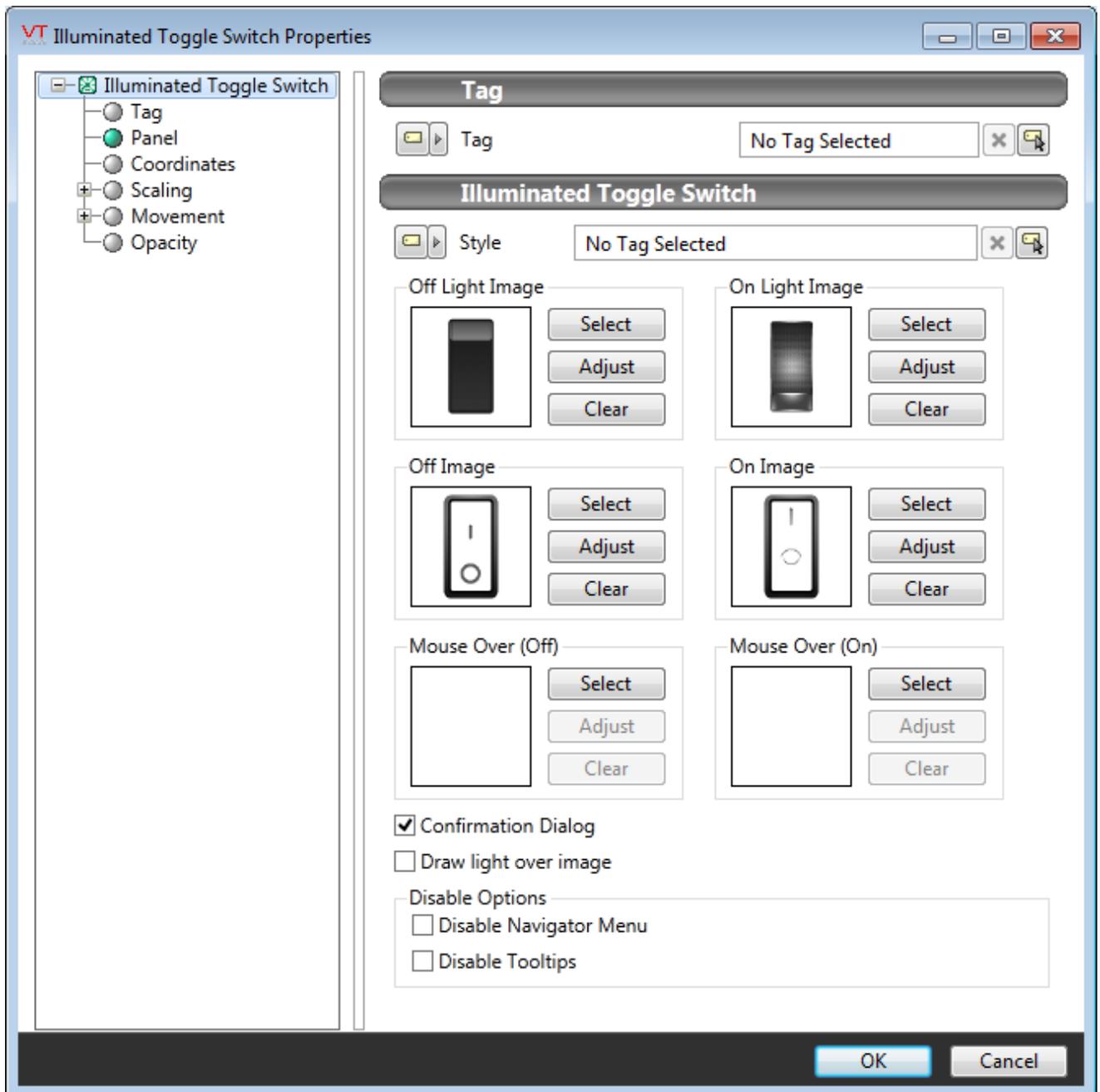
Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Off Light Image / Off Image

The image used for each position is composed of two parts: a basic light and a frame with semi-opaque lens that provides shape to that light.

Within the Bitmaps\Selector Switches\ folder of the Images palette, lights and lens images that are designed to work together will have matching names.

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Mouse Over

When left blank, the Illuminated Toggle will be highlighted when the operator moves the pointer over it. You may choose to display a new image instead. This field is used only when the pointer is over a toggle in position zero.

On Light Image / On Image

Similar to the Off Light Image and Off Image. Selects the light and lens images to be displayed when the toggle is in position one.

Mouse Over

The Mouse Over image to be used when the toggle is in position one.

Disabled

You may choose an alternate image to use when the operator does not have security privileges to use this tag.

Confirmation Dialog

When checked, operators must confirm that they intended to toggle this control.

Draw light over image

Reverses the order of the light and the lens. Typically, the lens will contain the On/Off label, therefore choosing this option will hide the label and accentuate the color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\

Image Change Widget

Used by: Alarm, Alarm Status, Analog Control, Analog In, Analog Out, Analog Status, Calculation, Counter, Deadband, Digital Control, Digital Input, Digital Output, Digital Status, Function, History Statistics, Modem, Network Status, Pump, Rate of Change, Roster, Selector Switch, Serial Port, SMS Appliance, SQL Logger, TCP/IP Port, Totalizer, Trigger, UDP/IP port.
* Does not use the Stye Settings tag.
The Image Change is a flexible widget, used to indicate a change in state of its associated tag by any of several methods.
It can:

- Display a different image for each state.
  For example, a valve might be displayed in a closed position when the value of the associated tag is 0 and in an open position when the value is 1.

- Display the same image for each of the tag's states, but with a different color applied.
  For example, a gray image can be chosen, then colorized so that it appears to be gray in state zero, but green in state one.

- Display the same image for each of the tag's states, but with a different color in the image's indicator area for each state
  A single pump image might be displayed with the indicator area at its base showing red when the pump is off and green when the pump is on.

- Represent the state of the associated tag with a color-changing area positioned over areas drawn with Active Orange (RGB 255, 160, 0).

Note: "2 state" versus "4 state". Not all tags will have all the states shown in these examples. Alarms and single-bit digitals have only state 0 and state 1 while 2-bit digitals also have states 2 and 3. The alarm state applies both to tags that have built-in alarms and to tags that have alarms attached to them. The alarm state will be active as long as

the corresponding alarm is unacknowledged, regardless of whether the alarm is active.

Note: As of VTScada release 11.1, default settings for each image will colorize the entire image rather than the indicator area. This makes the Image Change appear to work the same as an Equipment widget without actually having a link to a Style Tag.

To configure an image change as was done in legacy applications (where only the indicator area would change color) click the Adjust button for each state, then click the Reset to Defaults button within the Adjust Image dialog. This will remove the colorization from the entire image.

Examples follow for each of these ways of using the Image Change widget.

- Display a different color for each state:

State 0 State 1 Invalid State
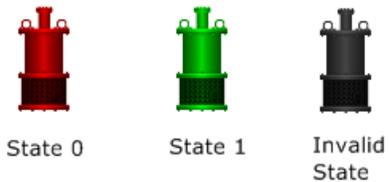
The properties dialog configuration to produce these images is as follows. Note that the colors are applied by colorizing each state, rather than using a Style Settings tag.
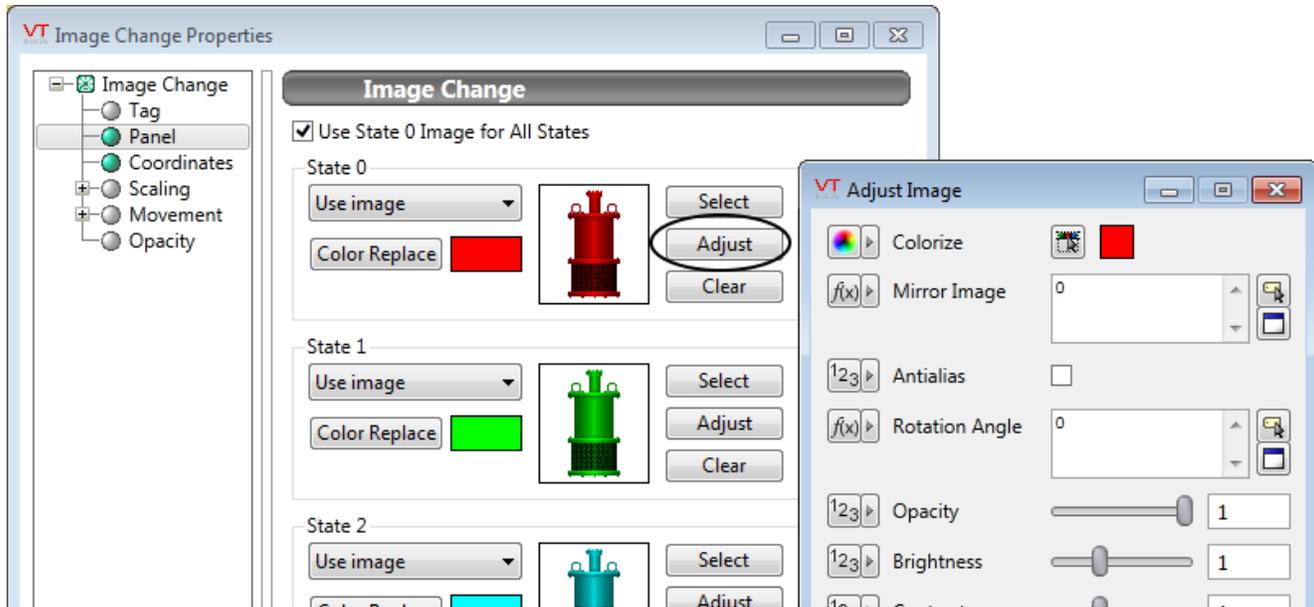
- Display a different image for each state:

| State 0 | State 1 | Invalid State |

The properties dialog configuration to produce these images is as follows. (The Restore Defaults button in the Adjust Image dialog has been used to remove colorization from each state in this example.)

- Display the same image for each of the tag's states, but with a different color in the image's indicator area for each state

Pump Status
Tag in State 0

Pump Status
Tag in State 1
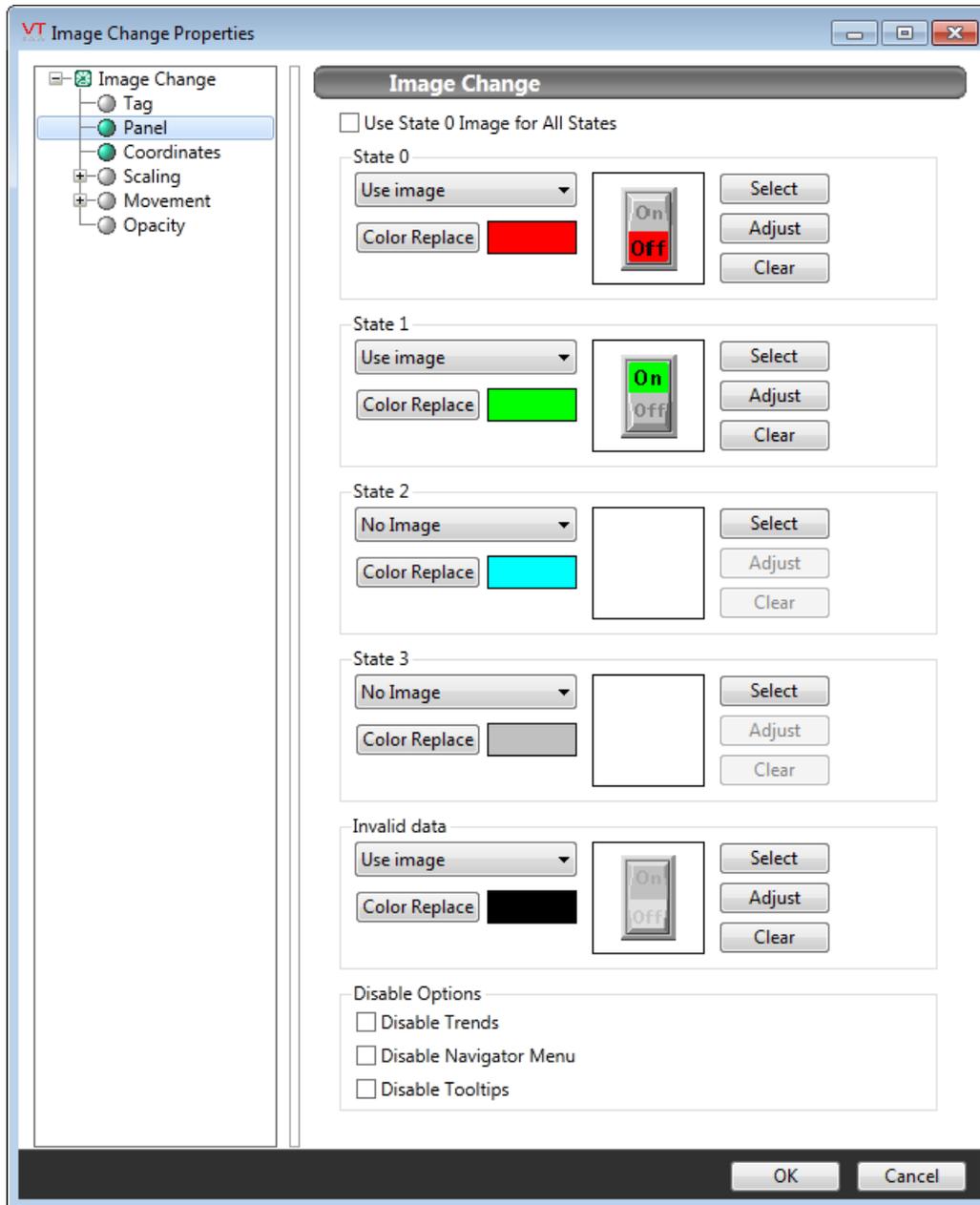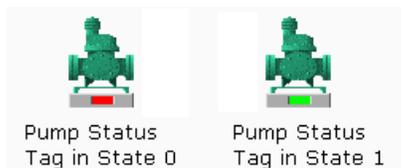
The properties dialog configuration to produce these images is as follows. Note that the Use State 0 Bitmap for All States check box has been selected.

(The Restore Defaults button in the Adjust Image dialog has been used to remove colorization from each state in this example.)

- Represent the associated tag with a color-changing area positioned over areas drawn with active orange (RGB 255, 160, 0).

Static Bitmap — Overlaid with Image Change — Tag in State 0 — Tag in State 1

Tag

The Tag field displays the name and description of the tag that is linked to this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Use State 0 Bitmap for All States

Check this box if you wish to use the image selected for State 0 for all states of the tag.
This option is often used when the selected image's indicator area (the small rectangle at its base) is to be replaced with a new color according to the state of the tag.

Use Image / No Image

Select whether or not to include an image for the Image Change.
If not using an image, you should select Color Replace, then position the Image Change over an area of Active Orange. Everything that is Active Orange below the Image Change will be replaced with the matching state's color.

Color Replace

Use the Color Replace option if the image includes a color indicator patch, or if you have selected the No Image option. (Color indicators patches must be areas of Active Orange.)

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the

image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators\
Basic Components\Color Indications\
Basic Components\Images\

Related Information:

...Choices for Operator Interaction

...Active Orange

Indicator Light Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm, Alarm Status, Analog Control, Analog Input, Analog Output, Analog Status, Calculation, Counter, Deadband Control, Digital Input, Digital Status, Digital Output, Digital Control, Function, History Statistics, Modem, Network Status, Pump Status, Rate of Change, Roster, Selector Switch, Serial Port, SMS Appliance, SQL Logger, TCP/IP Port, Totalizer, Trigger, UDP/IP Port.
Indicator Lights can take many forms. Each form is composed of a light and a frame. You can select different lights and frames for each indicator that you have drawn, mixing and matching the pieces.

The properties dialog for an Indicator Light:
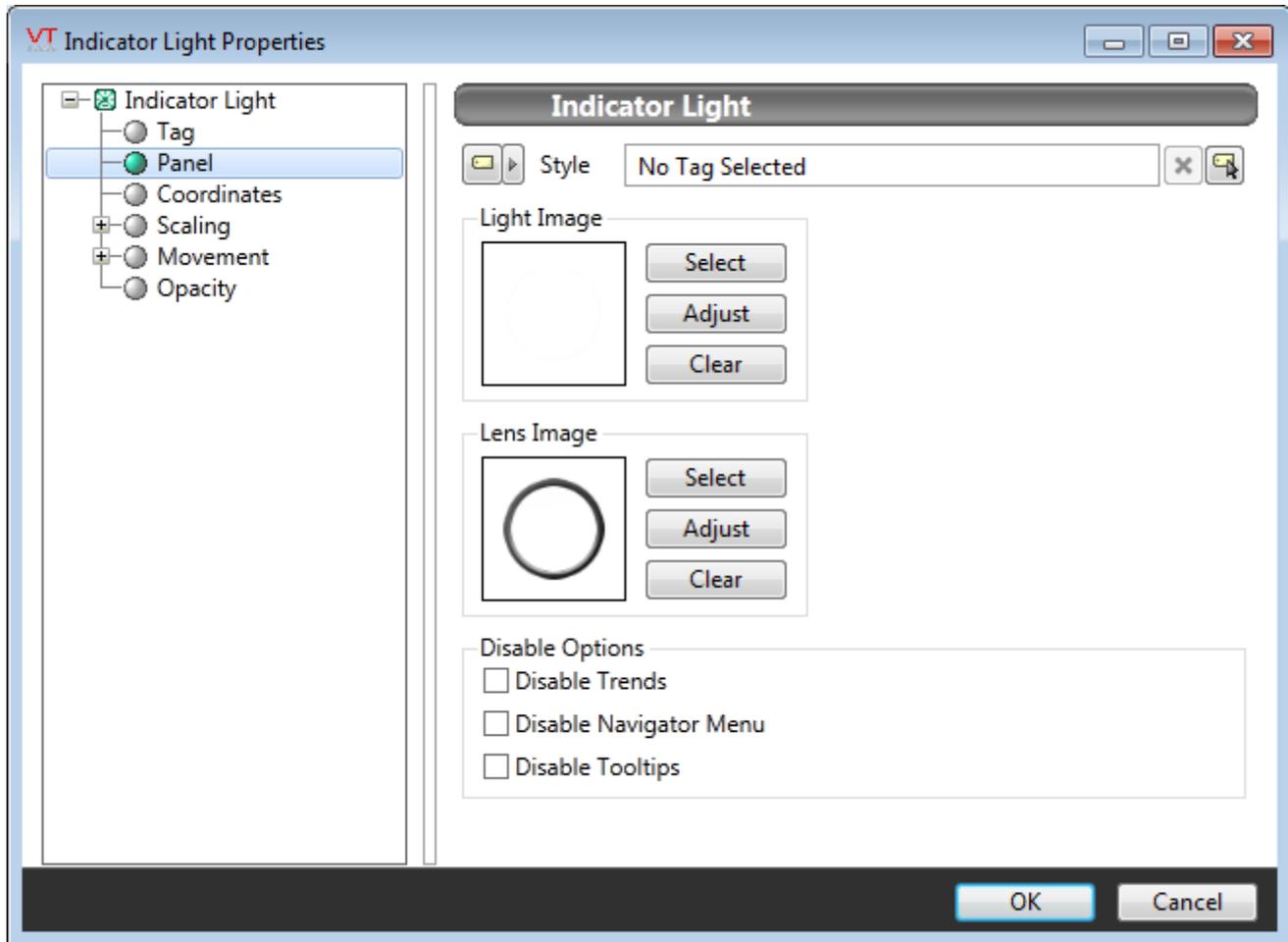
Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Light Image

Select an image that will change color according to the style setting. Must be a gray scale image. Light images and lens images that are designed to work together will have matching names in the Lights folder of images palette.

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Lens Image

Select an image that will frame the light, and provide a semi-opaque lens effect.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Indicators\Basic Indicators\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

Last Logged Value Widget

Used by: Counter, Totalizer.
* Does not use the Stye Settings tag.
The Last Logged Value widget is used to display the value last written to a log file (if any), for either a Totalizer tag or a counter tag.

No Border......	1704.4 L
Indented.......	1704.4 L
Beveled.......	1704.4 L
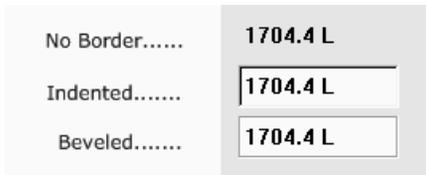
The properties dialog for the Last Logged Value:

Last Logged Value Properties

Last Logged Value

Font   No Tag Selected

1   Digits After Decimal

☑ Show Units

Horizontal Alignment
○ Align Left
◉ Align Center
○ Align Right

Type of Background
◉ Flat
○ Indented
○ Beveled

Colors
Text   Background

Disable Options
☐ Disable Trends
☐ Disable Navigator Menu
☐ Disable Tooltips

OK   Cancel

Tree items: Last Logged Value — Tag, Panel, Coordinates, Scaling, Movement, Opacity

Font

Select one of the font tags (or parameter linked to a font tag) that you with to use to display the text. The display will use the default system font if you do not select a font tag.
The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and

selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Digits after Decimal

Sets the accuracy with which the value is shown. This is for display only – there is no effect on the accuracy of the associated logged value.

Show Units

Sontrol whether or not the units defined within the Logger tag should be shown in this display.

Horizontal Alignment

Sets the position of the text within the display area. You may select left-aligned, right-aligned or centered. To align the display area itself with other elements on the page, see: Alignment and Positioning Tools.

Type of Border

Examples of the three border types are shown at the beginning of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the color of the text.
The background color against which the text will be displayed can also be defined.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

LCD 5x7 Matrix Widget

* Can be linked to a Stye Settings tag.
Used by: All numeric tags except Digitals.
The LCD 5x7 Matrix widget represents the associated tag's value as an dot matrix (LED) display. An active alarm will cause the numbers to use the alarm color.
The default colors look best against a dark background, but may be con-figured as required within the linked Style Settings tag.

The properties dialog for the LCD 5x7 Matrix widget:

Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Digits Before Decimal / Digits After Decimal

Sets the number of digits to be displayed. Digits beyond the range of the tag's current value are shown in dim shade, as configured in the linked Style Settings tag.

Show Alarms

When checked, values that exceed alarm setpoints will be drawn in the Style Settings alarm color (usually red).

Show Leading Zeros

Must be checked in order for leading zeros to be included in the display.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Gauges\

Text\

Related Information:

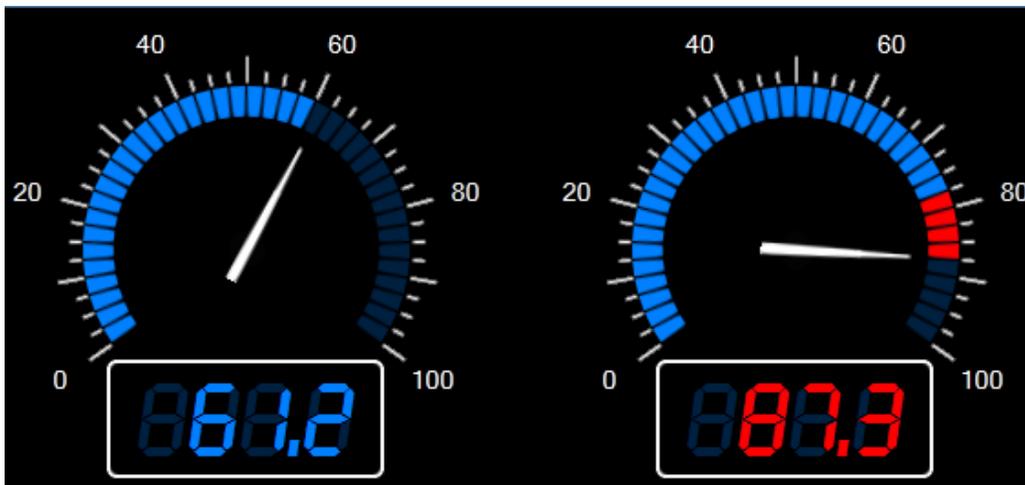...Style Settings Tags

...Choices for Operator Interaction

LCD 7 Segment Widget

* Can be linked to a Stye Settings tag.

Used by: Allan Bradley driver, Alarm Status, Analog Control, Analog Input, Analog Output, Analog Status, CalAmp Diagnostic driver, Calculation, Comm Link Sequencer, Counter, DataFlow Station, DDE Client, Deadband Control, Driver Multiplexer, Function, History Statistics, MDS driver, Modem, Motorola RTU driver, Omron driver, OPC Client driver, Polling driver, Rate of Change, Script, Serial port, Selector Switch, Siemens S7, SMS Appliance, SNMP driver, SQL Logger, Totalizer, Workstation driver.

The LCD 7 Segment widget represents the associated tag's value as an LCD display. An active alarm will cause the numbers to use the alarm color.

The default colors look best against a dark background, but may be configured as required within the linked Style Settings tag.

The properties dialog for the LCD 7 Segment widget:

Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Digits Before Decimal / Digits After Decimal

Sets the number of digits to be displayed. Digits beyond the range of the tag's current value are shown in dim shade, as configured in the linked Style Settings tag.

Show Alarms

When checked, values that exceed alarm setpoints will be drawn in the Style Settings alarm color (usually red).

Show Leading Zeros

Must be checked in order for leading zeros to be included in the display.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Gauges\

Text\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

LCD Arc Widget

* Can be linked to a Stye Settings tag.

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.

The LCD Arc widget represents the associated tag's value as a series of tick marks, a percentage of which are highlighted according to the tag's value. An option in the configuration enables the meter to display alarm values as red tick marks beyond the alarm set point.

The properties dialog for the LCD Arc widget:
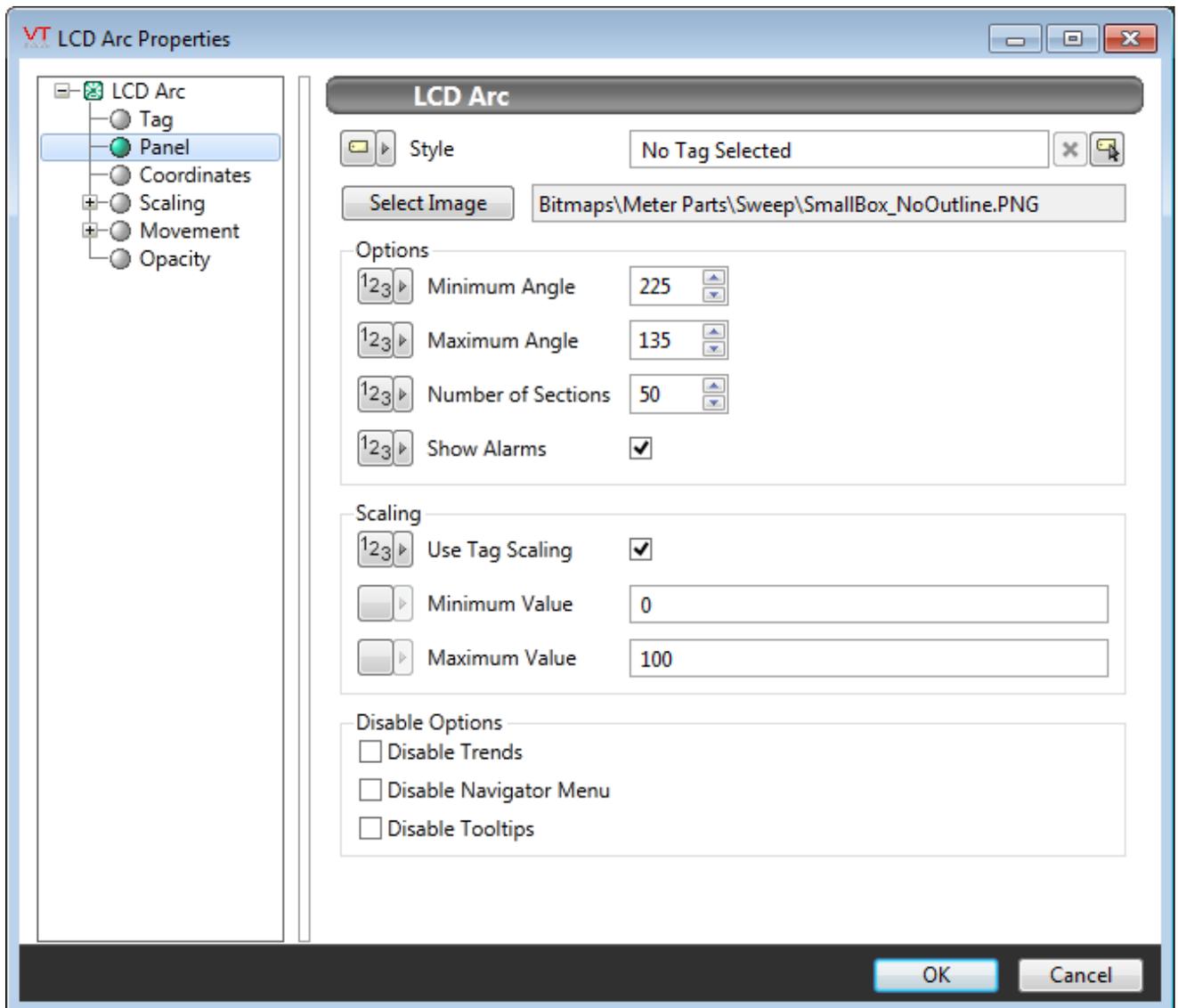
Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Minimum and Maximum Angles

Angles are measured clockwise from vertical. These two fields define the start and end points for LCD Arc's sweep.

Number of Sections

Controls the number of segments that will be shown within the meter's sweep. Defaults to 50, but should be reduced if a smaller sweep angle is defined.

Show Alarms

When checked, tick marks past the alarm set point will use the color configured in the style tag.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

LCD Bar Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Control, Analog Input, Analog Output, Ana-
log Status, Calculation, Comm Link Sequencer, Deadband Control, Func-
tion, History Statistics, Script, Selector Switch, SQL Logger.
The LCD Bar widget represents the associated tag's value as an
dot matrix (LED) display. If configured, an active alarm will cause the
dots above the alarm set point to use the alarm color. Two versions are
included in the palette: one with a single column of wide images and one
with an array of small images.
The default colors look best against a dark background, but may be con-
figured as required within the linked Style Settings tag.
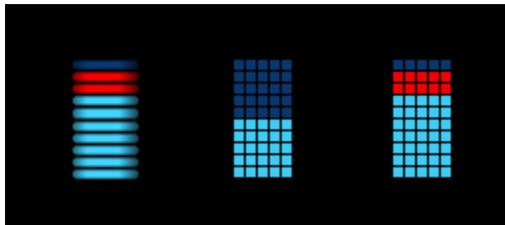
The properties dialog for the LCD Bar widget:
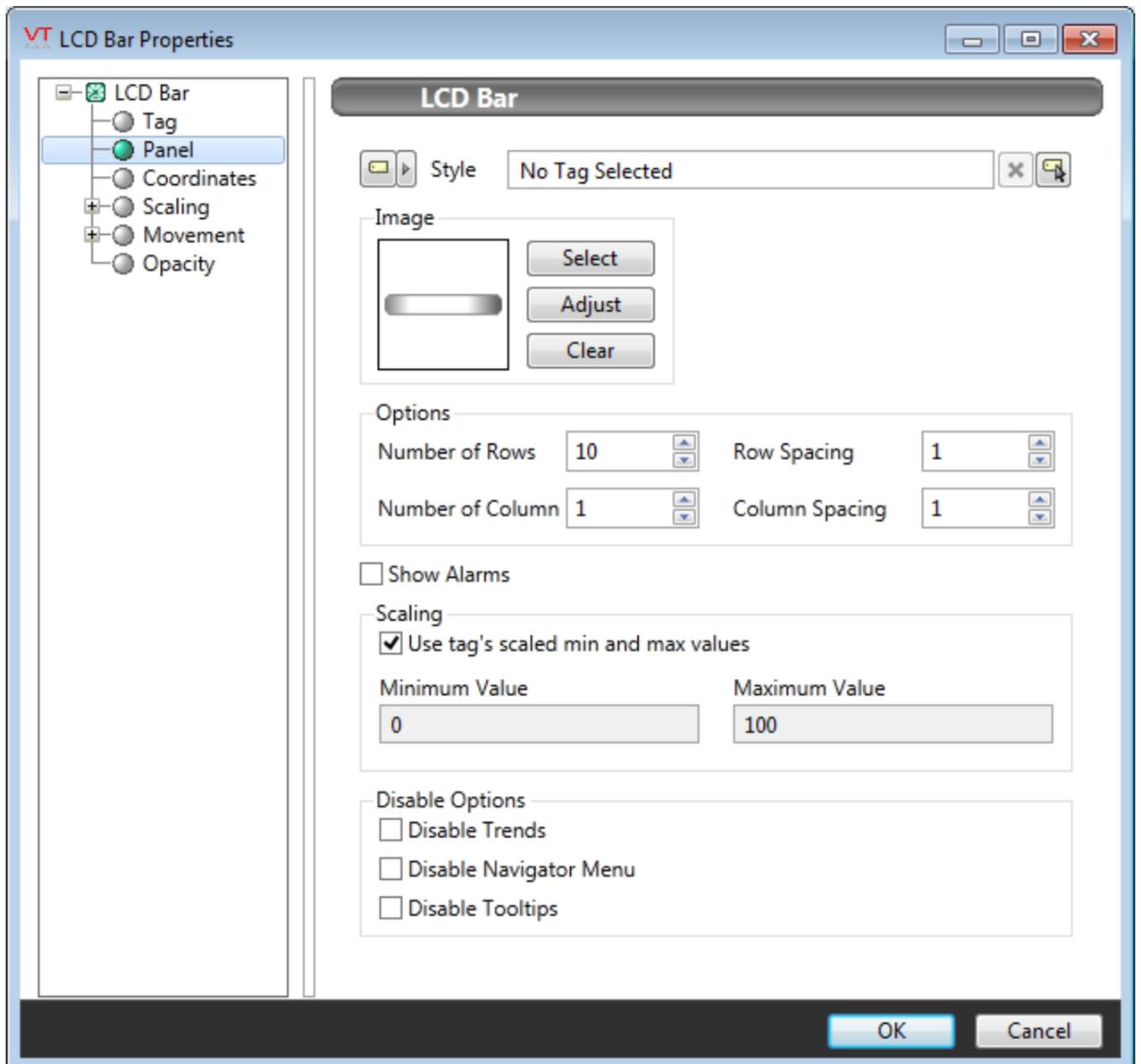
Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Select Bitmap

Choose the image to be displayed for each square of the bar.

The default image for the array version of this widget is a 5x5 pixel, light gray rectangle. For the series of stacked bars, it is a 65x11 pixel bar with shaded edges.

If choosing another image, ensure that it is a light shade of gray, in order to be compatible with the Style Settings tag. Suggested alternative images can be found in the images folder, Widget Parts >> Meter Parts >> LCD Elements.

Options

Set the number of rows and columns that will be displayed in the array, and the space between each element, measured in pixels.

Show Alarms

When selected, rows that represent values past the alarm set point are displayed using the alarm color in the Style Settings tag.

Scaling

In most cases, a widget will show the scaled values configured within the linked tag. You can override this setting to map the tag's values into any range you prefer to display.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Bars\

Related Information:

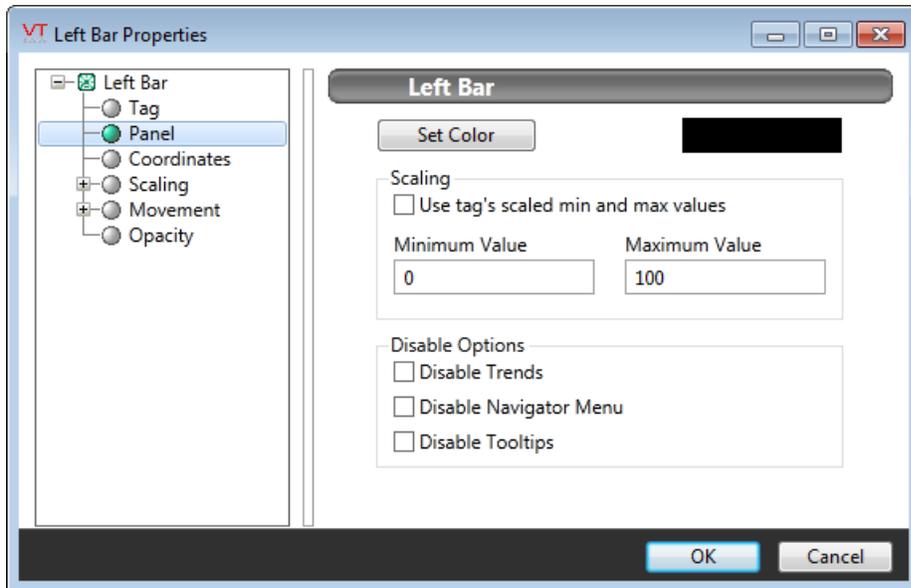...Style Settings Tags

...Choices for Operator Interaction

LCD Meter Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
The LCD Meter widget represents the associated tag's value as speedo-meter type of gauge with a series of tick marks, a needle, and a numeric scale. A percentage of the tick marks will be highlighted according to the tag's value. The numeric portion of the meter will flash red when the linked tag is in an alarm condition.
LCD Meter tends to look best on a dark background.

The properties dialog for the LCD Meter widget:
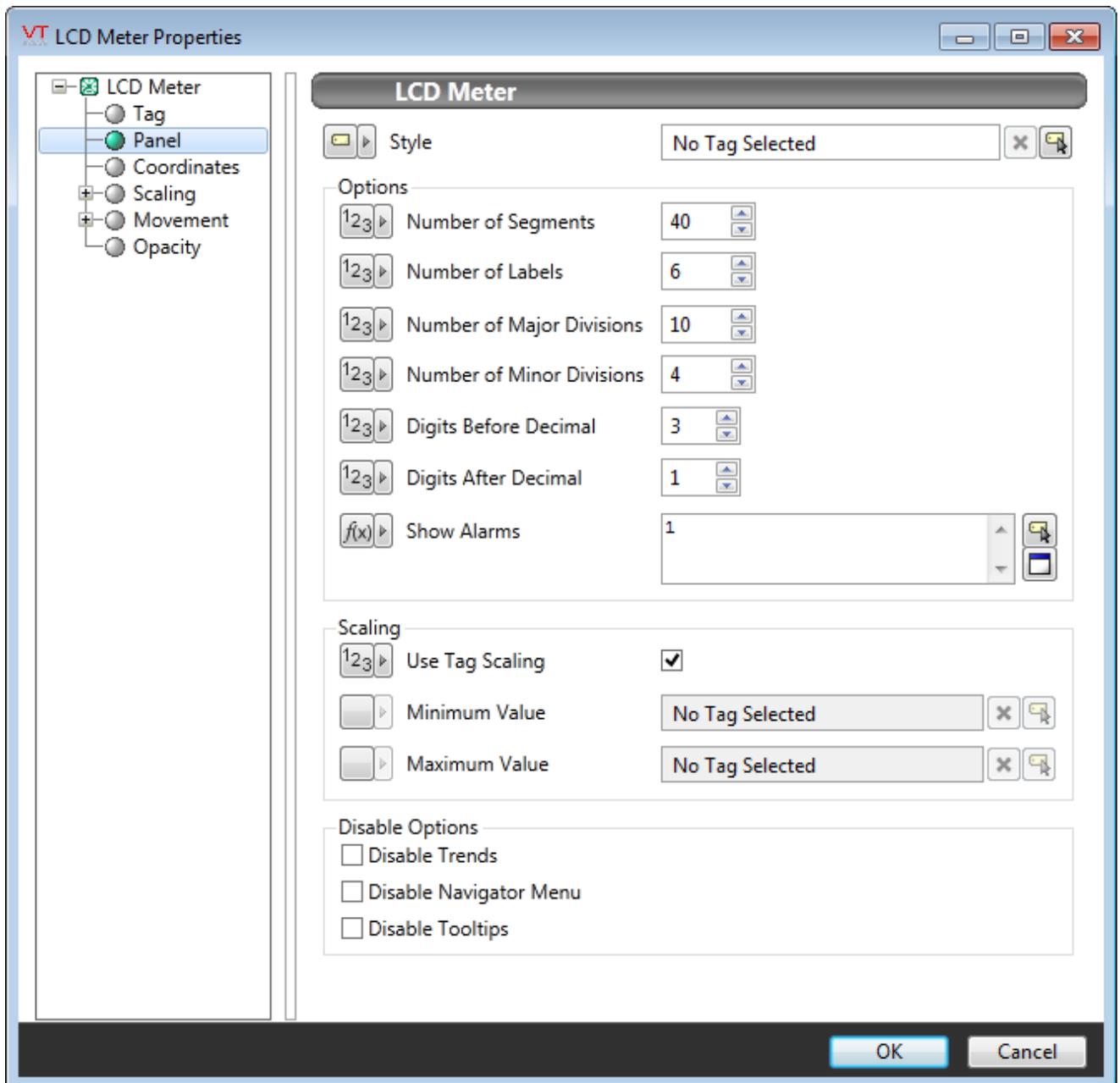
Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Number of Segments

The number of LCD tick marks displayed around the sweep of the meter. The meter will usually look best when this number matches (or is a multiple of) the number of Major Divisions times the number of Minor Divisions.

Number of Labels

Controls the number of text labels displayed around the meter.

Number of Major Divisions

The number of longer white lines around the meter.

Number of Minor Divisions

The number of shorter white lines between each major division.

Digits Before Decimal & Digits After Decimal

These two parameters work together to define the number of LCD digits displayed in the numeric portion of the meter. The display has room for a maximum of four digits. If adding a fourth digit before the decimal, you must remove the one following.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on

the meter. By default, these values will match the tag's scaled process values.

You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.
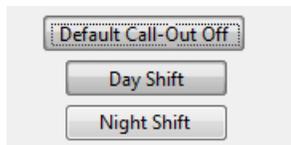
Widget Folders:

Gauges\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

LCD Ring Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
The LCD Ring widget represents the associated tag's value as a highly stylized meter, with a numeric display and a series of tick marks, the number of which fluctuates according to the tag's value. An option in the configuration enables the meter to display alarm values as red tick marks beyond the alarm set point.
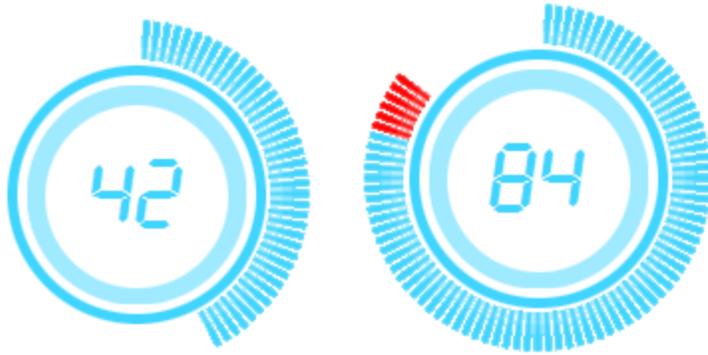
The properties dialog for the LCD Ring widget:

Style

Optionally, select a Style Settings tag to control the colors

shown by the light, in response to the linked tag's value and alarm state.

Show Alarms

When checked, tick marks past the alarm set point will use the color configured in the style tag.

Digits After Decimal

Sets the number of digits to show following the decimal point. Digits that would extend outside the ring will be clipped from view. You may need to decrease the font size in order to display a large number of digits.

Font Size

Measured in pixels rather than points, this sets the overall size of the font used for the numeric portion of the display.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full

range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

Left Bar Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Deadband Control, Function, History Statistics, Script, Selector Switch, SQL Logger.
* Does not use the Stye Settings tag.
The Left Bar widget is used to display a tag's value as a horizontal bar, the length of which increases from the right towards the left as the tag's value increases. When the tag's value is at a minimum, the bar will be shown as a thin line at the right edge of its range. When the tag's value reaches its maximum, the widget will be displayed as a rectangle, filling the range.

The graphic editor for the Left Bar widget:

Set Color

Opens the Select Color Dialog from which you can select a color for this object.

Low Scale Range and High Scale Range

In most cases, these fields will be display-only, showing the Scaled Process Data Min and Scaled Process Data Max values configured for the associated tag.

These fields become editable when the widget is used to display a function or Calculation tag. The most common use is to provide an output range of 0 to 100 (%) given a data source whose numbers do not match these values (perhaps ranging from 20 to 80, for example). Set the low scale range to the value expected from the data source when it is at its minimum and the high scale range to the value expected from the data source when it is at its maximum.

Disable Options

Disable selected operator-interaction features of this widget.

Linear Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Linear Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value. This meter displays the tag's value as text, and includes the engineering units in addition to the dial.
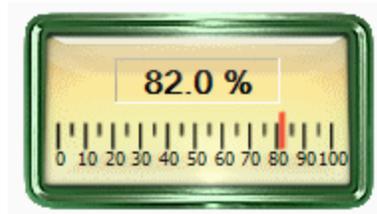
The properties dialog for the Linear Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.

You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Make Active Widget

Used by: Roster tag.
* Does not use the Stye Settings tag.
Used in association with the Alarm Notification System, the Make Active tag widget will create a button that can be clicked to select the associated Roster tag as the active roster. This is used to select who will be contacted by the call-out system in the event that an alarm has been triggered and has gone unacknowledged for a defined period of time.

The properties dialog for the Make Active widget:
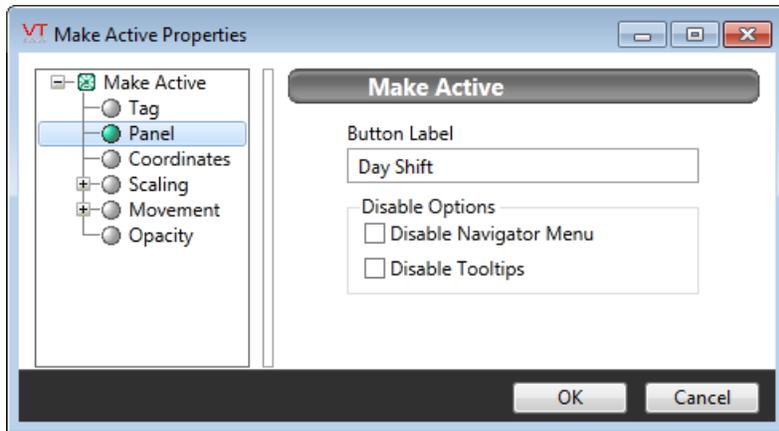
Button Label

Provide the text (if any) that will be displayed on the button.
The widget must be linked to a Roster tag.
If the label is to include an ampersand (&), then the & character
will need to be doubled in order to appear.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Specialty\Roster Control\

Related Information:

...Choices for Operator Interaction

Metal-Themed Meter Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm
Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Metal widget represents the associated tag's value as photorealistic
meter.

The properties dialog for the Metal widget:
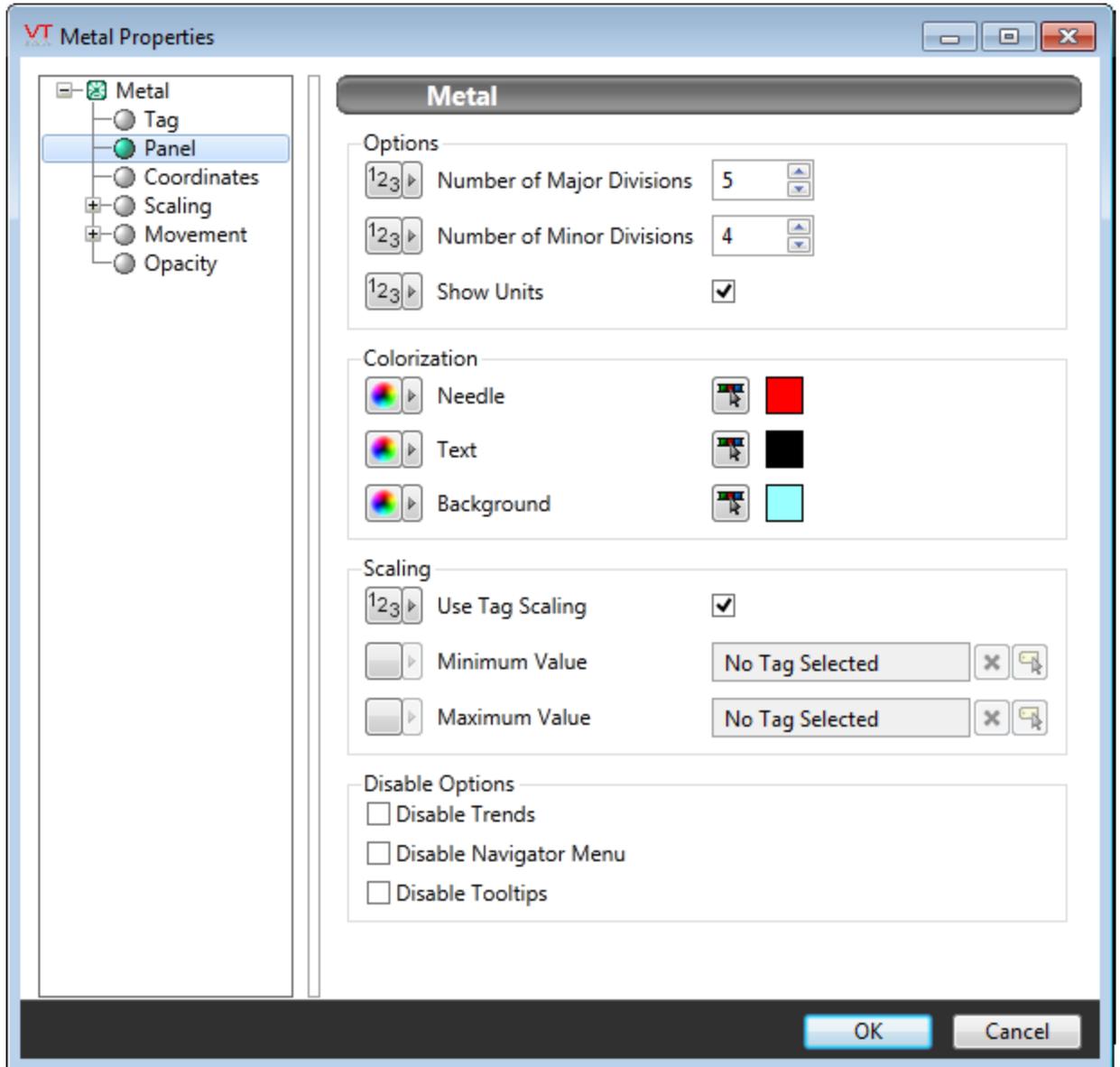
Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

The Tag field is only present when editing a completed widget object.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and 5.

Choose whether to include the text used for engineering units within the gauge. Defaults to show units.

Colorization

Use the color selector dialogs to set the color of the needle, text and gauge background. This widget does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.

Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

Choices for Operator Interaction

Meter 1 Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband Control, Function, History Statistics, Rate of Change, Script, Selector Switch, Totalizer SQL Logger.
* Does not use the Stye Settings tag.
The Meter 1 widget is used to represent the associated tag's value as a round meter whose dial fluctuates according to the tag's value, and which also displays the value as text. The engineering units of the tag (% in this example) are also shown.

The properties dialog for the Meter 1 widget:
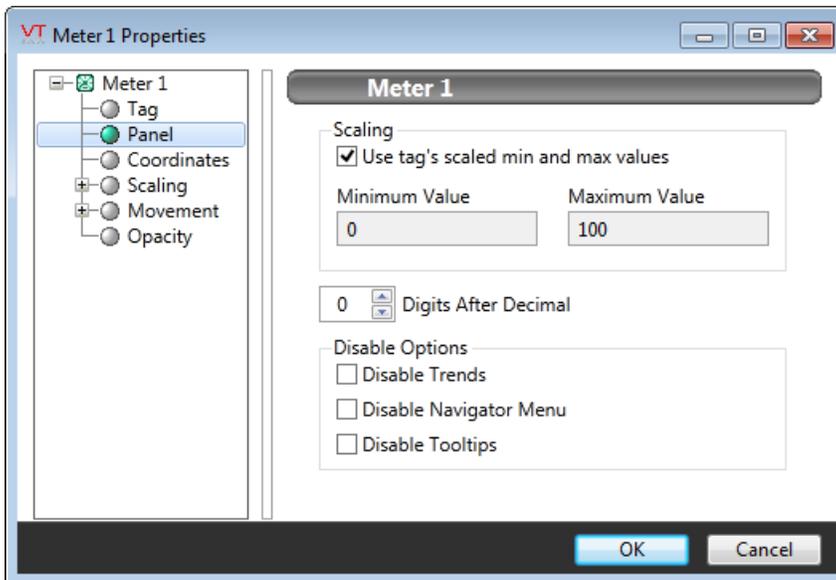
Low Scale Range and High Scale Range

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag as a meter 1 with a low scale range of 0 and a high scale range of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Digits After Decimal

Used to adjust the displayed accuracy of the meter. The displayed value will be rounded to the number of decimal points set. This is for display purposes only and has no effect on the tag's value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\

Related Information:
...Choices for Operator Interaction

Meter 2 Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband Control, Function, History Statistics, Rate of Change, Script, Selector Switch, Totalizer SQL Logger.
* Does not use the Stye Settings tag.

The Meter 1 widget is used to represent the associated tag's value as a half-round meter whose dial fluctuates according to the tag's value, and which also displays the value as text. The engineering units of the tag are also shown.
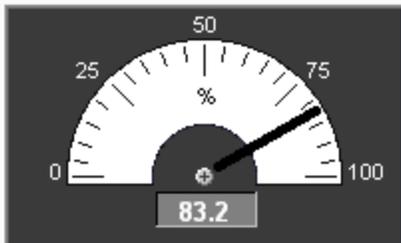
The properties dialog for the Meter 2 widget:
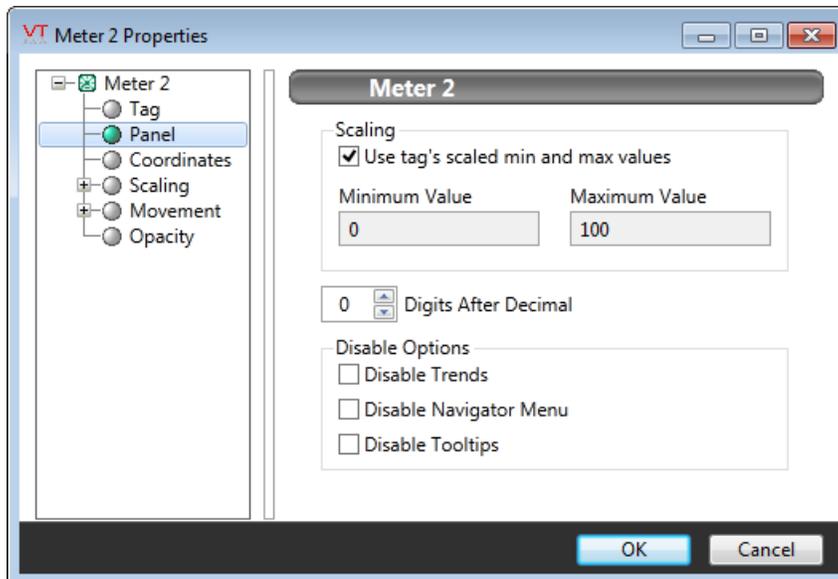
Low Scale Range and High Scale Range

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag`s scaled process values.

You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag as a meter 2 with a low scale range of 0 and a high scale range of 100 will result in a meter that shows

a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Digits After Decimal

Used to adjust the displayed accuracy of the meter. The displayed value will be rounded to the number of decimal points set. This is for display purposes only and has no effect on the tag's value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\

Related Information:
...Choices for Operator Interaction

Meter 3 Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband Control, Function, History Statistics, Rate of Change, Script, Selector Switch, Totalizer SQL Logger.
* Does not use the Stye Settings tag.
The Meter 1 widget is used to represent the associated tag's value as a numeric display, including the engineering units of the tag (% in this example) in the lower right corner.

The properties dialog for the Meter 3 widget:
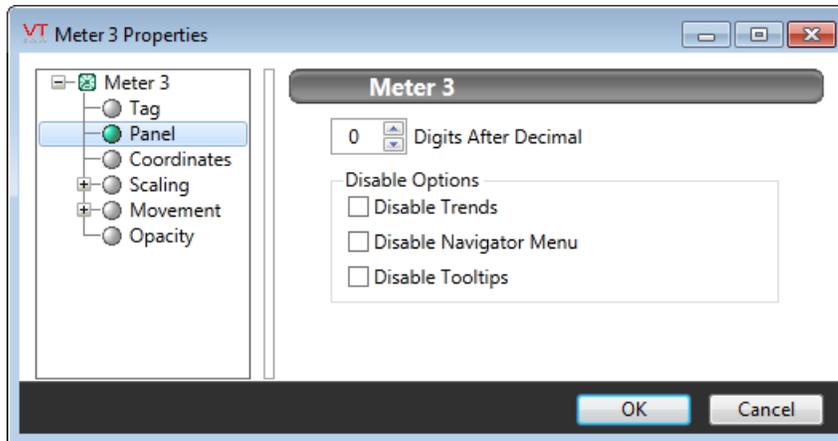
Digits After Decimal

Used to adjust the displayed accuracy of the meter. The displayed value will be rounded to the number of decimal points set. This is for display purposes only and has no effect on the tag's value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Gauges\Basic Analogs\

Related Information:

...Choices for Operator Interaction

Modem Indicator Widget

Used by: Modem tags.
* Does not use the Stye Settings tag.
The Modem Indicator widget displays a graphic of an active modem control, showing the name of the associated Modem tag. When the associated modem is dialing out or answering an incoming call:

- The widget will flash (green and red activity lights will flash on the control when the modem is active).

- The text "Incoming Call" will be displayed (for incoming calls).
- The number being dialed will be displayed (for outgoing calls).

If the modem has failed for whatever reason, a red X will be drawn through the image.

A modem can be reset by clicking on the widget when the application is in operation mode. A modem can also be marked as failed or unavailable using the Disable option of the shortcut menu (right-click on the widget when the application is in operation mode and select the Disable option).
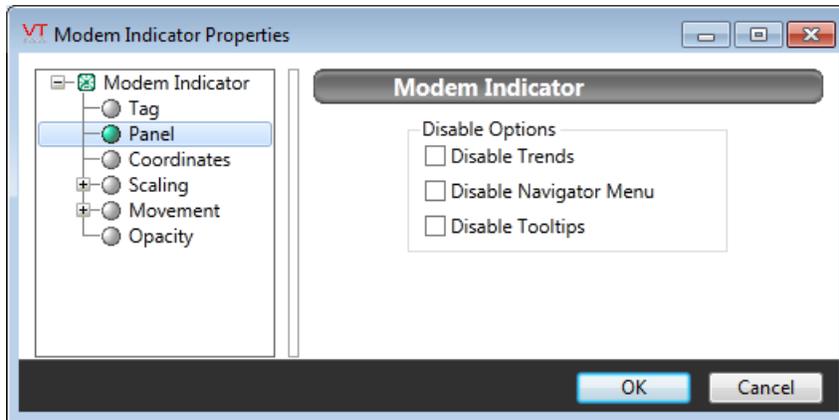
Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Communications\Modems\
Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Momentary Button Widget

Used by Analog Input, Analog Status, Digital Input, Digital Status, Pump Status, Analog Output, Analog Control, Digital Output, Digital Control, MultiWrite, Pump, Selector Switch, Trigger.

* Does not use the Stye Settings tag.

The Momentary Button widget is used to draw a button that will send one value when pushed and the same or a different value when released. Both the value to write when pushed, and the value to write when released, must be provided before the button will be enabled.

This object can be displayed as a button, a hotbox, or a button surrounded by a hotbox.

Only the button will have a label (whether or not surrounded by a hotbox).

If the label is to include an ampersand (&) and you are not using the legacy style, then the & character will need to be doubled in order to appear.

The properties dialog for the Momentary Button:
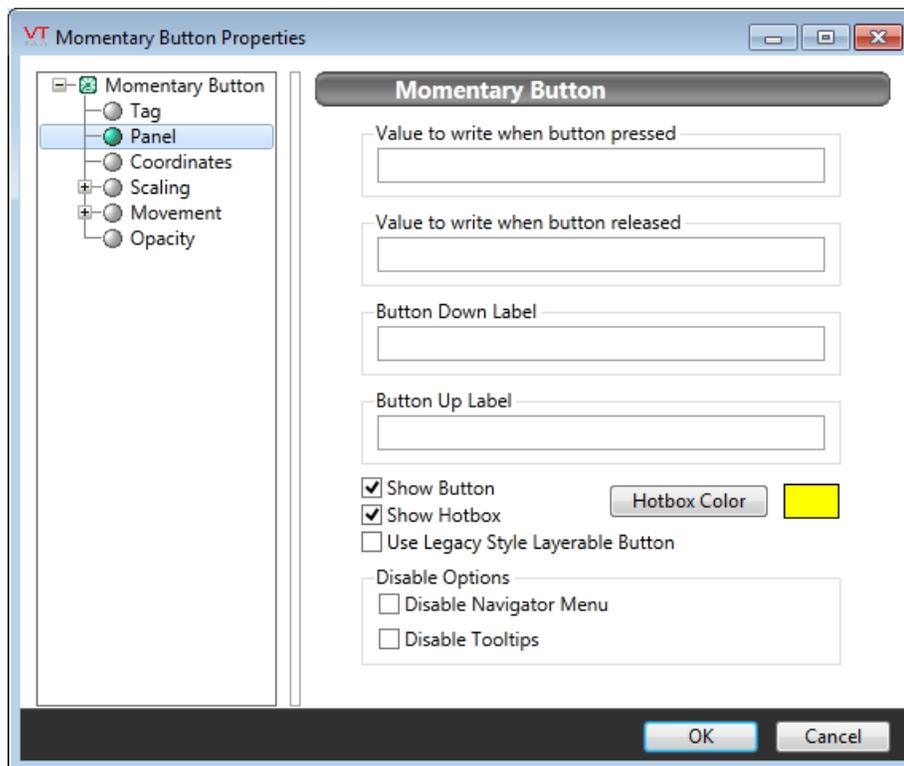
Value to write when button pressed

Set the value that will be written when the completed button is

clicked. If the associated tag is an analog output, this may be any value that is legal for that tag. Otherwise, this value should match one of the configured states for the associated digital or selector switch.
If this value is not set, the button will not work.

Value to write when button released

Set the value that will be written at the moment when the operator releases the button.

Button Down Label and Button Up Label

Provide the labels that will be displayed on the button. A different label will be displayed while the button is being clicked than will otherwise be shown. This can provide useful feedback to the operator who can be assured that the button has indeed been clicked and is presumably doing what it is supposed to.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Hotbox Color

Opens the Select Color Dialog from which you can select a color for the hotbox object.
The Hotbox Color button will be disabled in the event that the Show Hotbox check box is not selected.
Show Button and Show Hotbox
Use these options to control whether the hotbox, the button, both or neither are displayed.
If Show Button is not selected the label fields will be disabled.
If Show Hotbox is not selected, the Hotbox Color button will be disabled.

Use Legacy Style Button

If checked, the button will be shown using an older (resizable) button style. The updated button will provide a more modern appearance to your pages but, it can only be resized hori-zontally.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Multi-Color Widget

Used by: Allen Bradley driver, Alarm Status, Analog Input, Analog Status, Analog Output, Analog Control, Calculation, Counter, Data Flow RTU, CalAmp Diagnostic driver, DDE Client, Deadband Control, DNP3 driver,   Driver Multiplexer, Function, History Statistics, MDS driver, Modem,   Modbus Compatible Device , Omron driver, OPC Client, Polling Driver, Rate of Change, Script,   Selector Switch, Serial Port, Siemens S7 driver, SNMP driver, SMS Appliance, SQL Logger, Total-izer.
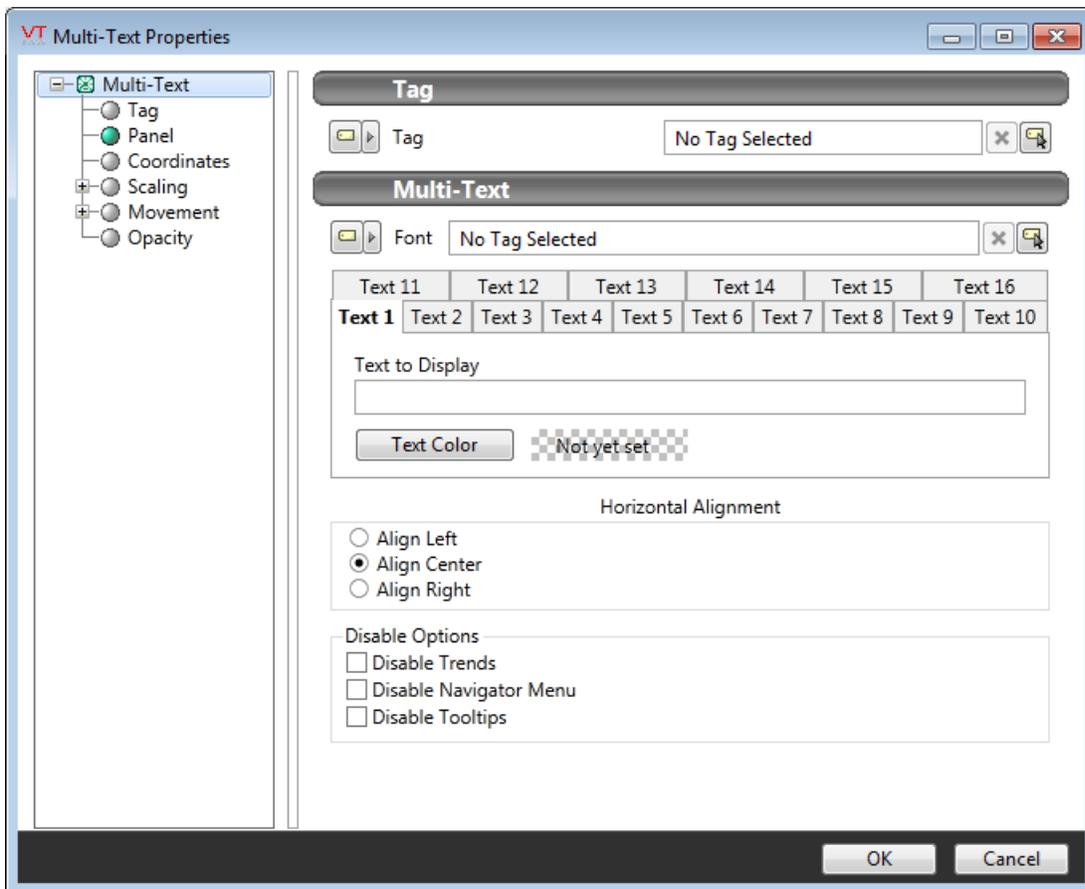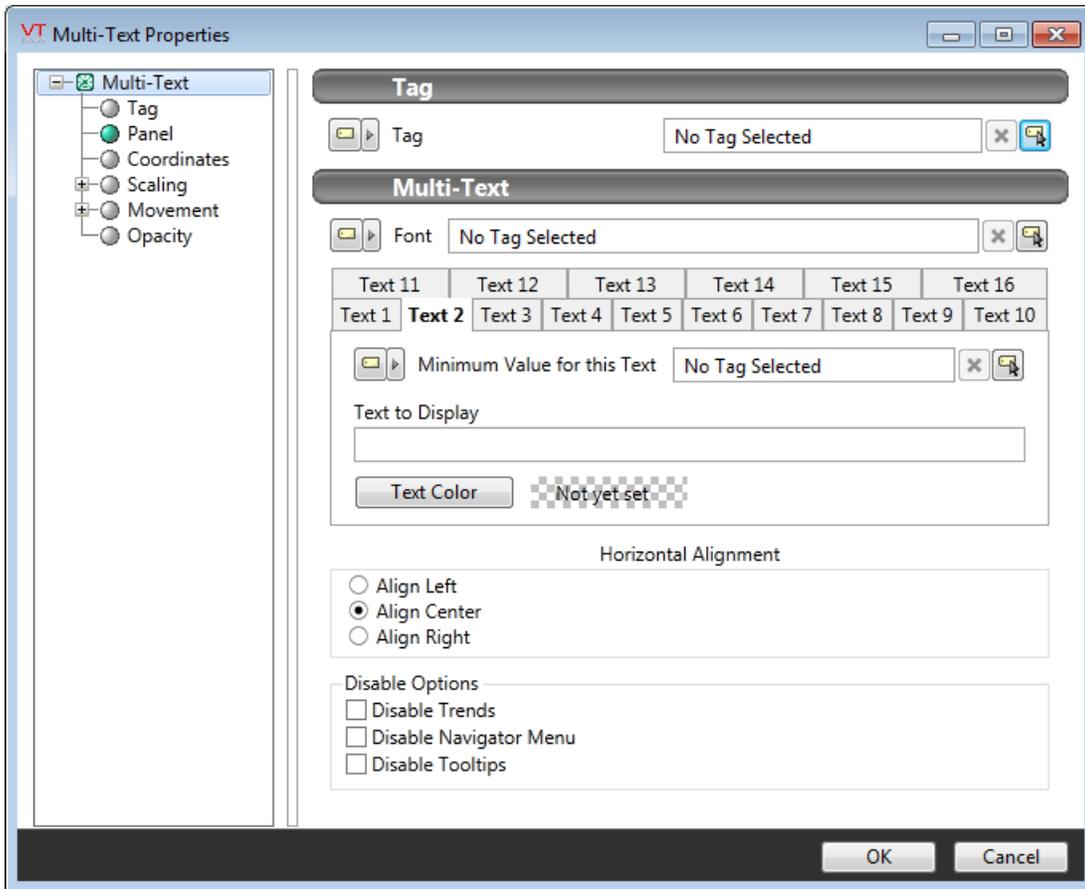* Does not use the Stye Settings tag.
The Multi-color tag widget is used to display a variety of rectangular color swatches in response to changing conditions within the application. You can create up to 16 colors using the Color 1 through Color 16 tabs. The Color 1 swatch is the default to be shown when no user-defined con-ditions have been met for any of the other colors.

In the following example, a Multi-color widget has been used to create a background for a Linear Gauge widget. As the associated tag's value increases, the multi-color serves as a visual warning.
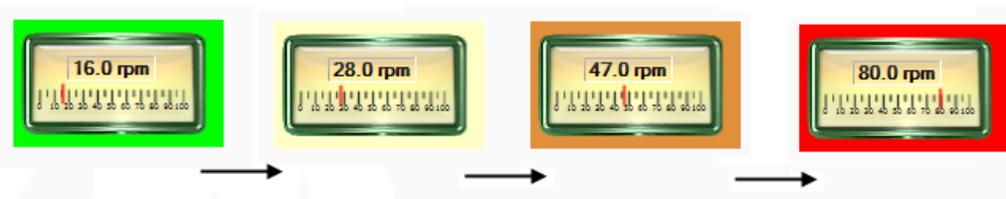
The Multi-color is configured by creating a series of increasing trigger points and assigning a color to be displayed as each point is reached. The trigger points can be given constant values, or can take their value from either a tag or parameter.

The initial color, color 0, does not have a trigger point since it will be the color in effect until the tag's value reaches the first assigned value. Two versions of the properties dialog for this widget are shown:
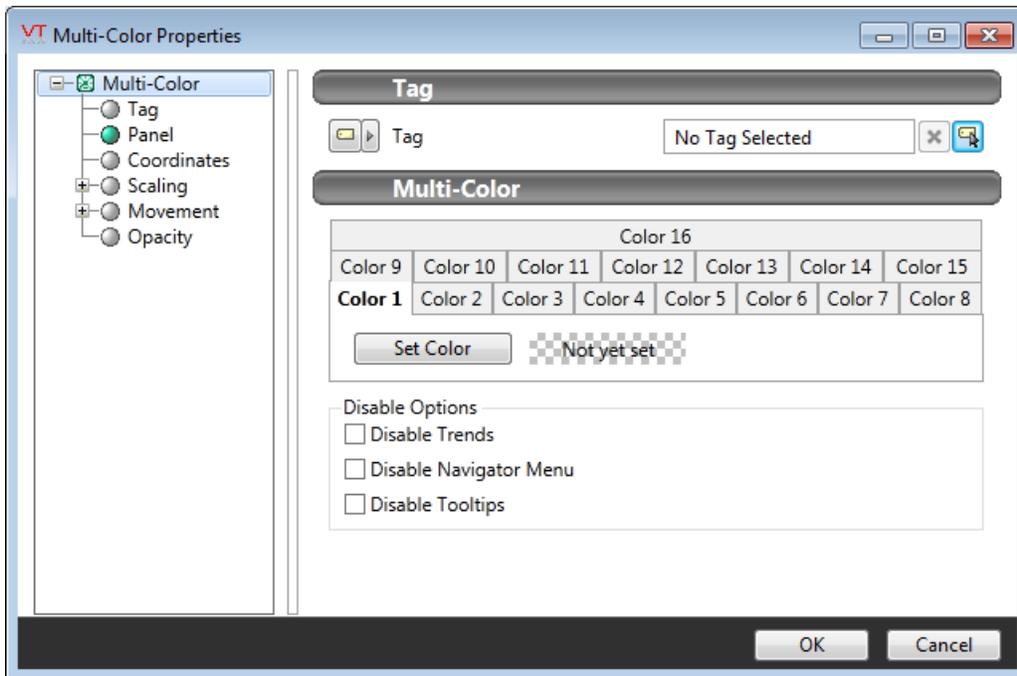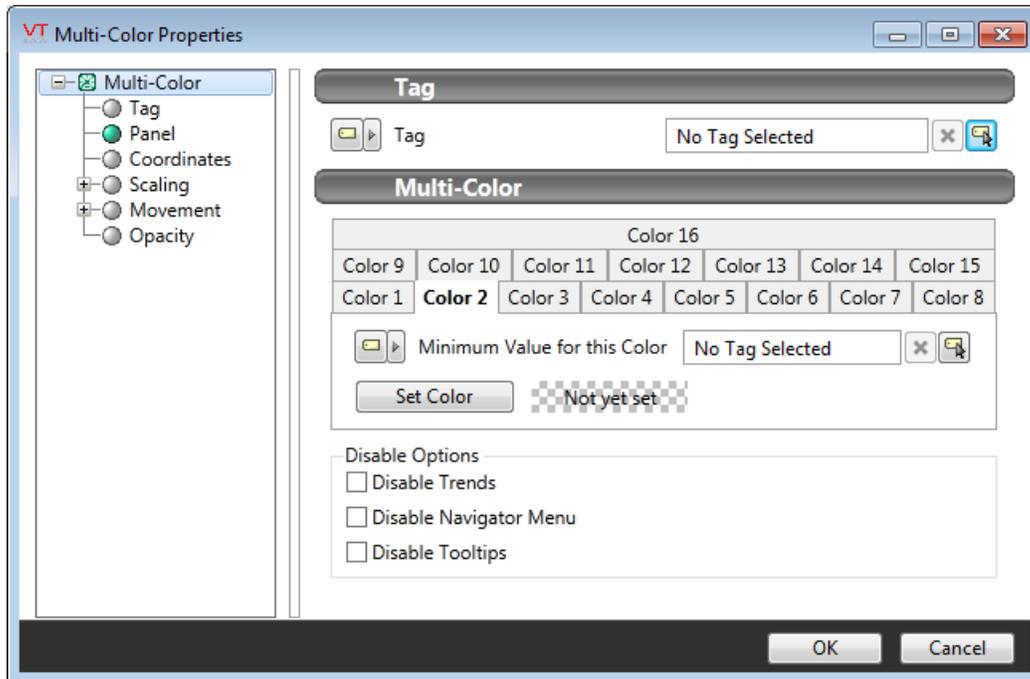
What happens if the set point for Color 3 is reached before that of Color 2?

If using constant values, you cannot assign a lower trigger point for a higher color number. This situation can occur if the trigger point for each color is provided by different tags.

The multi-color widget will display each defined color in order from Color 1 through to Color 16. The swatch for Color 1 will be displayed until the comparison condition for Color 2 has been reached. Until the set point for Color 2 has been reached, it does not matter whether any other comparison condition has been exceeded.

After the comparison for Color 2 has been passed, the multi-color will go on to check the comparison for Color 3. If it has already been passed, then the swatch for Color 2 will be skipped in favor of the swatch for Color 3.

The same comparison order is used whether the Tag's value is rising or falling.

Color 1... Color 16 tabs

Select each tab in turn, up to the number of colors you wish to

display. Each tab after Color 1 is used to set a trigger point using a constant, tag or parameter, and a color to be displayed when that point is passed. Each successive tab used must have a greater value than the previous.

Set Color

Opens the Select Color Dialog from which you can select a color to be displayed when the matching trigger point has been reached.

Minimum Value for this Color

Sets the trigger point, above which the set color will be displayed (until the next trigger point is reached). The value for this point may be provide by any of a user-defined constant, an expression another tag or a parameter.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators\
Basic Components\Color Indicators\

Related Information:

...Choices for Operator Interaction

...Multi-Text Widget – Display a new message, in a unique color, for up to 16 pre-set values.

Multi-Line Text Widget

Used by: None
* Does not use the Stye Settings tag.
The Multi-Line widget is not linked to any tag. It can be found within the Widgets palette, within the Tools folder >> Standard sub-folder.

Operational Instructions:
1) Ensure that readings are within spec. before operating valve.
2) Use the slider (A) to set the valve position.
3) Click button (B) to open the valve.

The properties dialog for the Multi-Line Text widget:
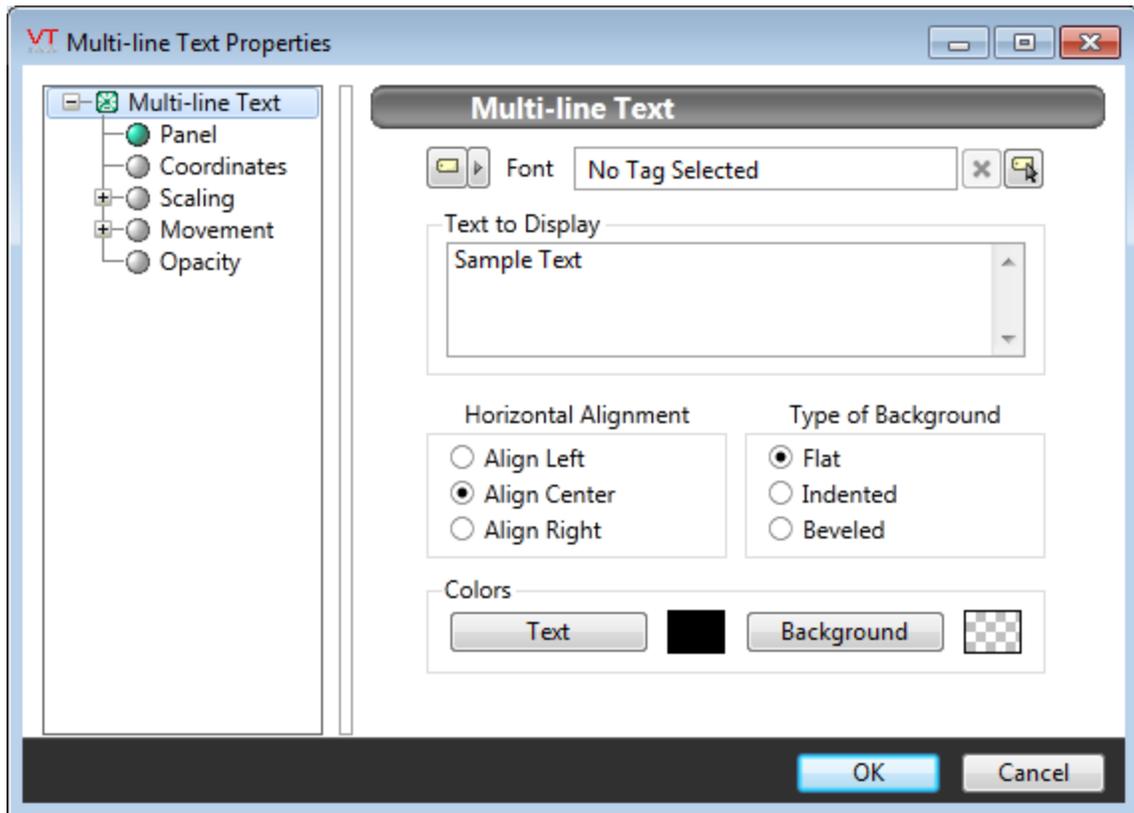
Font

Choose the Font tag that holds the font selection and other style settings for this text.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Text to Display

The editing dialog for the multi-line text.
Note that while text will wrap to fit the visible space, only
those line breaks that you type are added to the text.

Horizontal Alignment

Choose between Left, Center and Right alignment.

Type of Border

The two border types simulate an area that is inset in the page
or one that is a raised panel on top of the page.

Colors

Sets the color of the text and the background color, against
which the text will be displayed.

Widget Folders:
Tools\Standard Library\
Text\

Related Tasks:

...Draw Text

Multi-Text Widget

Used by: Allen Bradley driver, Alarm Status, Analog Input, Analog
Status, Analog Output, Analog Control, Calculation, Counter, Data
Flow RTU, CalAmp Diagnostic driver, DDE Client, Deadband Control,
DNP3 driver,   Driver Multiplexer, Function, History Statistics, MDS
driver, Modem,   Modbus Compatible Device , Omron driver, OPC
Client, Polling Driver, Rate of Change, Script,   Selector Switch, Serial
Port, Siemens S7 driver, SMS Appliance, SNMP driver, SQL Logger, Total-
izer.
* Does not use the Stye Settings tag.

The Multi-text tag widget displays a variety of messages, each in its own color, in response to changing conditions within the application.

You can create up to 16 messages using the Text 1 through Text 16 tabs. The Text 1 message is the default, to be shown when the defined conditions have not been met for any of the other messages.

The following example shows a multi-text display below a Linear Gauge widget. As the tag's value increases, the multi-text displays a different value to either inform or direct the operator, and uses a different text color to catch the operator's attention.

The Multi-text is configured by creating a series of increasing trigger points and assigning a color and message to be displayed as each point is reached. The trigger points can be given constant values, or can take their value from either a tag or parameter.

The initial message, text 0, does not have a trigger point since it will be the text displayed until the tag's value reaches the first assigned value. Two versions of the properties dialog for this widget are shown:

Multi-Text Properties

Multi-Text
- Tag
- Panel
- Coordinates
- Scaling
- Movement
- Opacity

Tag

Tag   No Tag Selected

Multi-Text

Font   No Tag Selected

| Text 11 | Text 12 | Text 13 | Text 14 | Text 15 | Text 16 |

| Text 1 | Text 2 | Text 3 | Text 4 | Text 5 | Text 6 | Text 7 | Text 8 | Text 9 | Text 10 |

Text to Display

Text Color   Not yet set

Horizontal Alignment

- ○ Align Left
- ● Align Center
- ○ Align Right

Disable Options
- ☐ Disable Trends
- ☐ Disable Navigator Menu
- ☐ Disable Tooltips

OK   Cancel

What happens if the set point for Text 3 is reached before that of Text 2?
If using constant values, you cannot assign a lower trigger point for a higher text number. This situation can occur if the trigger point for each text message is provided by different tags.

The multi-text widget will display each defined message in order from Text 1 through to Text 16. The message and color for Text 1 will be displayed until the comparison condition for Text 2 has been reached. Until the set point for Text 2 has been reached, it does not matter whether any other comparison condition has been exceeded.

After the comparison for Text 2 has been passed, the multi-color will go on to check the comparison for Text 3. If it has already been passed, then the message for Text 2 will be skipped in favor of the message for Text 3.

The same comparison order is used whether the Tag's value is rising or falling.

Font Name

Select one of the font tags (or parameter linked to a font tag) that you wish to use to display the text. The display will use the default system font if you do not select a font tag.
The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Text 1… Text 16 tabs

Select each tab in turn, up to the number of messages you wish to display. Each tab after Text 1 sets a trigger point using either a constant, tag or parameter, and message + color combination to be displayed when that point is passed. Each consecutive tab used must set a greater trigger point than the previous.

Minimum Value for this Text

Sets the trigger point, above which the set color will be displayed (until the next trigger point is reached). The value for this point may be provide by any of a user-defined constant, an expression another tag or a parameter.

Text to Display

Enter the text that you want to have displayed when the tag's value goes above this trigger point.

Set Color

Opens the Select Color Dialog from which you can select a color to be displayed when the matching trigger point has been reached.

Horizontal Alignment

Configure the text to be left, center or right-aligned within the widget's bounding box.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\
Text\

Related Information:

...Choices for Operator Interaction

...Multi-Color Widget – Display up to 16, solid colors for increasing values.

Multi-Write Button Widget

Used by: Multi-Write tag.
* Does not use the Stye Settings tag.
The Multi-Write button is used to create a button, with which operators can trigger the operation of a Multi-Write tag.

Set Value
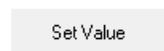
The properties dialog for the Multi-Write button will appear as shown:
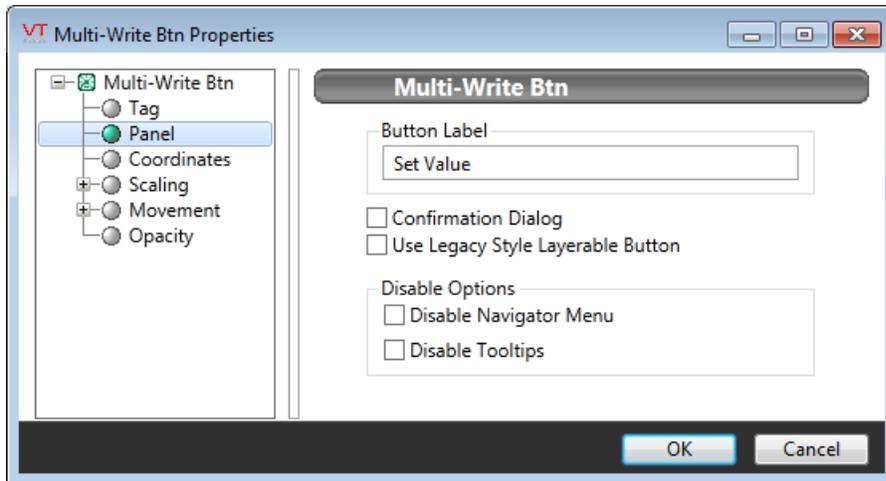
Button Label

Provide the text for the button label in this space.
If the label is to include an ampersand (&), then the & character
will need to be doubled in order to appear.

Confirmation Dialog

If checked, the operator is required to confirm the control
action before the configured values are written.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Multi-Variable\
Analytics\Communications\Output Controls\

Related Information:

...Choices for Operator Interaction

... Multi-Write Tags

Multi-Write Hotbox Widget

Used by: Multi-Write tag

* Does not use the Stye Settings tag.

The Multi-Write hotbox is used to create an area that operators can click within to trigger the operation of a Multi-Write tag.

The properties dialog for the Multi-Write Hotbox:
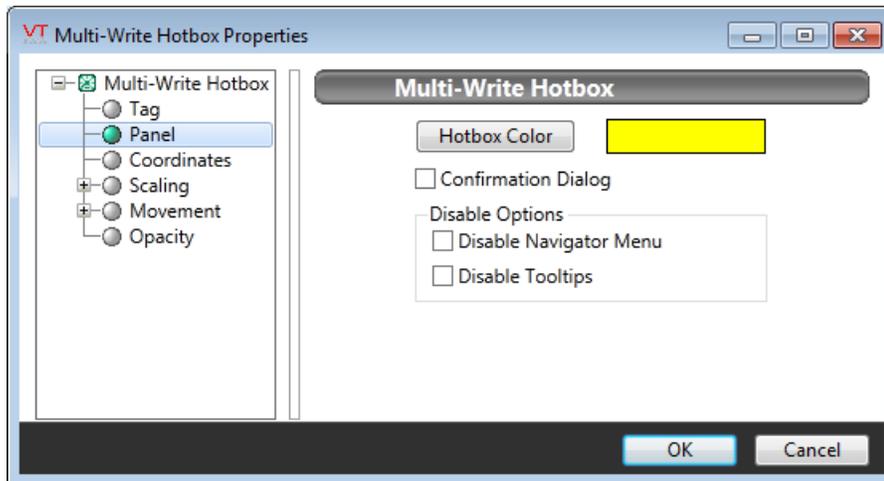
Hotbox Color

Select the color that will be used to show the hotbox area. Yellow is standard.

Confirmation Dialog

If checked, the operator is required to confirm the control action before the configured values are written.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Multi-Variable\
Analytics\Communications\Output Controls\

Related Information:

...Choices for Operator Interaction

...Multi-Write Tags

Nautical Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Nautical Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.
Unlike other meters, Nautical Gauge provides the option of adding a label to the display
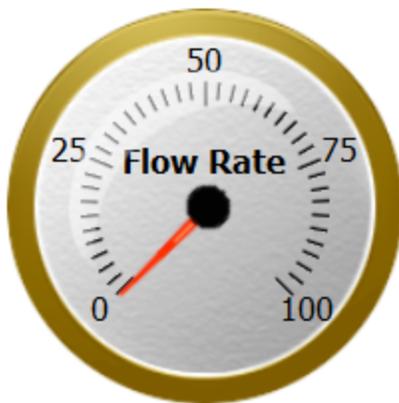
The properties dialog for the Nautical Gauge widget:
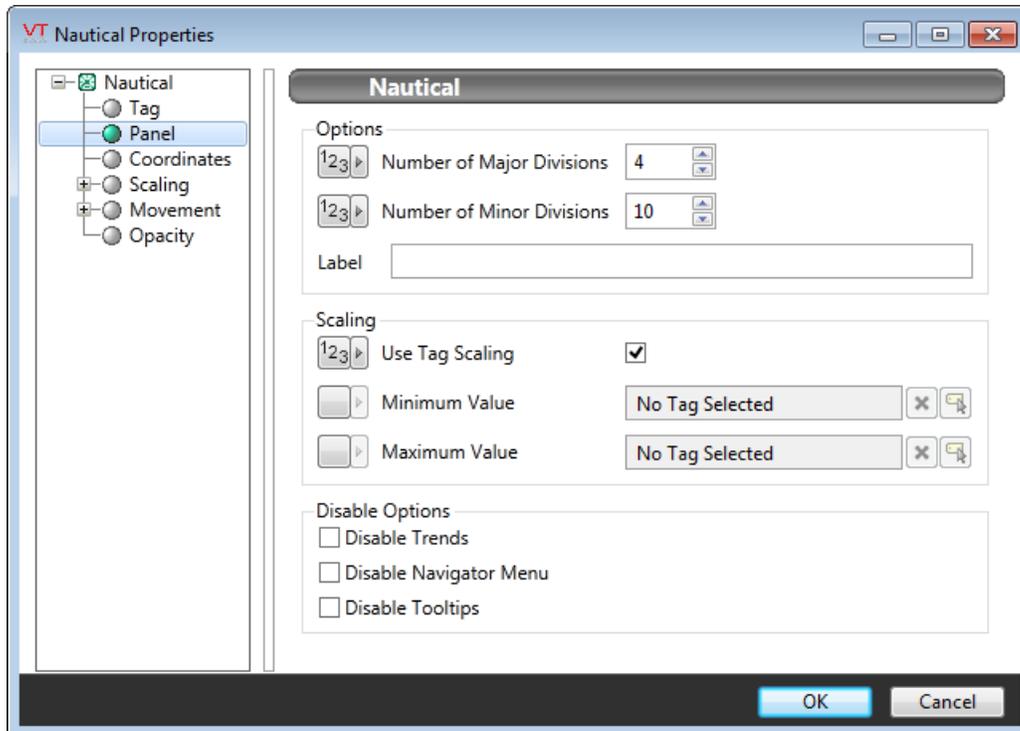
Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Label

Text to display on the meter. Due to font size limitations, a maximum of nine characters including spaces will be displayed. The label is centered, therefore clipping will occur on both sides if you provide more characters than will fit.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.
You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Network Link Widget

Used with: Network Status
* Does not use the Stye Settings tag.
The Network Link tag widget is used to draw a line indicating the value of the associated Network Status tag. It can tell an operator whether the network link is active, broken or if its current status is unknown.
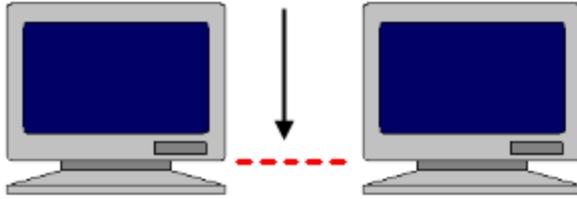
Use the properties dialog for the network link widget to select the line color, width and style for each status, as well as the line's orientation on the page.
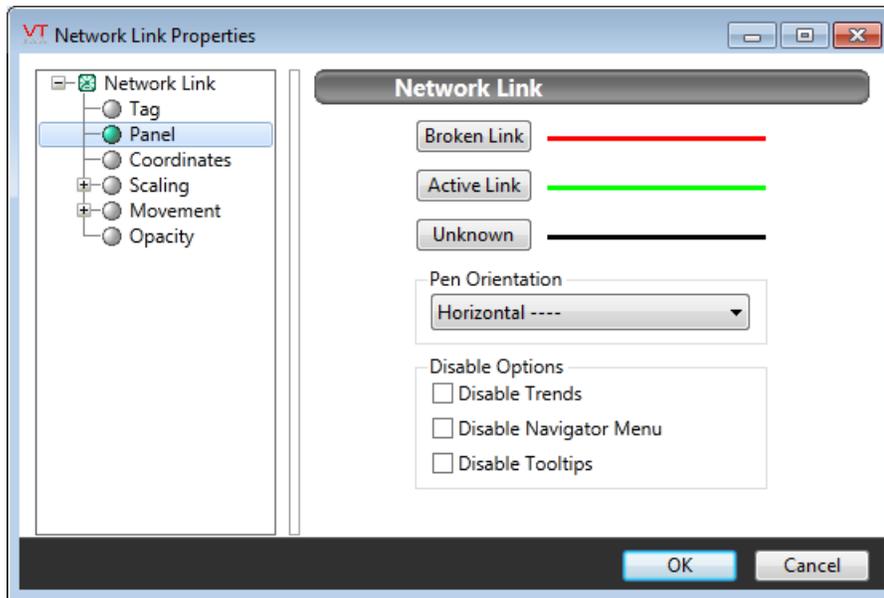
Broken Link, Active Link and Unknown

These three buttons all open the Select Pen dialog. Select the color, line style and pen width that you want to use to represent the Network Link line object for each matching state.

Pen Orientation

Select one of four orientations for the completed Network Link line object (horizontal, vertical, left 45 degrees, or right 45 degrees).
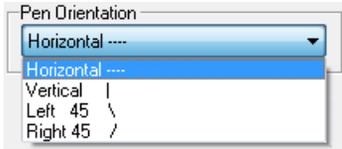
Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Communications\Connection Status\

Analytics\Diagnostics\

Related Information:

...Choices for Operator Interaction

Note List Widget

Used with: Notebook.
* Does not use the Stye Settings tag.
Creates a display area on a page for notes that have been added to the associated Notebook tag. A scroll bar will be added to the right edge of the widget when there are more notes than will fit in the given area.
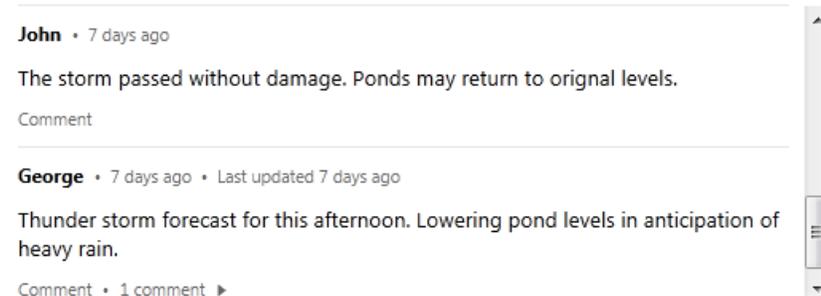
All the information in each note will be displayed, including the date and time when the note was recorded, the name of the operator, and the text of the note. New notes are added at the top of the list. Comments may be expanded or hidden within the list.
The display can be configured for color (text and background), font, etc.
The properties dialog for the Note List widget:
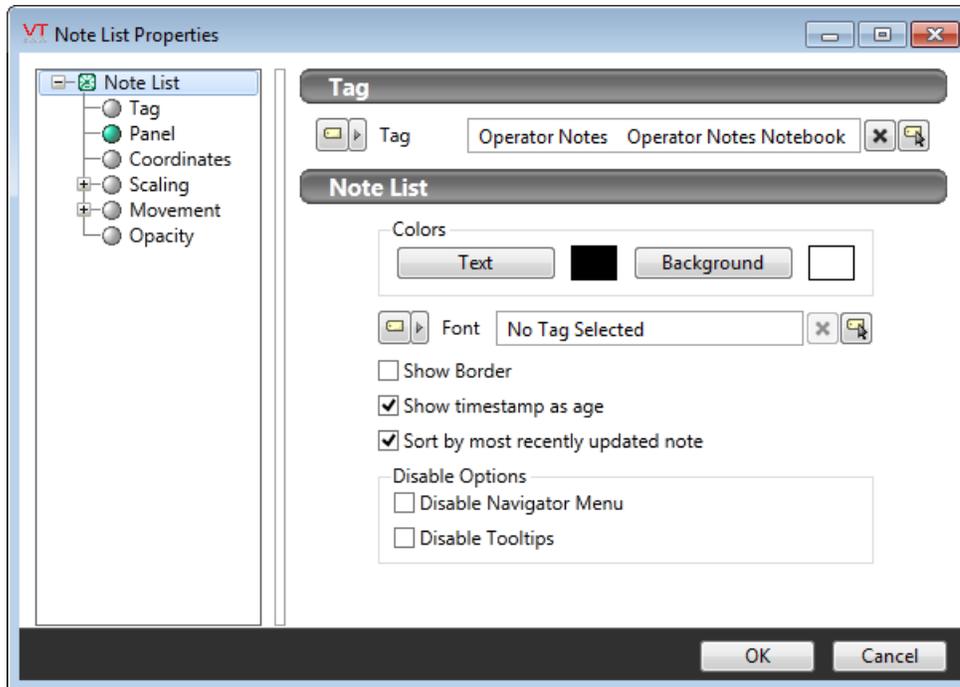
Colors

Both the text color and the background color may be configured. Clicking either button will cause the Select Color dialog to open.

Font

Use the tag browser button to the right of this field to select the font with which the note will be displayed. If the current page contains Font tag parameters, then you may select from one of these.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Show border

Choose whether to add a border around the note display. Borders can be helpful to draw attention to an area and to delineate the note display from the rest of the page.

Show timestamp as age

By default, each entry in the list will show the date and time when it was created. You can choose to display the note's age instead.

Sort by most recently updated notes.

Controls whether the date of comments added to a note are considered when calculating the sorting order. If this option is not checked, sorting will be done only by the date of the original note, regardless of later comments.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Specialty\Notebook

Numeric Entry Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control, Counter, Selector Switch, Totalizer.
* Does not use the Stye Settings tag.

Note: Do not link this widget to an Analog Input or Analog Status unless those tags are configured to write as well as read.

The Numeric Entry widget is used to create an entry field into which operators may enter numeric values to output to the associated PLC or RTU.
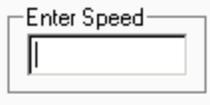
Attempts to enter text in this field will produce the following error message:
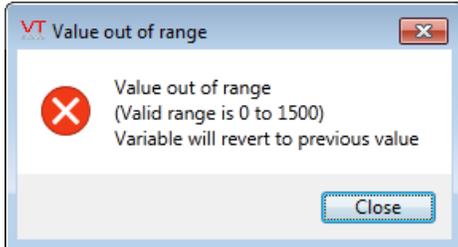
This widget is most commonly used with analog outputs and analog controls. The minimum and maximum permitted values match the tag's configured Scaled Process Data Min and Scaled Process Data Max properties For a totalizer or counter tag, it can be used to allow the operator to set the value. In the case of a totalizer that is updating every second, this may prove to be impractical.

For a selector switch, this object provides a way to numerically set the switch position to 0, 1 or 2.

The properties dialog for the Numeric Entry widget is as shown:
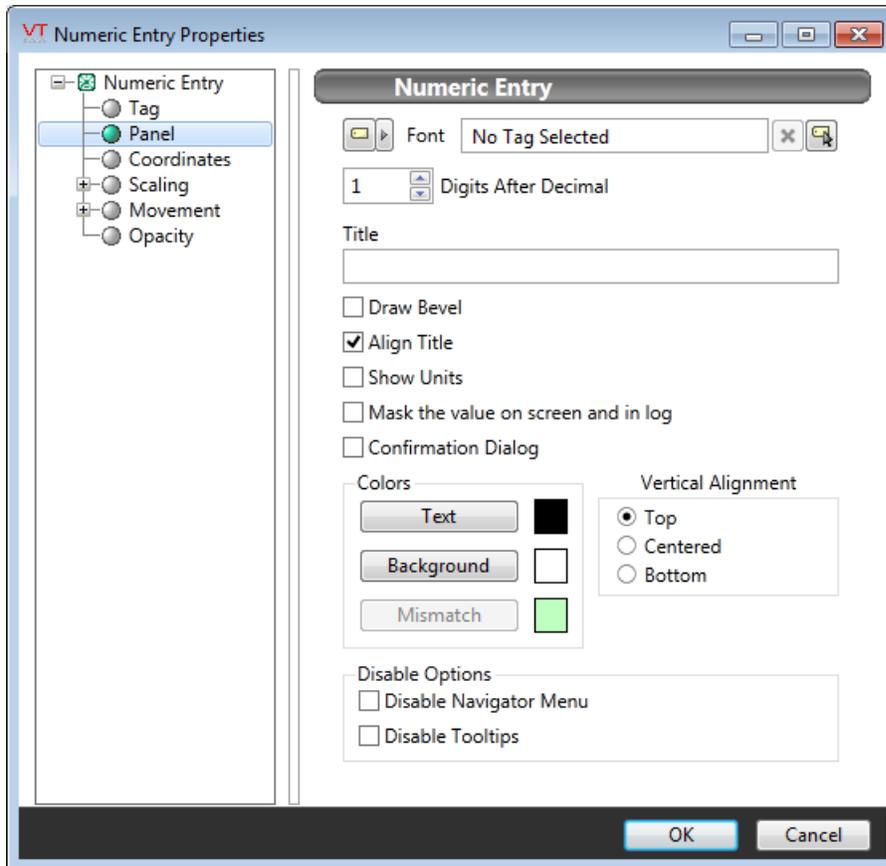
Font

Controls the display characteristics of the data entry field, but not the label. If you set individual characteristics using the Format ribbon, this will be an expression, combining those characteristics. Otherwise, select a font tag.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Digits After Decimal

Controls how many digits will be displayed after the decimal point. Has no effect on the number that the operator may enter.

Title

Provide text that will tell the operator what to use this Numeric Entry field for.

Draw Bevel

Select whether or not the completed object should be drawn with a beveled border. If the Draw Bevel check box is selected, the completed object will be drawn with a border surrounding it.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom as shown. You can save time by putting the object where you would like on the screen and not making the drawing area larger than the object.

| Top
With Align Title | Top
Without Align Title | Centered | Bottom |
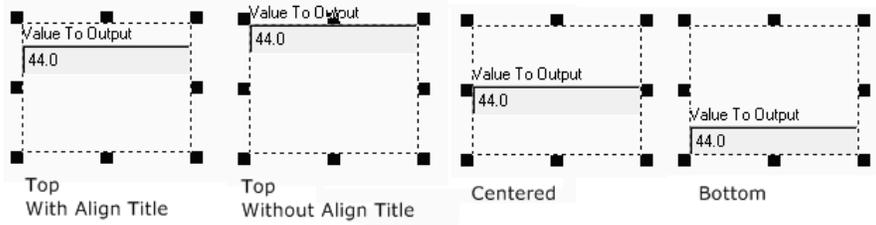
Mask Value

When checked, the value will be masked by asterisks rather than being displayed on the screen or in the event log. This option will most often be selected when the Numeric Entry is being used to write a password to a hardware device, thus pre-venting other users from discovering the password.
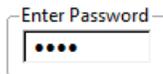
Confirmation Dialog

When checked, the operator will be prompted to confirm the control action.

Text Color

Select the color to be used for the input text. Does not apply to the label.

Background Color

Select the color to be used for the background of the input field. Does not apply to the label.

Mismatch Color

Used only when the associated tag is a Selector Switch, or an Analog Input or Analog Status type, configured to write values. When the last value written does not match the value being

read, the widget will display the configured mismatch color.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analog Controls\

Related Information:

...Choices for Operator Interaction

Numeric Value Widget

Used by: All driver tags, All analog tags, Alarm Status, Calculation,
Counter, Deadband Control, Function, History Statistics, Modem, Rate of
Change, Script, Selector Switch, Serial Port, Totalizer, SQL Logger.
* Does not use the Stye Settings tag.
The Numeric Value widget is used to display the current value of any of a
tag as a number. Units (if any) will be displayed, depending on how you
configure the object.
Should the associated tag have a text value, it will not be displayed. If
your tag is likely to have a text value, then the Draw Text Widget will be a
better choice.
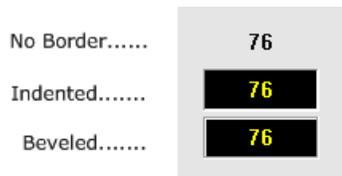
The properties dialog for the Numeric Value object:

Font

Select one of the font tags (or parameter linked to a font tag) that you wish to use to display the text. The display will use the default system font if you do not select a font tag.
The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Display Format

May be any of: Decimal, Hex, Octal, or Binary. If a display format other than Decimal is chosen, the Integer Size drop-down will be enabled as will the option, Display Leading Zeros".

Integer Size

Applies only to Binary, Octal and Hex display formats. May be any of: Byte, Word (two bytes), Dword (double word or 4 bytes) or Qword (quadruple word or 8 bytes)

Digits After Decimal

Applies only to the Decimal display format. Controls how many decimal places will be shown in the display. Values will be rounded to the number set here.

Show Units

Controls whether or not the tag's engineering units (if set) will be shown as part of the display.

Display Leading Zeros

Applies only to Binary, Octal and Hex display formats. If not checked, leading zeros for these values will be truncated. Leading zeros are not displayed for the Decimal display format.

Horizontal Alignment

Sets the position of the text within the display area. You may select left-aligned, right-aligned or centered. To align the display area itself with other elements on the page, see: Alignment and Positioning Tools.

Type of Background

Examples of the three border types are shown at the beginning of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the color of the text.
The background color against which the text will be displayed can also be defined.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\
Basic Components\Text\

Open HDV Button

When drawn on a page, this button provides a link to open the Historical Data Viewer as a pop-up with predefined settings for pen group and time duration.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.
The configuration dialog for the button:

The Button Label may be changed to display any message you like. Since more than one of these buttons may be added to a page, each displaying the HDV with a different pen group or time duration, the label should indicate what will be shown when the operator presses this particular button.

The Pen Group Selector drop-down list will display all pen groups previously configured in the Historical Data Viewer. The groups must exist before configuring this button – there is no mechanism for creating a new pen group in this dialog.

The Duration Selector is a drop-down list of all the standard time durations available in the HDV, ranging from 1 second to 5 years. When the HDV opens, it will display the selected time duration ending with the current time (i.e. a live data display).

Note: If no pen group is configured, clicking on an instance of this button on a page will simply open the HDV.

This tool may also be drawn as a hotbox rather than as a button.

Widget Folders:
Analytics\Trends & Historical Data\
Tools\Standard Library\

Related Information:

...Save a Pen Group

...Open HDV Hotbox

...Draw HDV Widget

...Plot Data Widget

Open HDV Hotbox

Found in the Standard Library folder of the Widgets palette. When drawn on a page, this hotbox provides a link to open the Historical Data Viewer. This tool differs from the Historical Data Viewer entry in the menu in that it can be pre-configured to open a particular pen group and display a pre-set time duration.
The properties dialog for the hotbox:
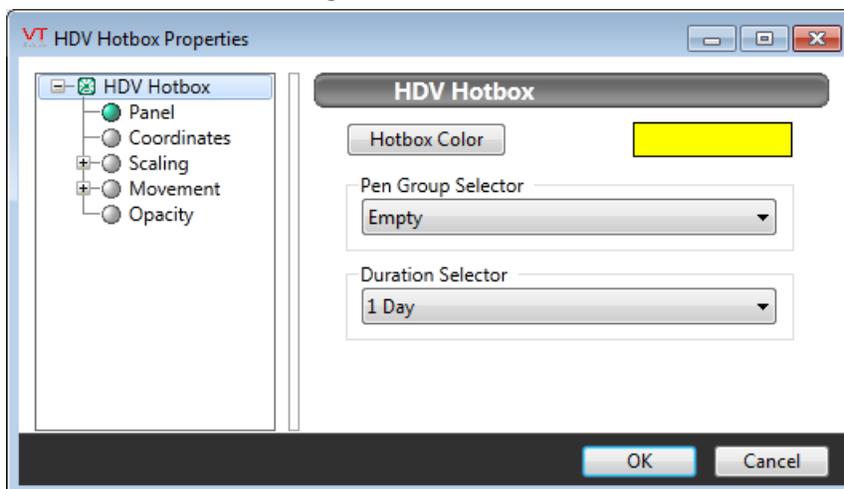
The Hotbox Color may be changed to display any color you like. Yellow is the default and is most often recognized as indicating a hotbox link. Since more than one of these hotboxes may be added to a page, each opening the HDV with a different pen group or time duration, the hotbox should be placed around a label that indicates what will be shown when the operator clicks on this particular link.
The Pen Group Selector drop-down list will display all pen groups previously configured in the Historical Data Viewer. The groups must exist before configuring this hotbox – there is no mechanism for creating a new pen group in this dialog.
The Duration Selector is a drop-down list of all the standard time durations available in the HDV, ranging from 1 second to 5 years. When the

HDV opens, it will display the selected time duration ending with the current time (i.e. a live data display).

Note: If no pen group is configured, clicking on an instance of this hotbox on a page will simply open the HDV.

This tool may also be drawn as a button rather than as a hotbox.

Widget Folders:
Analytics\Trends & Historical Data\
Tools\Standard Library\

Related Information:

...Save a Pen Group

...Open HDV Button

...Draw HDV Widget

...Plot Data Widget

Page Button Widget

Standard Library. Not linked to any tag.
* Does not use the Stye Settings tag.
The Page Button widget is a navigational tool, used to create a connection from one page to another.
Buttons are always visible, and can be configured with their own label so that operators will know what to expect when the click.

Open Control Page

The properties dialog for the Page button widget:

Button Label

The label should always tell the operator what to expect to happen when the click a button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Page Name

Select the page to be opened when the operator clicks the button.

Parameters

Visible only if the selected page has parameters. If so, it is important to select the tags and values that will be used for those parameters when the operator opens the page using this particular button. Failing to do so will result in the operator being prompted for parameter values with every return visit.

Whether the page opens as a pop-up window or full screen, depends on the configuration of the page, and not on any configuration of the button.

Widget Folders:

Page Navigation\
Tools\Standard Library\

Related Information:

...Define Navigation Between Pages

Page Close Button Widget

Standard Library. Not linked to any tag.
* Does not use the Stye Settings tag.
The Page Close Button widget is a navigational tool, used to allow operators to close pop-up pages. Some operators may find that a Close button is more convenient to use than the "x" in the upper right corner of the window.
Page Close buttons do not operate on full-screen pages. Operators may close only pop-up (windowed) pages.

The properties dialog for the Page button widget:

Button Label

The label should always tell the operator what to expect to happen when the click a button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tip Label

An optional field that may be used to provide more information to the operator when the cursor is over the button.

Widget Folders:
Page Navigation\
Tools\Standard Library\

Related Information:

...Define Navigation Between Pages

Page Hotbox Widget

Not linked to any tag.
* Does not use the Stye Settings tag.
The Page Hotbox widget is a navigational tool, used to create a connection from one page to another.
The hotbox is invisible unless the pointer is over it, at which time it turns yellow (or any other configured color). It is common to place a hotbox around a text label or an image so that operators will know where to find it, and what page to expect when they click.

The properties dialog for the Page Hotbox widget:

Hotbox Color

While most developers will leave the default color in place for the sake of consistency, you might decide to color-code the hotboxes for various groups of pages in your application. You might also want to ensure that Page Hotboxes use a different color than Set Value Hotboxes, which are analog control widgets.

Page Name

Select the page to be opened when the operator clicks the hotbox.

Parameters

Visible only if the selected page has parameters. If so, it is important to select the tags and values that will be used for those parameters when the operator opens the page using this particular hotbox. Failing to do so will result in the operator being prompted for parameter values with every return visit.

Whether the page opens as a pop-up window or full screen, depends on the configuration of the page, and not on any configuration of the hot-box.

Widget Folders:
Page Navigation\
Tools\Standard Library\

Related Information:

...Define Navigation Between Pages

Pie Widget

* Can be linked to a Stye Settings tag.
Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.

The Pie widget represents the associated tag's value as portion of a circle. The darker portion of the chart starts at the angle you specify, where vertical (North) is zero degrees. Increasing values are indicated by the darker portion sweeping in a clockwise direction. The colors are set by the LED (analog) values of a selected Style Settings tag, and values above the linked tag's alarm setpoint can be indicated.

The properties dialog for the Pie widget:
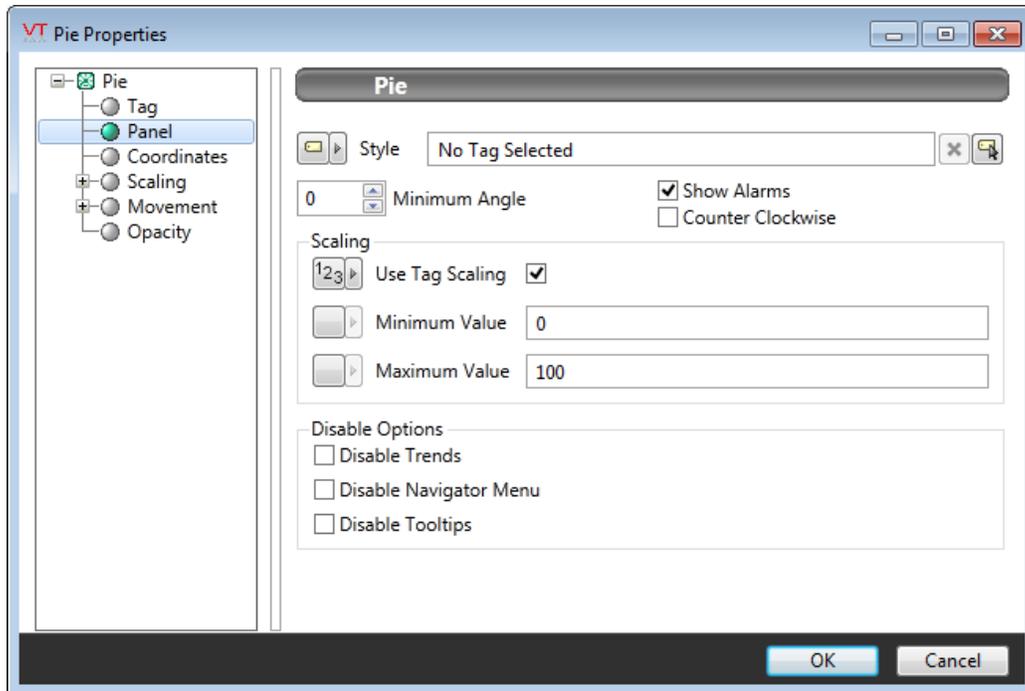
Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state. Style Settings Hierarchy.

Minimum Angle

Sets the starting point for the sweep of the pie chart. Defaults to zero, which is vertical. (The North or 12-o'clock position). The angle is measured in degrees, and increases in the clockwise direction.

Show Alarms

If selected, the pie chart will indicate when the associated tag is in an active alarm state by using the alarm indication settings as configured in the associated Style Settings tag. Only that portion of the sweep above or below the alarm setpoint will use the alarm indication.

Counter Clockwise

If selected, sections of the pie chart will sweep (increase) in a counter-clockwise direction rather than the standard clockwise.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
Choices for Operator Interaction
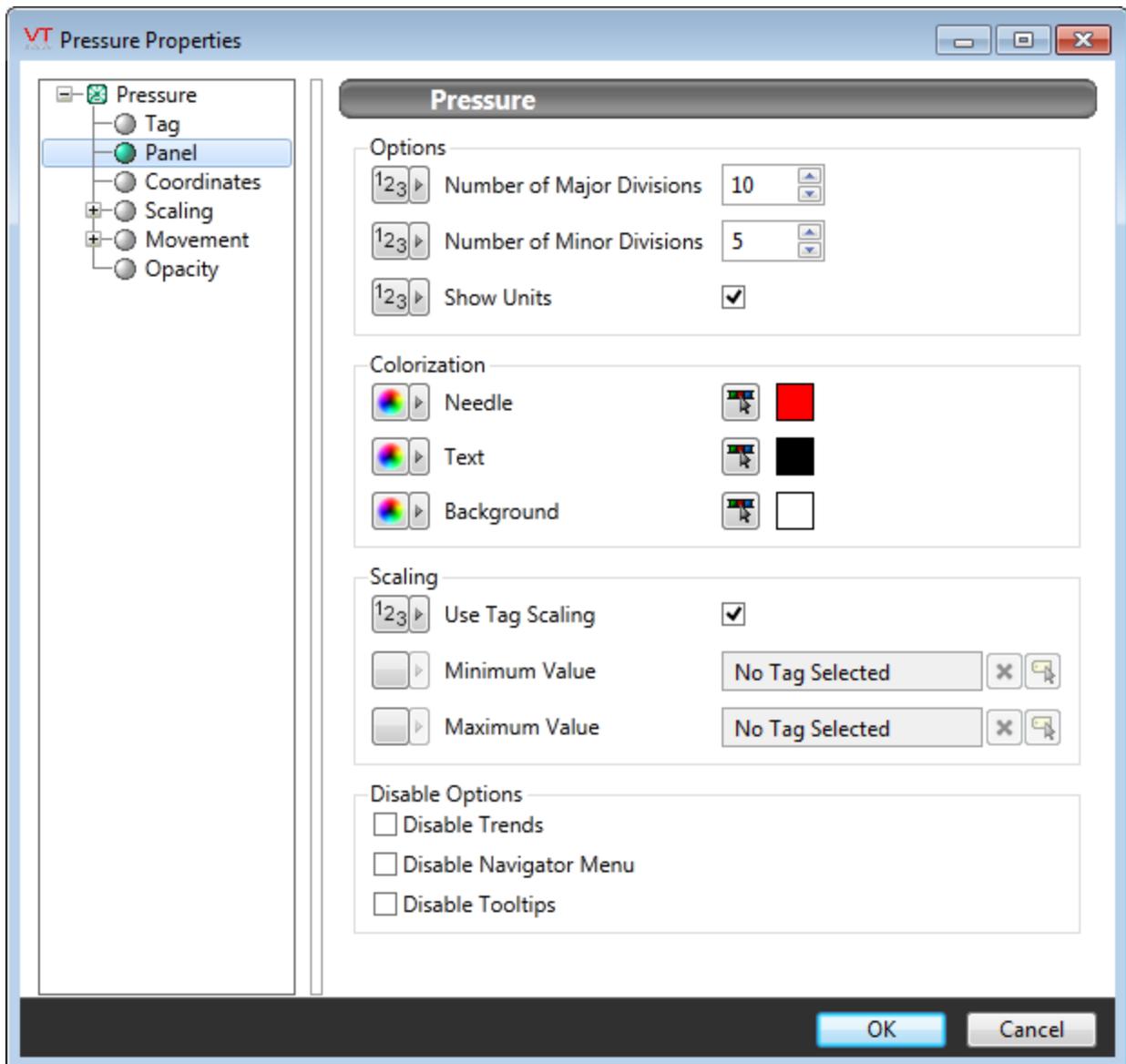
Plot Data Widget

Used by: All driver tags, All analog tags, Alarm Status, Calculation, Comm Link Sequencer, Counter, Deadband Control, All

digital tags, Function, History Statistics, Modem, Pump Status, Rate of Change, Script, Selector Switch, Serial Port, Totalizer, SQL Logger.
* Does not use the Stye Settings tag.
The Plot Data tag widget is used to plot the value of the tag being drawn on a graph similar to that displayed on the Historical Data Viewer page.
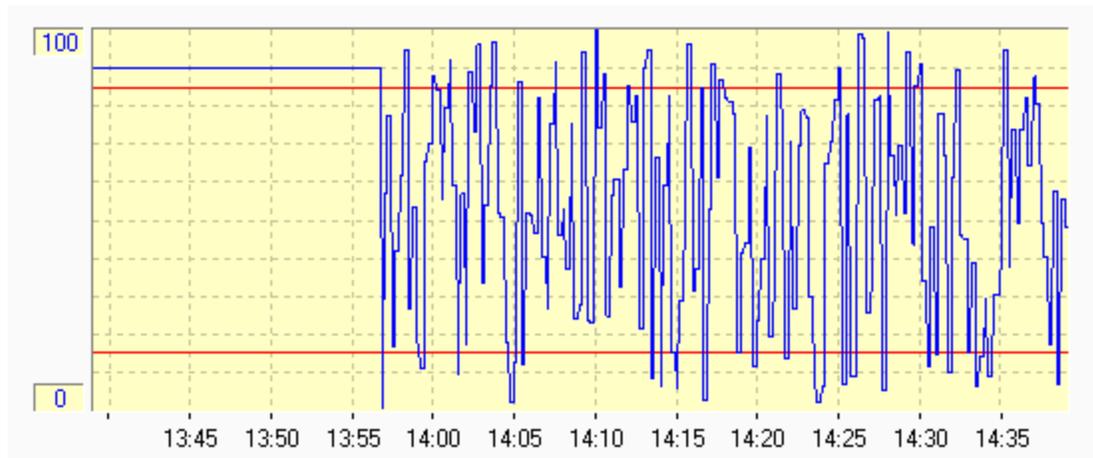
Note: Tags drawn using the Plot Data widget must be logged, either by an attached Historian or Logger.

The properties dialog for the Numeric Entry widget:
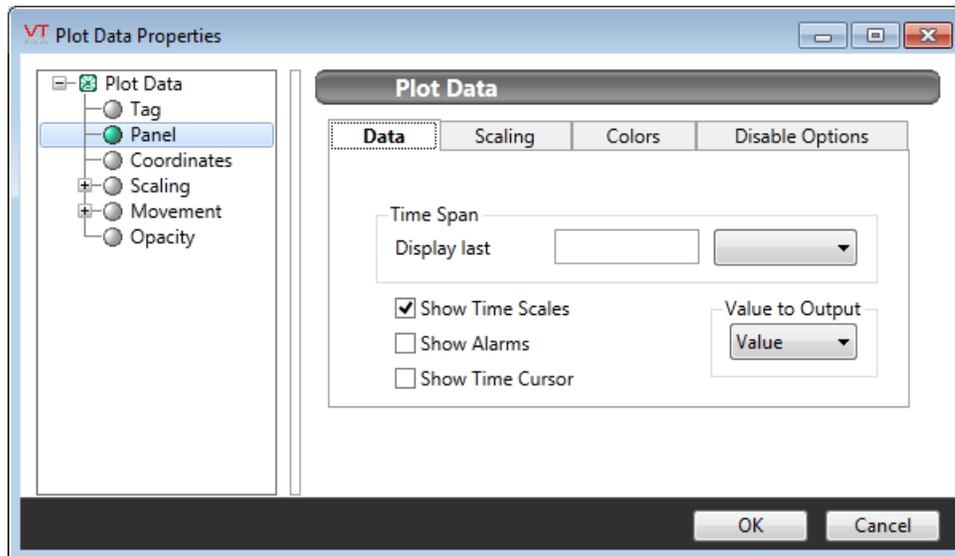
Time Span

The Display Last field and drop-down list are used to

configure the time span to be displayed in the completed Plot Data object.

You may select Minutes, Hours, or Days from the drop-down list, and enter a corresponding number in the field to its left.

Display Time Labels

Select this option if you wish to display incremental units of time along the bottom of the completed Plot Data object.

Show Alarms

Select this option if you wish to display the alarm setpoints (if any), that associated with the tag being drawn on the completed Plot Data object. Alarm setpoints will be drawn as horizontal lines at the values matching the setpoints. This check box will not be enabled if the tag has no associated alarms.

Value To Output

Select which value associated with the tag, should be plotted. For the majority of tags, there will only be one default option, "Value," indicating the value of the tag.
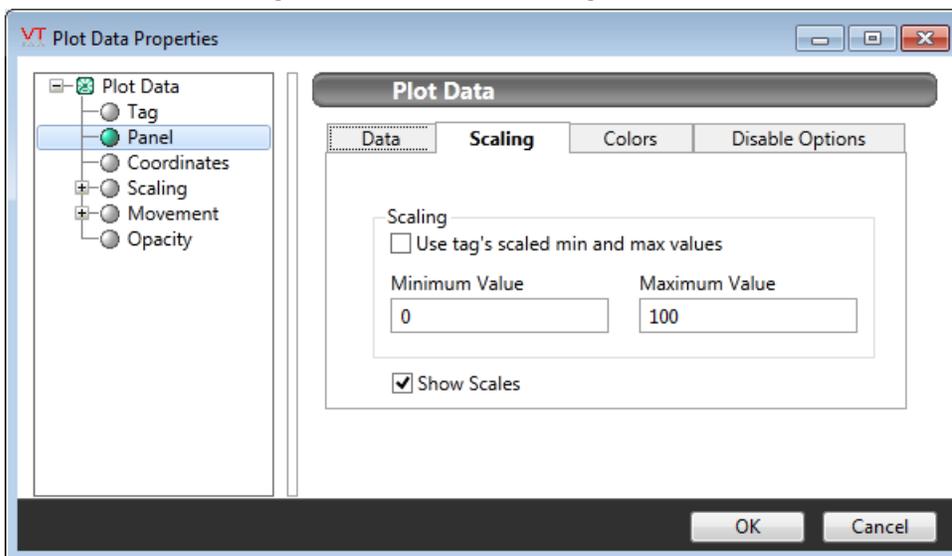
Use Tag's Scaled Min and Max Values

Selected this option if you wish to use the Scaled Process Data Min and Scaled Process Data Max properties configured for the tag being drawn. This will be the most common case.
If plotting a tag such as a Calculation, that does not have scaled process values, you can uncheck this box in order to provide the minimum and maximum values expected.

Minimum Value and Maximum Value

These fields adjust the minimum and maximum values shown on the graph's vertical axis. They have no effect on the tag's value, which is what will be plotted.

Show Scales

Select if you wish the minimum and maximum scaling values to be displayed to the left of the completed Plot Data object. Without these values displayed, it can be difficult to judge the meaning of the plotted data.
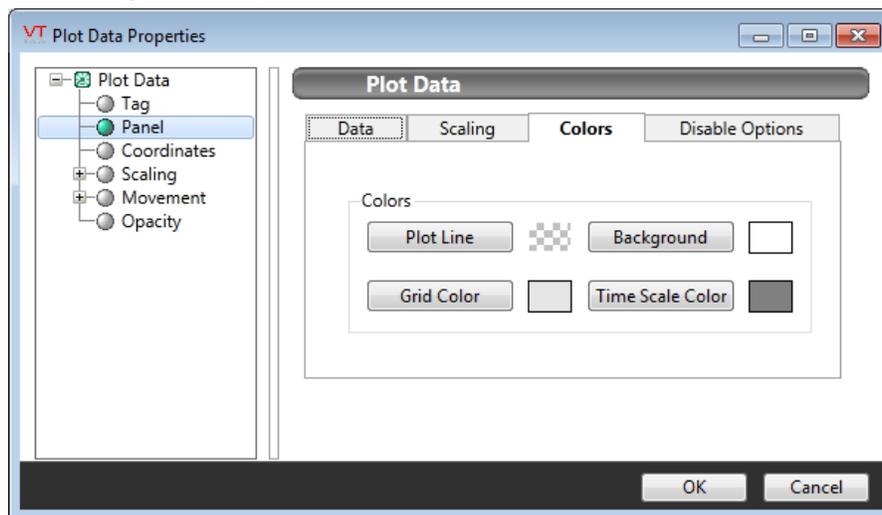
Plot Line

Opens the Select Color dialog from which you can select the

color with which this tag's data will be plotted on the completed object. You should select a color that provides a good contrast from the background.

Grid Color

Opens the Select Color dialog from which you can select the color with which the grid will be drawn on the completed object. You should select a color that provides only a slight contrast from the background so as to not obscure the plot line.

Background

Opens the Select Color dialog from which you can select the background color with which the Plot Data will be drawn.

Time Scale Color

Opens the Select Color dialog from which you can select the color for the numeric scale. A choice that provides a clear contrast from the background will help operators read the numbers.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Analytics\Trends & Historical Data\

Related Information:

...HDV Button Widget

...HDV Hotbox Widget

...Draw HDV Widget

...Choices for Operator Interaction

Polled Station Widget

Used by: DataFlow RTU, Polling Driver.
* Does not use the Stye Settings tag.
The Polled Station tag widget is used to display the number of the RTU being polled.

1

The properties dialog for the Polled Station:
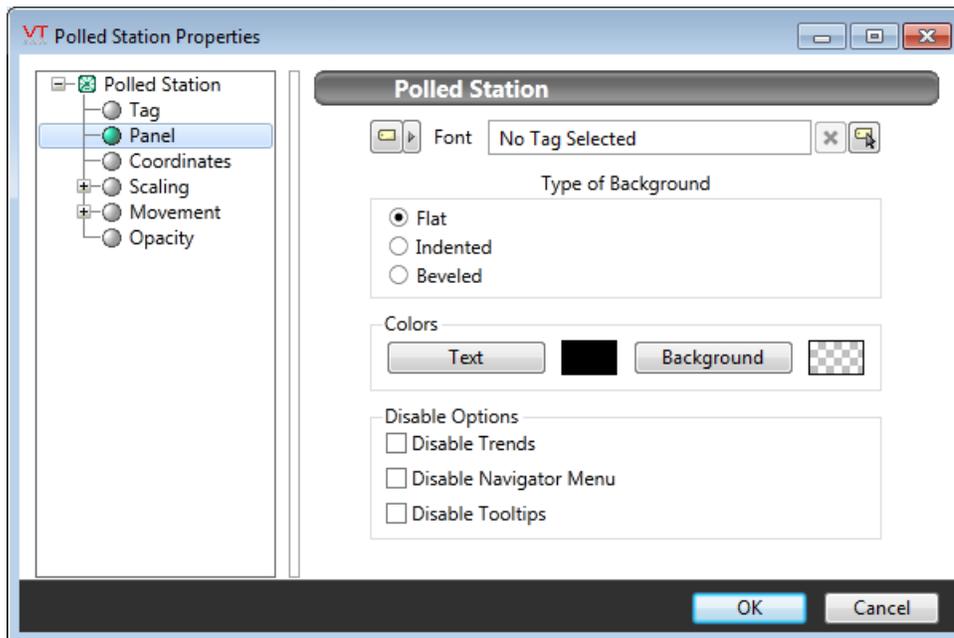
Font

Select one of the font tags (or parameter linked to a font tag) that you wish to use to display the text. The display will use the default system font if you do not select a font tag.
The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to

manage a group of similar controls by defining a font tag
and selecting that for each. Changes to all can then be made
by adjusting the properties of a single font tag.

Type of Border

Examples of the three border types are shown at the beginning
of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the
color of the text.
The background color against which the text will be displayed
can also be defined.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Communications\Polling\
Analytics\Diagnostics\

Related Information:
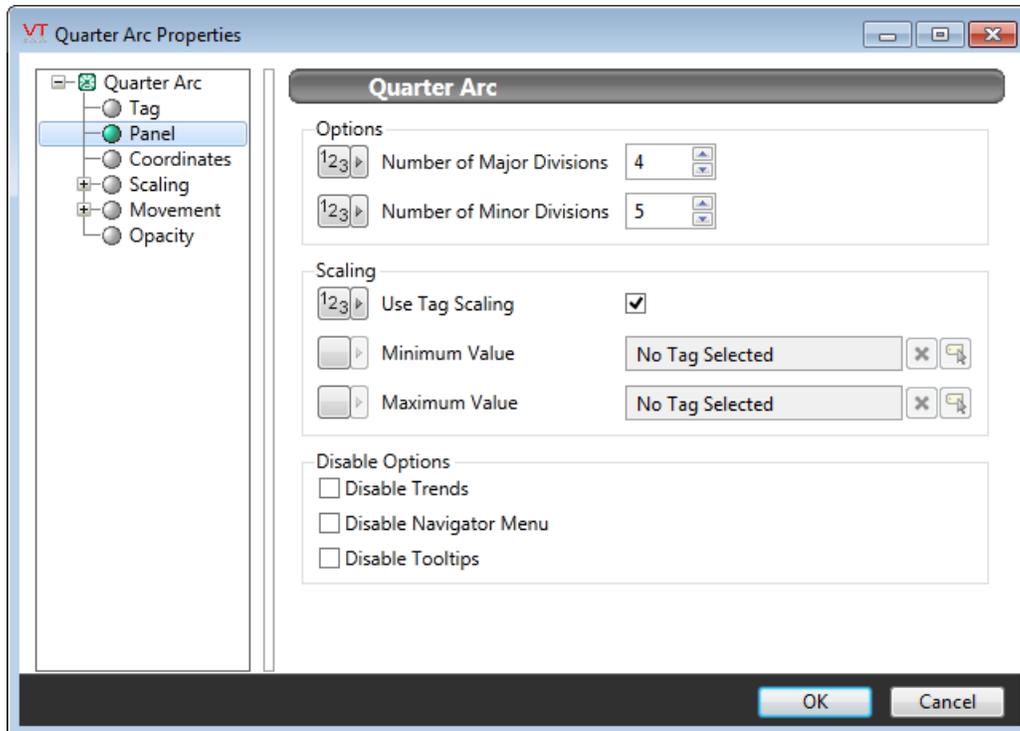
...Choices for Operator Interaction

Power Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm
Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Power widget represents the associated tag's value as a photoreal-
istic meter.

The properties dialog for the Power widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and 5.
Choose whether to include the text used for engineering units within the gauge. Defaults to show units.

Colorization

Use the color selector dialogs to set the color of the needle, text and gauge background. This widget does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
Choices for Operator Interaction

Pressure Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm
Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Pressure Gauge widget represents the associated tag's value as
radial meter.

The properties dialog for the Pressure Gauge widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and 5.

Choose whether to include the text used for engineering units within the gauge. Defaults to show units.

Colorization

Use the color selector dialogs to set the color of the needle, text and gauge background. This widget does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
Choices for Operator Interaction

Program Spawn Widget

Not linked to any tag.
* Does not use the Stye Settings tag.
The Program Spawn widget is drawn as a button on any page. In response to being clicked, it will run any configured program that can be found on your computer.

Program Spawn is commonly used to open a notebook or spreadsheet. The properties dialog for the Program Spawn widget:
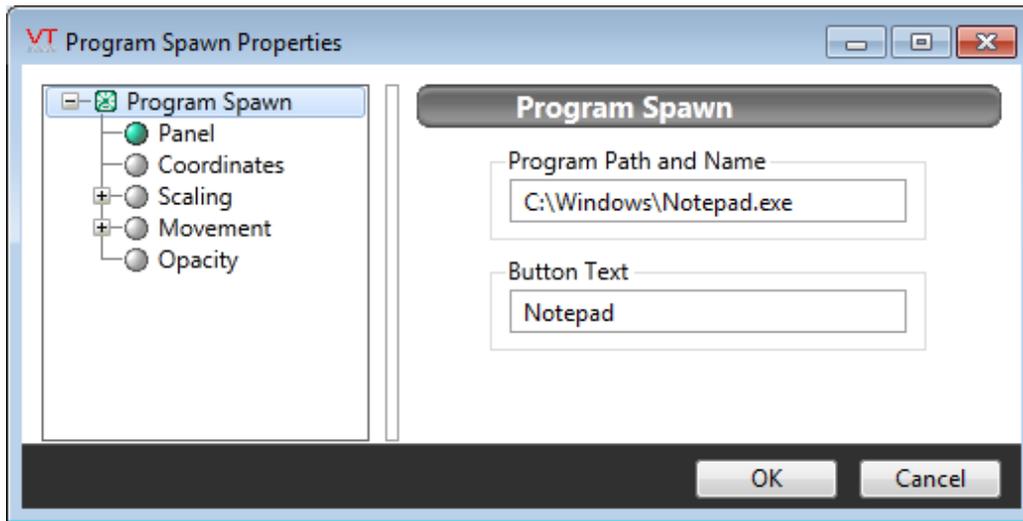
Program Path and Name

The full path to the program to run.
If the program accepts parameters (such as the name of a file to open), they may be provided after the name of the execut-able file.

Button Text

The label that will be displayed on the button. Should inform the operator of what to expect when the button is pressed.

Widget Folders:
Basic Components\Specialty\Misc\
Tools\Standard Library\

Pulse Beacon Widget

Used by: Custom site icons
* Does not use the Stye Settings tag.
The Pulse Beacon widget indicates the selected site on a map. It is one component of a site icon and is not meant to be used alone.

The expanding circle is a pulse beacon.

There are no user-configurable properties for the Pulse Beacon widget.

Widget Folders:
Basic Components\Specialty\Mapping\
Tools\Standard Library\

Related Tasks:

...Create Custom Map Icons

Push Button Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control, Digital Input, Digital Status, Digital Output, Digital Control, MultiWrite, Pump Status, Selector Switch, Trigger.
* Does not use the Stye Settings tag.
The Push Button widget provides a method for operators to issue control actions, via the linked tag. The widget's configuration holds the value that will be written to the linked tag when the button is clicked.

The properties dialog for the Push Button widget:
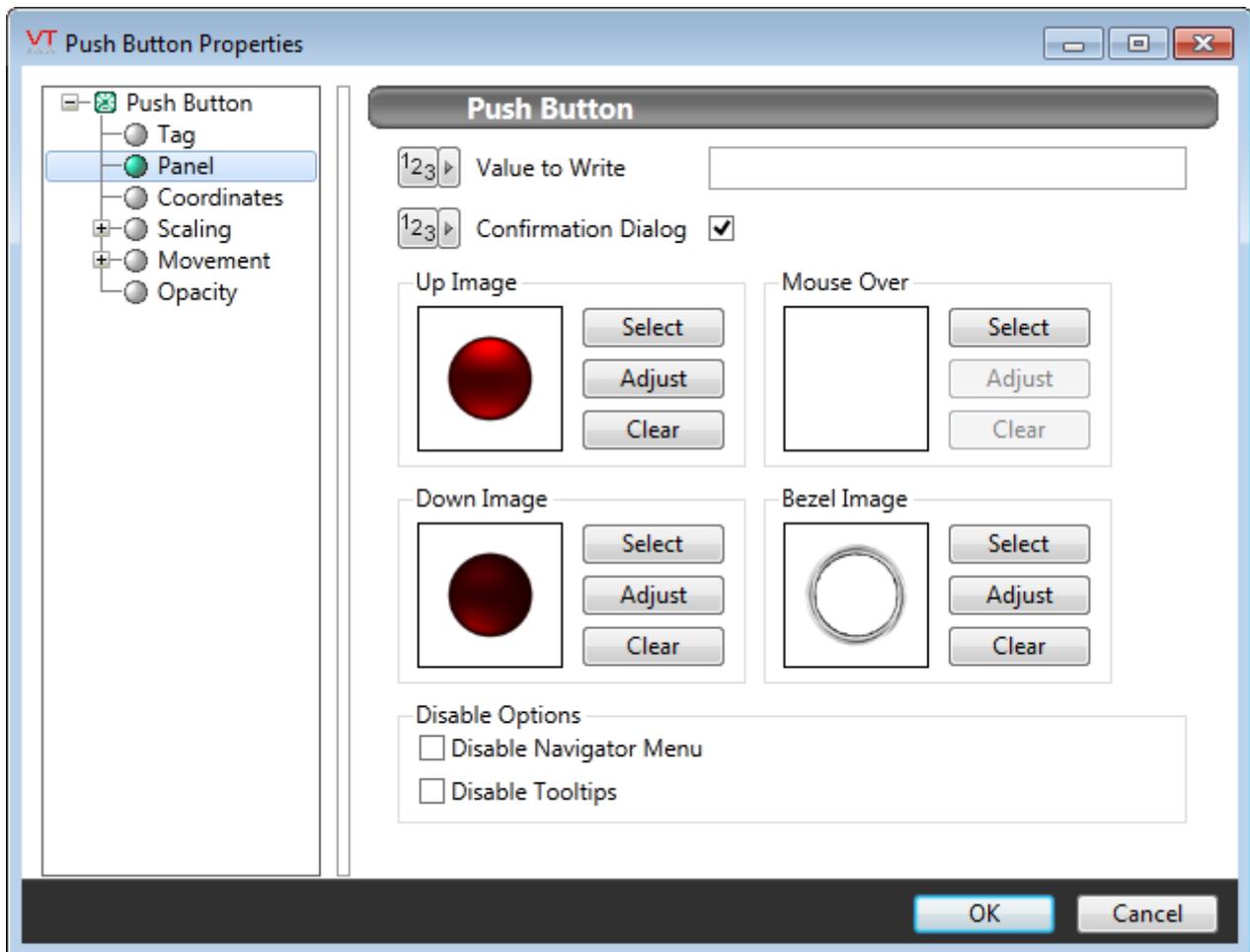
Value to Write

Set the value that will be written when the completed object is
clicked. If the associated tag is an analog or digital output, this
may be any value that is legal for that tag. Otherwise, this
value should match one of the configured states for the asso-
ciated selector switch.
If this value is not set, the button will not work.

Confirmation Dialog

If set, the operator will need to confirm every value change
made by this method.

Up Image

The image that will be shown when the button is not being clicked.

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Down Image

The image that will be shown when the button is being clicked.

Mouse Over

The image that will be shown when the cursor is over the image.

Bezel Image

An image that should be scaled to surround the button, so that it does not appear to be floating above a panel.

Disabled

The image that will be shown if the button cannot be used (for example, due to a communication error). Can be helpful to let operators know that a control action will not be issued.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Quarter Arc Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Quarter Arc Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.
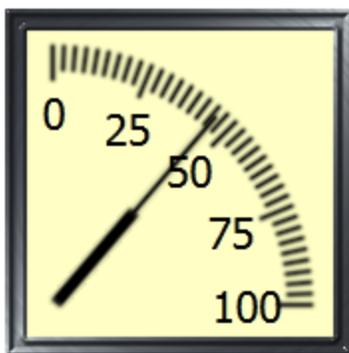
The properties dialog for the Quarter Arc Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.

You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:
...Choices for Operator Interaction

Report Button Widget

Used by: Report tag.
* Does not use the Stye Settings tag.
The Report button tag widget is used to create a button that the operator may click to generate a predefined report. The report must be configured within the associated Report tag, but other widgets are available to allow the operator to change the configuration.
If the Report tag is not configured, the button will be drawn but not enabled.

The properties dialog for the Report Button widget:
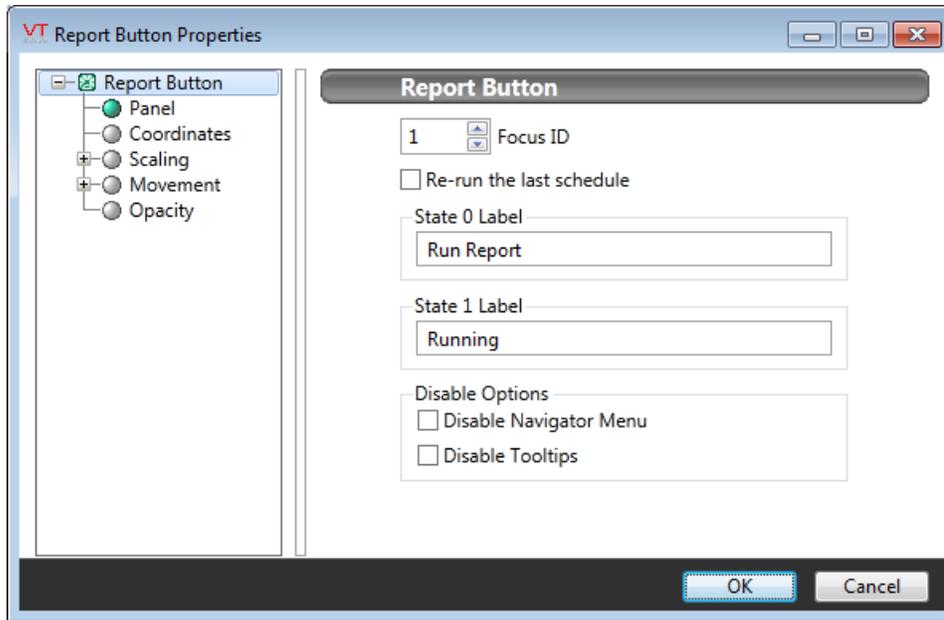
Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)

When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).

You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Re-run the last schedule

When checked, clicking the button will re-run the last scheduled report rather than generating a report that uses the current time.

State 0 Label and State 1 Label

The state 0 label will be shown most of the time. The state 1 label is shown only when the button is pressed and the report runs.

The State 0 label should contain just enough text to allow the operator to know which report will run when the button is clicked and the time frame that will be included in the report (especially if the Re-run last schedule option is checked). The State 1 label should provide confirmation that the button has indeed been clicked upon.

If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tools\Report Tools\
Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Report Destination Widget

Used by: Report tag.
* Does not use the Stye Settings tag.
The Report Output Type tag widget is used to draw a group of objects that operators can use to configure the output format and path for their report.
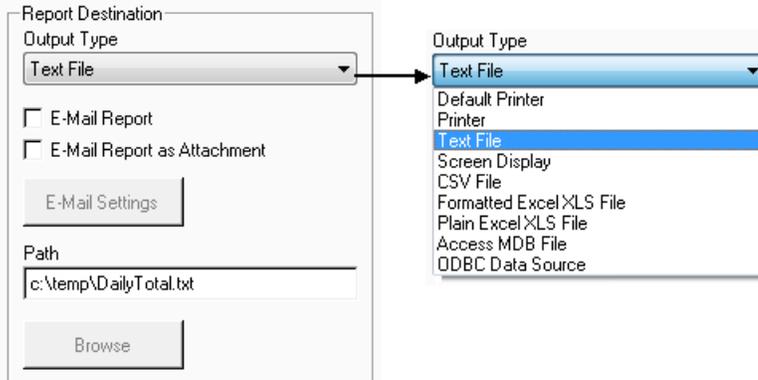
Information on each of the report output types shown can be found in the topic: Types of Reports. Not all types will include an email option, and for those that do, you must have an email server capable of sending the file. VTScada relies on your email server – it does not provide its own. The Path option is enabled only when the output type involves saving the report to disk.

The properties dialog for the Report Options widget:

Title

Provide the text you wish to be displayed above the completed Report Output Type object group. By default, the text "Report Destination" will be provided. You may leave this field blank if you prefer not to have a title.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)

When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).

You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

All Users Can Edit

Controls whether or not all logged on users may use the objects in the completed Report Option List group, without regard as to the privileges that have been granted to their user account.

If not selected, then only users who have the Configure privilege granted to their user account may use the objects in the Report Option List group.

This option is useful for allowing administrative personnel to create and configure reports without granting configuration privileges. Information on the Configure security privilege can be found in "System Privileges".

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom as shown. You can save time by putting the object where you would like on the screen and not making the drawing area larger than the object.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Tools\Report Tools\

Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Report Iterations

This tool is used to draw a field into which operators can enter the number of report iterations (i.e. consecutive reports) they wish the report generator to produce. For example, if an operator wishes to generate a report for each day in the last month, they could set the report time period to the last day of the last month (e.g. Thu Nov 30, 2001 12:00 pm), and then enter "30" in the Iterations Entry field.

The report generator produces the first report based on the duration specified by the Start Time and End Time spin boxes, and then generates consecutive reports based on the same time period, but on prior dates (present to past).

The following image shows an example of the Report Iterations dialog.
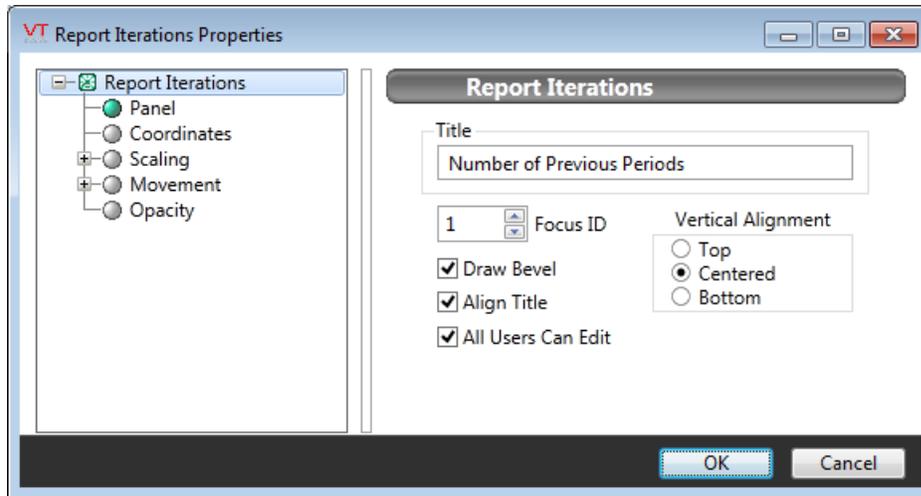
The result of the above configuration is an entry field labeled, "Number of Previous Periods" into which operators can enter a number representing the number of report iterations they wish the report generator to produce.

Title

Whatever you type for the title will be displayed above the Iterations Entry field. The default title is "Number of Previous Periods".

Focus ID

The Focus ID spin box enables you to select a number representing the order of the completed object in the overall tab order.

When you configure some or all of the report tools on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).

Draw Bevel

The Draw Bevel check box is used to indicate whether or not the completed iterations entry object should be drawn with a beveled border. If the Draw Bevel check box is selected, the completed iterations entry object will be drawn with a beveled border surrounding it. If the Draw Bevel check box is not selected, the completed iterations entry object will be drawn without a beveled border.

Align Title

The Align Title check box is used to specify whether or not you wish the Iterations Entry field's title to be included in the Vertical Alignment calculation.

Vertical Alignment

The Vertical Alignment radio buttons enable you to specify the alignment to be applied to the Iterations Entry field, and its title and bevel vertically within its drawing area. You may select one of:

- Top: The Iterations Entry field, its title (if the Align Title check box has been selected), and its bevel (if the Draw Bevel check box has been selected) will be aligned to the vertical top of its drawing area.
- Centered: The Iterations Entry field, its title (if the Align Title check box has been selected), and its bevel (if the Draw Bevel check box has been selected) will be aligned to the vertical center of its drawing area.
- Bottom: The Iterations Entry field, its title (if the Align Title check box has been selected), and its bevel (if the Draw Bevel check box has been selected) will be aligned to the vertical bottom of its drawing area.

All Users Can Edit

Use to select whether or not all logged on users may enter values in the completed Iterations Entry field, regardless of their assigned privileges. If this check box is not selected, then only users who have the Configure privilege may enter values into the completed Iterations Entry field.

Report Options Widget

Used by: Report tag.
* Does not use the Stye Settings tag.
The Report Option List tag widget is used to draw a trio of check boxes that operators can use to select certain formatting attributes for their report.
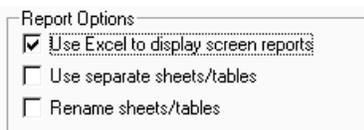
Use Excel to Display screen reports.

This first option applies only to reports that are configured to output to the screen. If that is the case, then selecting this option will cause Microsoft Excel™ to be opened to view the report (if Excel is installed on the workstation). The report would otherwise be shown using a simple text format.

Use separate sheets/tables.

This option applies only if the report is being output to either a spreadsheet (Microsoft Excel™) or a database (Microsoft Access™) and if it has been configured to report on more than one consecutive time period. Each time period will be stored in its own worksheet or table.

The default names will be "Sheet 1," "Sheet 2," or "Table 1," "Table 2" …

Rename sheets/tables.

Applies only if the preceding option also applies and is selected. If checked, each sheet or table will be named for the time period it contains. This name will use the following template:

```
<Date (Month/day - MMMDD)><Period Start (Hours/Minutes -
HHMM)>_<Period End (Hours/Minutes - HHMM)>
```

For example, a report run on May 25th at 17:20 and covering three consecutive 1-hour time periods will have sheets with the following names:

May251720_1820
May251620_1720
May251520_1620

The properties dialog for the Report Options widget:
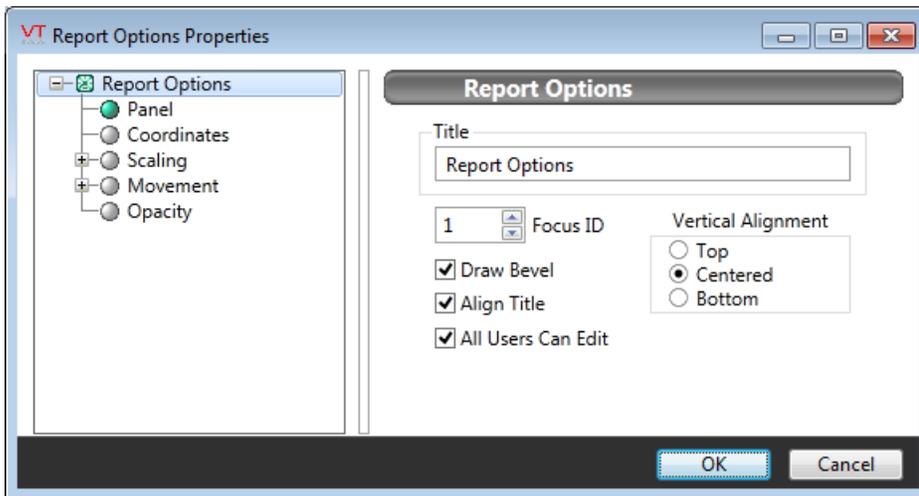
Title

Sets the title, should you wish to include one above the option list. Enabled whether the bevel is drawn or not.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)

When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).

You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

All Users Can Edit

Controls whether or not all logged on users may use the objects in the completed Report Option List group, without regard as to the privileges that have been granted to their user account.

If not selected, then only users who have the Configure privilege granted to their user account may use the objects in the Report Option List group.

This option is useful for allowing administrative personnel to create and configure reports without granting configuration privileges. Information on the Configure security privilege can be found in "System Privileges".

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom as shown. You can save time by putting the object where you would like on the screen and not making the drawing area larger than the object.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Tools\Report Tools\

Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Report Tag List Widget

Used by: Report tag.
* Does not use the Stye Settings tag.
The Report Tag List tag widget is used to draw a group of tools that operators can use to select the tags to be included in a report.
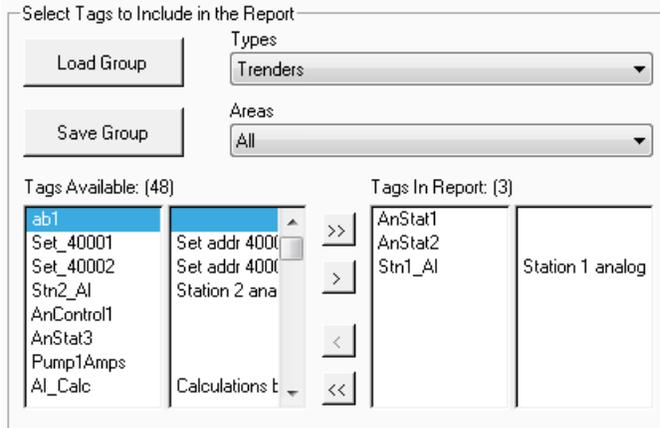
If you provide this report drawing tool to operators, you must ensure that they know the following:

- Only tags that are being logged will have data available for a report.
- Merely selecting a group of tags with this tool is not sufficient for the associated Report tag to actually use them. The selected tags must either be saved as a group, or loaded as a group.

The filtering and selection tools here are the same as those provided in the Reports page. For details on how to use this, please see: Select Tags for a Report.

The properties dialog for the Report Tag List widget:
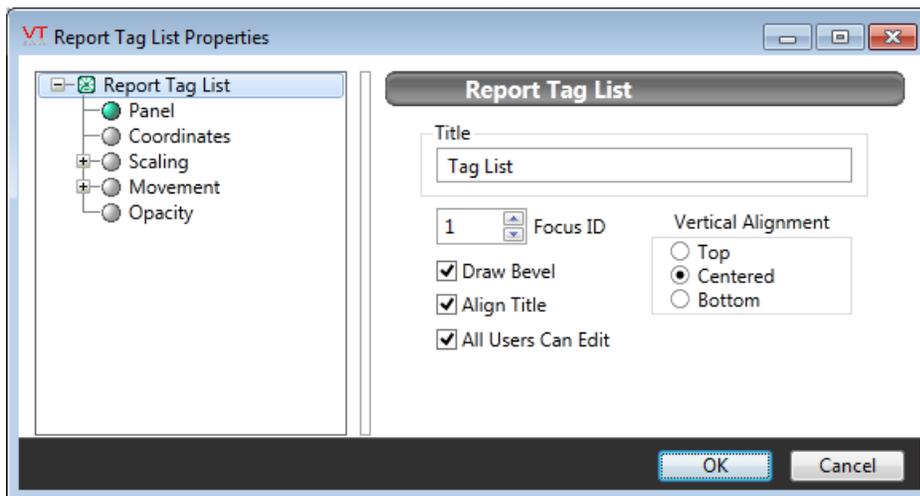
Title

Provide the text you wish to be displayed above the completed

Report Tag List. By default, the text "Tag List" will be provided. You may leave this field blank if you prefer not to have a title.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)
When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).
You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

All Users Can Edit

Controls whether or not all logged on users may use the objects in the completed Report Tag List, without regard as to the privileges that have been granted to their user account.

If not selected, then only users who have the Configure priv-
ilege granted to their user account may use the objects.
This option is useful for allowing administrative personnel to
create and configure reports without granting configuration
privileges. Information on the Configure security privilege can
be found in "System Privileges".

Vertical Alignment

Locates the object vertically within its drawing area. May be
one of Top, Centered or Bottom as shown. You can save time
by putting the object where you would like on the screen and
not making the drawing area larger than the object.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tools\Report Tools\
Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Report Type Widget

* Does not use the Stye Settings tag.
The Report Type widget is used to create a drop-down list from which
operators can select the type of report they wish to generate.
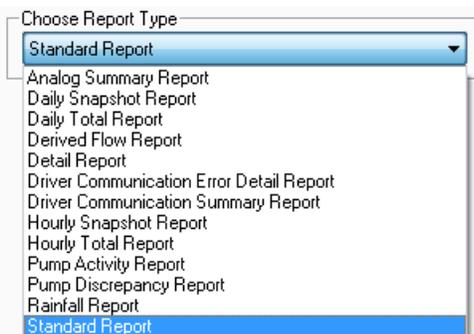
The list of report types available will vary depending on whether you have any custom reports defined.

At the very least, you can expect to see:

- Daily Snapshot Report: Displays a "snapshot" of the value of each tag in a selected set of loggers at the current time for a user–defined time period. For example, if at 2:00 pm you select a series of tags, set the time period to "Last Week" and run the report, the daily snapshot report will display the value of each tag at 2:00 pm for the last 7 days.

- Daily Total Report: Displays the total data accumulated for a selected set of tags at the current time within a user–defined time period. For example, if at 2:00 pm you select a series of tags, set the time period to "Last Week," and run the report, the daily total report will display the value accumulated for each tag by 2:00 pm for the last 7 days.

- Hourly Snapshot Report: Displays a "snapshot" of the value of each tag in a selected set of loggers at each hour within a user–defined time period. For example, if at 2:00 pm you select a series of tags, set the time period to "Last Week," and run the report, the hourly snapshot report will display the value of each tag every hour for the last week, starting at 2:00 pm today.

- Hourly Total Report: Displays the total data accumulated hourly for a selected set of tags (starting at the current time) within a user–defined time period. For example, if at 2:00 pm you select a series of tags, set the time period to "Last Week," and run the report, the hourly total report will display the accumulated total for each tag every hour for the last week, starting at 2:00 pm today.

- Standard Report:Displays the value of the selected tags, every X seconds, for a user–defined time period.

Note: These and other report types are described in greater detail in Selecting a Report Type.

The properties dialog for the Report Type widget is as shown:

Title

Provide the text you wish to be displayed above the completed Report Output Type object group. By default, the text "Report Destination" will be provided. You may leave this field blank if you prefer not to have a title.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)

When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).

You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

All Users Can Edit

Controls whether or not all logged on users may use the objects in the completed Report Option List group, without regard as to the privileges that have been granted to their user account.
If not selected, then only users who have the Configure privilege granted to their user account may use the objects in the Report Option List group.
This option is useful for allowing administrative personnel to create and configure reports without granting configuration privileges. Information on the Configure security privilege can be found in "System Privileges".

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom as shown. You can save time by putting the object where you would like on the screen and not making the drawing area larger than the object.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

...Choices for Operator Interaction

Reporting Period Widget

Used by: Report tag
* Does not use the Stye Settings tag.
The Reporting Period widget is used to draw spin boxes and a drop-down list that operators can use to configure the time period for a report. The available presets are shown to the right of the completed widget:
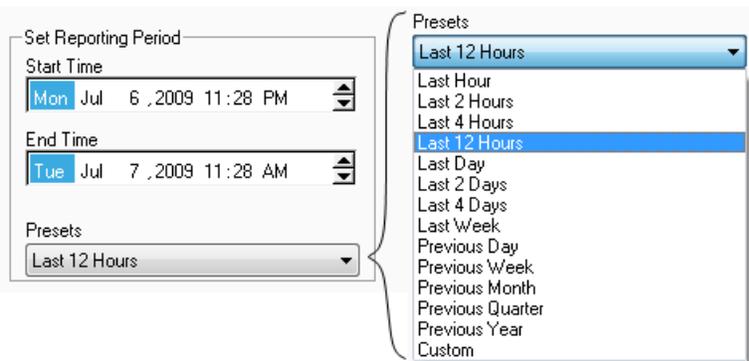
The properties dialog for the Reporting Period widget:
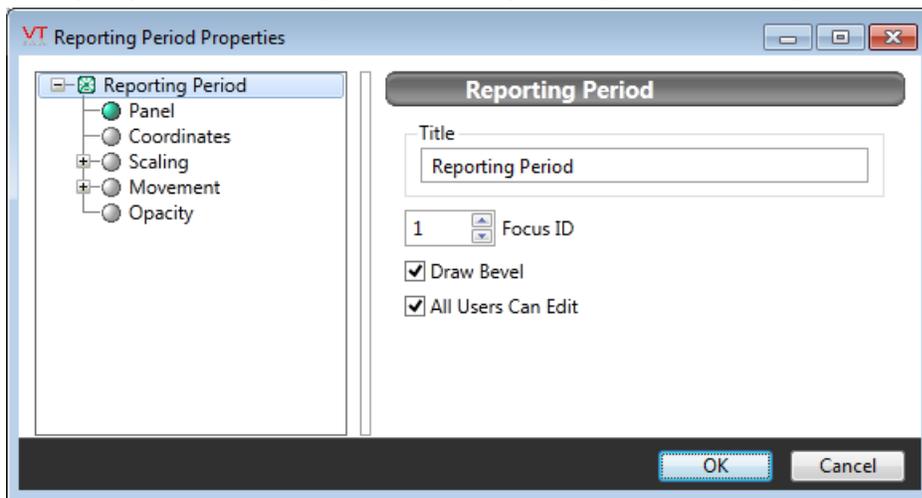
Title

Provide the text you wish to be displayed above the completed Reporting Period controls group. By default, the text "Reporting Period" will be provided. You may leave this field blank if you prefer not to have a title.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)
When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).
You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

Align Title

Used in conjunction with the Vertical Alignment choice. Selects whether the title is included when aligning the object vertically within its overall bounding area. See Vertical Alignment for an example.

All Users Can Edit

Controls whether or not all logged on users may use the

objects in the completed Reporting Period controls group, without regard as to the privileges that have been granted to their user account.

If not selected, then only users who have the Configure privilege granted to their user account may use the objects in the Report Option List group.

This option is useful for allowing administrative personnel to create and configure reports without granting configuration privileges. Information on the Configure security privilege can be found in "System Privileges".

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom as shown. You can save time by putting the object where you would like on the screen and not making the drawing area larger than the object.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Tools\Report Tools\

Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Reporting Period (Enhanced) Widget

Used by: Report tag.

* Does not use the Stye Settings tag.

The Reporting Period Enhanced widget provides every possible option to configure the time period for a report. You can select from of:

- A set of predefined full time periods.
- A set of predefined time lengths prior to the trigger.
- A custom time period, of any duration and end time.

Presets include:

- Last XX before trigger time
  where XX is one of :
  7 days, 4 days, 2 days, 1 day, 12 hours, 4 hours, 2 hours or 1 hour
- Previous XX
  Where XX is one of the following full time periods:
  Day, week, month, quarter or year
- Custom
  Where custom can be either a Previous (full) Time Period, selected using the
  Report Duration fields or, Duration and End Time, where the end time is
  defined to occur at some length of time before the report was triggered and
  the included time period is defined using the Report Duration.

The properties dialog for the Reporting Period widget:
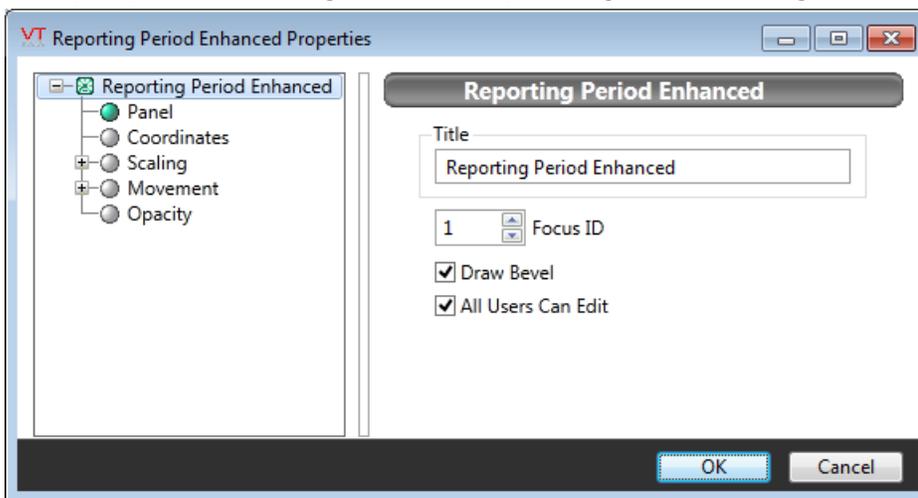
Title

Provide the text you wish to be displayed above the completed Reporting Period controls group. By default, the text "Reporting Period Enhanced" will be provided. You may leave this field blank if you prefer not to have a title.

Focus ID

Select a number representing the placement of this object in the page's overall tab order. (Optional)
When you draw objects on a page, each object is given a focus ID that identifies that object's place in the tab order (that is, which object will be selected when the Tab key is pressed). This does not control which object will initially have focus when a page is opened – that belongs to the first item drawn on the page that can receive focus. (i.e. the first graphics statement in the page's source file).
You can disable a control widget by setting its Focus ID value to zero. The widget will remain visible, but faded.

Draw Bevel

Encloses the object in a beveled rectangle when selected. The example shown at the beginning of this topic uses a title and bevel.

All Users Can Edit

Controls whether or not all logged on users may use the objects in the completed group, without regard as to the privileges that have been granted to their user account.
If not selected, then only users who have the Configure privilege granted to their user account may use the objects in the Report Option List group.

This option is useful for allowing administrative personnel to create and configure reports without granting configuration privileges. Information on the Configure security privilege can be found in "System Privileges".

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tools\Report Tools\
Tag Types\Data Logging & Reports\Report\

Related Information:

...Choices for Operator Interaction

Reset Button Widget

Used for: Totalizer, Counter.
* Does not use the Stye Settings tag.
The Reset Button widget is used to create a button in your application that will reset a Totalizer tag or a Counter tag to zero.
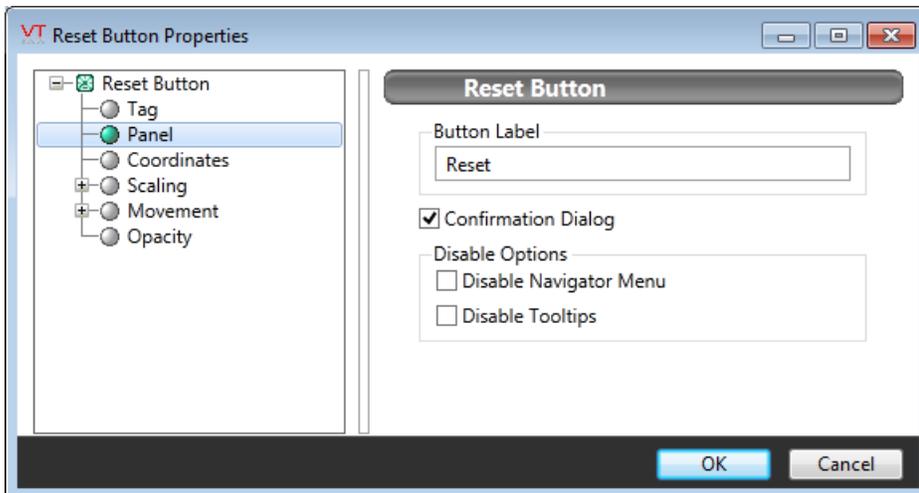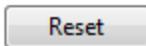
Button Label

Provide a label for the button. Defaults to "Reset".
If the label is to include an ampersand (&), then the & character
will need to be doubled in order to appear.

Confirmation Dialog

Check this option if you want users to be prompted to confirm
that they meant to press the reset button. Defaults to checked.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Reset Polling History Widget

Used by: Enron Modbus Driver

The Reset Polling History widget is used only by the Enron Modbus
Driver. It is drawn as a button, which operators may click in order to
change the next index to be polled back to the first index, which may be
a one or zero depending on driver settings. This can be used to re-read
all data in the device.
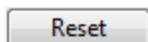
The properties dialog for the Reset History Polling widget:
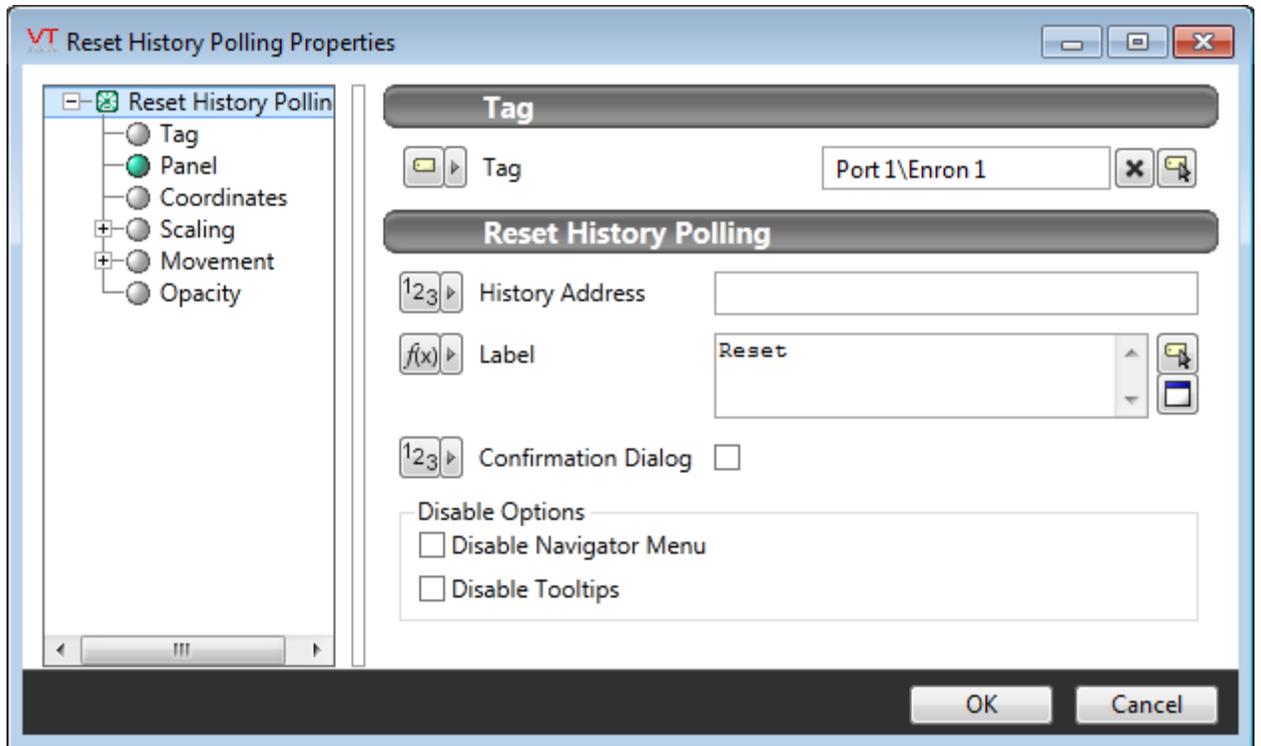
Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

History Address

Provide the numeric value of the history address being monitored. History addresses typically start at 701.

Label

Provide a label for the button. This should describe what will happen when an operator clicks this widget.

Confirmation Dialog

When selected, operators must click through a confirmation

prompt before the reset will proceed.

Disable Options

Disable selected operator-interaction features of this widget.

Reset Target Widget

Used for: Totalizer, Counter.
* Does not use the Stye Settings tag.
The Reset Target widget is used to create a hotbox that will function in the same way as a Reset Button widget. When an operator clicks on the hotbox, the associated Totalizer tag or counter tag will be reset to its starting value. The reset target is shown as a simple rectangle of the configured color. To make its purpose clear, you should consider placing it above a symbol or text that describes its purpose, as shown here (label added as text).
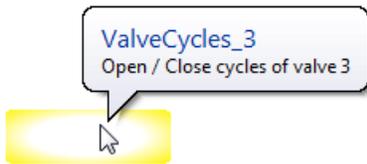
Note: Do not place a hotbox around another tag widget. Clicking on a tag widget will open a historical data viewer for that tag, if that option has not been disabled. If this happens when an operator was expecting to reset a Totalizer tag or counter tag, the result is likely to be confusing.

Besides the Disable Options, there are only two configurable options for this widget: the color to be used for the hotbox, and a choice of whether a confirmation dialog will be shown when the operator clicks in the hotbox area.
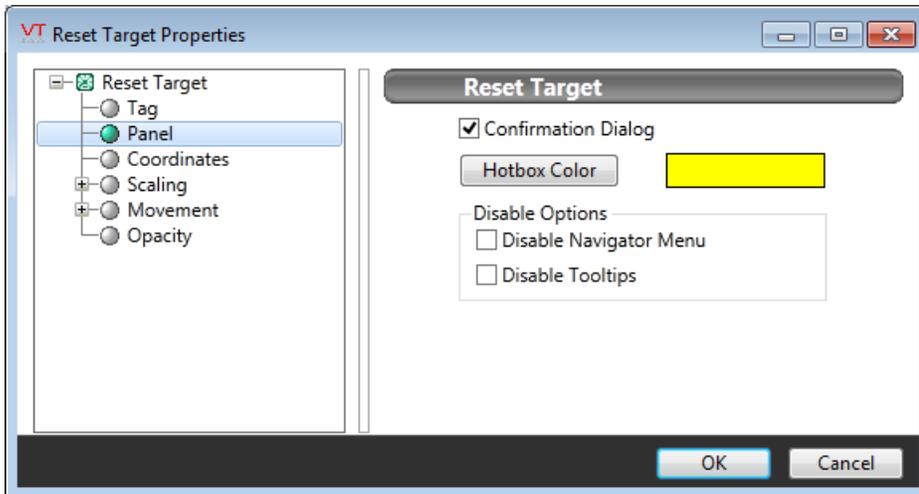
Widget Folders:

Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Retro Gauge Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.
The Retro Gauge widget represents the associated tag's value as a photorealistic meter, displaying both labeled major divisions and unlabeled minor divisions, the dial of which fluctuates according to the tag's value.

The properties dialog for the Retro Gauge widget:

Number of Major Divisions

Controls how many labeled divisions will be drawn around the meter. Each labeled division will also be drawn with a longer line.

Number of Minor Divisions

Controls how many unlabeled divisions will be drawn within each major division. Each minor division will be drawn with a shorter line.

Use tag's scaled min and max values

If checked, the minimum and maximum range of the meter will be controlled by the tag's scaled minimum and maximum values

Minimum Value and Maximum Value

Used to set the smallest and largest values to be displayed on the meter. By default, these values will match the tag's scaled process values.

You can override those values to set your own scale range. For example, assume that you have a tag for which the minimum and maximum scaled process values are 20 and 80 respectively. Drawing this tag with a minimum value of 0 and a maximum value of 100 will result in a meter that shows a full range of numbers from 0 to 100, but whose needle will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\

Related Information:

...Choices for Operator Interaction

Rewrite Outputs Widget

Used by: Allen Bradley Driver, CIP Driver, DNP3 Driver, Driver Multiplexer, Modbus Compatible Device , Omron Driver, Siemens S7 Driver, SNMP Driver.
* Does not use the Stye Settings tag.
The Rewrite Outputs widget is used to force a driver to re-send the last set of values, written to outputs. The button will be enabled only if the driver has been configured to store the values.
This feature may be required if your application uses certain types of hardware that do not retain a memory of their register values during a power loss. It may also be useful when the physical hardware is replaced, since the new device cannot know what was stored in the registers of the old.

Rewrite Outputs

The properties dialog for the Rewrite Outputs widget:
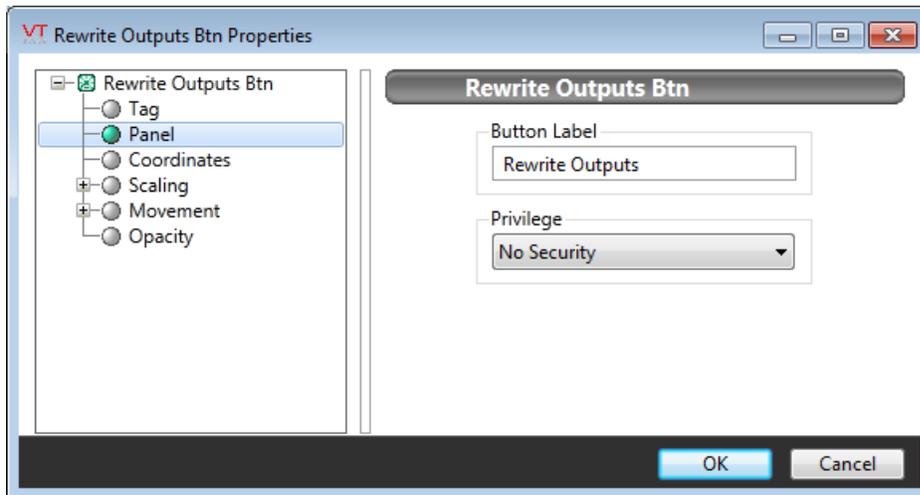
There are only two user-configurable options. You may change the text for the button, and you have the option of restricting access to this function by applying an application-specific security privilege.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Widget Folders:
Basic Components\Multi-Variable\
Analytics\Communications\Output Controls\

Right Bar Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Deadband Control, Function, History Statistics, Script, Selector Switch, SQL Logger.
* Does not use the Stye Settings tag.
The Right Bar widget is used to display a tag's value as a horizontal bar, the length of which increases from the left towards the right as the tag's value increases. When the tag's value is at a minimum, the bar will be shown as a thin line at the left edge of its range. When the tag's value reaches its maximum, the widget will be displayed as a rectangle, filling the range.

	being edited at 54%
	at minimum
	at maximum

The properties dialog for the Right Bar widget:

Set Color

Opens the Select Color Dialog from which you can select a color for this object.

Low Scale Range and High Scale Range

In most cases, these fields will be display-only, showing the Scaled Process Data Min and Scaled Process Data Max values configured for the associated tag.

These fields become editable when the widget is used to display a function or Calculation tag. The most common use is to provide an output range of 0 to 100 (%) given a data source whose numbers do not match these values (perhaps ranging from 20 to 80, for example). Set the low scale range to the value expected from the data source when it is at its minimum

and the high scale range to the value expected from the data source when it is at its maximum.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\
Basic Components\Bars\

Related Information:

...Choices for Operator Interaction

ROC Driver History Info Button

Used by the Fisher ROC driver.
Creates a button that operators may press to access details about the driver's history. The contents of the table will vary depending on the device and whether the device is storing hourly or daily history.
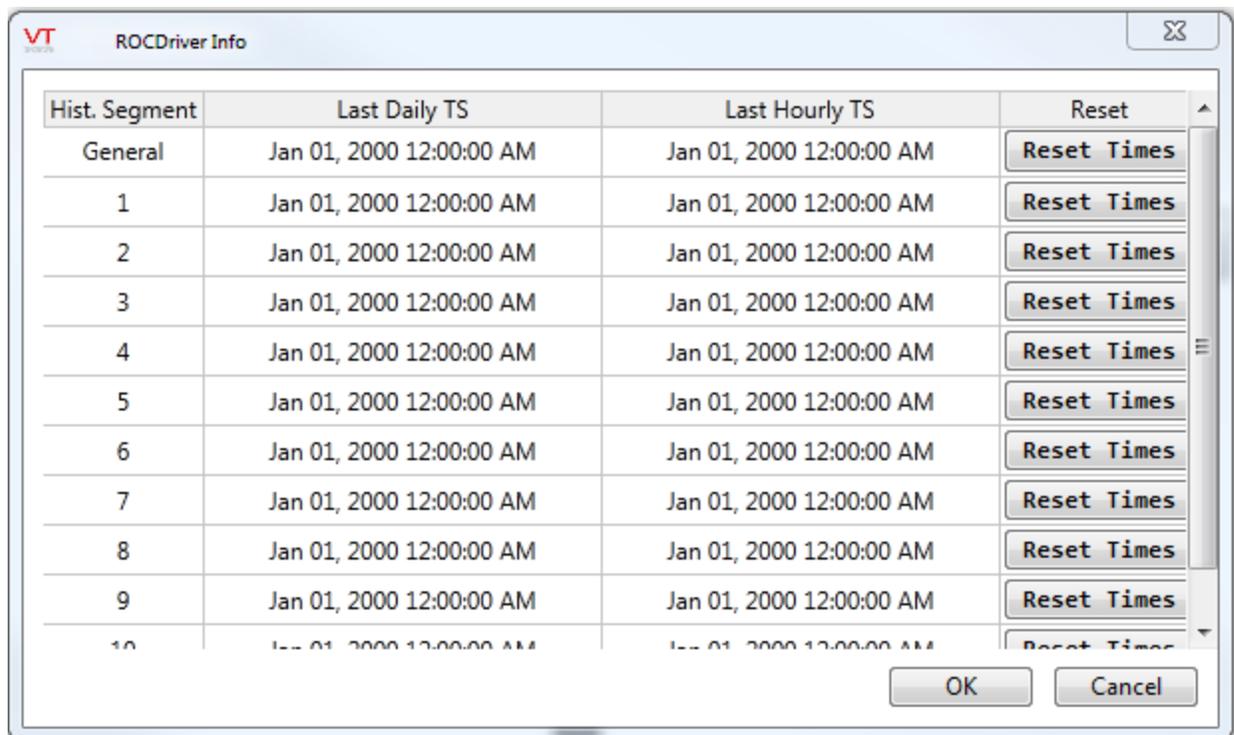
Hist. Segment	Last Daily TS	Last Hourly TS	Reset
General	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
1	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
2	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
3	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
4	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
5	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
6	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
7	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
8	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
9	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times
10	Jan 01, 2000 12:00:00 AM	Jan 01, 2000 12:00:00 AM	Reset Times

ROCDriver Info

OK   Cancel

Sample history for a ROC Plus device.

The ROC driver reads history from tables within the ROC RTU. In the standard ROC protocol, the values in these tables are arranged by RAM Areas, whereas in the ROC Plus protocol they are arranged in History Segments.

Each time that VTScada reads the device's history, it saves the timestamp of the last value read. For the next read, it compares the new timestamp to the stored value. If there is no new information then the tables will not be read again. Each value is read only once.

If you wish to re-read values that have already been read, click the Reset Times button. This causes the next read to re-read the entire history table, not just the most recent information.

The only user-configurable properties for the button widget are the label and the standard Disable Options.

Related Information:

Fisher ROC Driver Tags

Roster Alarm Test Widget

Used by: Roster tags.
* Does not use the Stye Settings tag.
The Roster Alarm Test widget is used to create a check box that operators can click to generate an alarm that will test the roster configured for the Alarm Notification System.

☐ Night Roster Alarm Test

The properties dialog for the Roster Alarm Test widget:
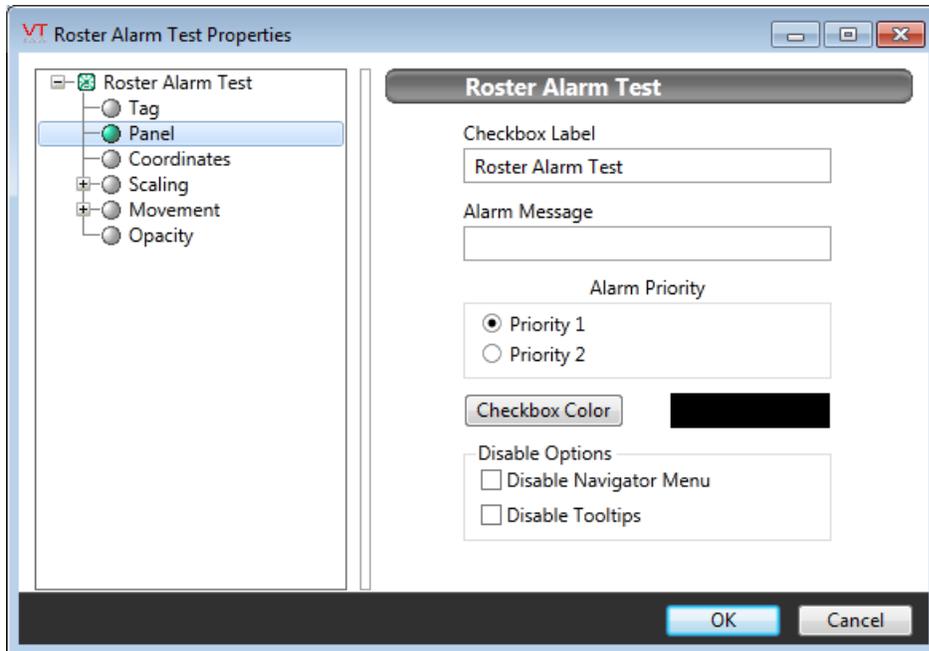
Check Box Label

The text you enter here will be displayed to the right of the completed Roster Alarm Test check box. It should clearly tell operators what the purpose of that check box is. By default, the tag name will be used before the words "Alarm Test".

Alarm Message

Set the message you wish the VTScada call-out system to transmit when performing this test. Operators receiving this message should have no doubt that they are seeing a test, not a real alarm.

Alarm Priority

Select the alarm priority you wish to configure for this test alarm. This can be either Priority 1 (i.e. Critical) or Priority 2 (High).

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Basic Components\Specialty\Roster Control\

Related Information:

...Choices for Operator Interaction

Rotary Control Widget (Knob)

Used by: Analog Input, Analog Output, Analog Status, Analog Control
* Does not use the Stye Settings tag.
The Rotary Control enables operators to select a value to write to hardware by turning a knob. This is done by clicking on, then dragging the control clockwise or counter-clockwise. The value that will be set is shown in a tooltip while dragging.

Properties dialog for the Rotary Control:
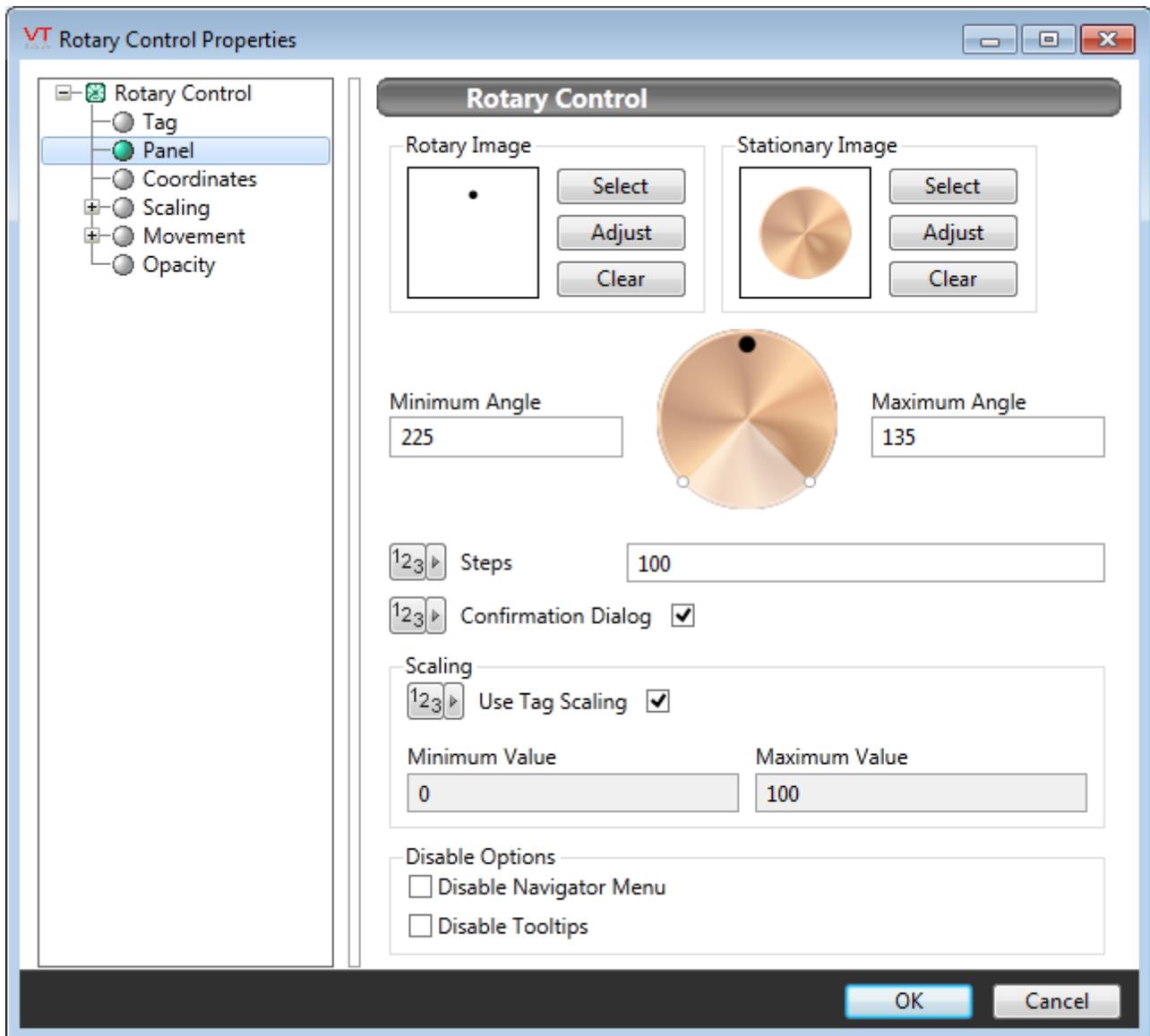
Rotary Image

The image that simulates the knob motion by rotating in response to the operator action. Usually a dot or bar that indicates the current position.

If creating your own image, it must be transparent other than the position indicator, which must be directly above the center of the image:
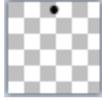
Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Stationary Image

The background image, providing context for the rotary image.

Minimum Angle / Maximum Angle

Limit the angle through which the control can rotate. In both cases, the angle is measured clockwise from vertical in units of degrees.

Steps

Sets the precision of the control, relative to the scaling value. The default value enables the operator to select 100 possible values within the scale range. This may not be appropriate for

all controls.

Confirmation Dialog

When checked, the operator must acknowledge each action before the tag will write to equipment.

Scaling

Choose whether to use the tag's minimum and maximum scaling values or to set alternative scaling values that will be applied to the output value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analog Controls\

Selector Switch Widget

Used by: Selector Switch Tags
* Does not use the Stye Settings tag.
The Selector Switch widget is used to represent a Selector tag as either a 2-position or a 3-position switch. A common use is to create a Hand-Off-Auto selector.

If the matching Selector tag has been configured for only two positions, the middle position cannot be used. The label will be blank in this case, even if you have specified an override within the widget configuration. The widget is composed of several user-configurable parts: the base image, the switch image (i.e. the knob) and the labels. Additionally, you

can change the switch angle. Any value from 0 to 120 is permitted, but small angles may be difficult for operators to use. The labels will be relocated to match the new switch positions.

The selector switch has built-in mechanisms to provide feedback to the operator. If an alarm occurs due to a mismatch between expected and actual feedback values, the text in the selected position will glow red. In the event of a mismatch between the requested and actual switch position, visual feedback will be provided by the switch blinking between the current position in solid shades and the requested position as semi-transparent. (see the following image:)
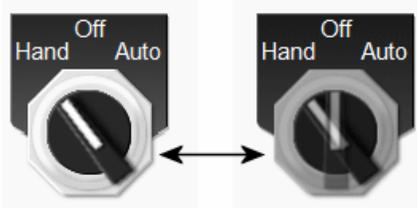
If the position feedback is Invalid, the image will simply blink in its set position.

If the Selector tag has only two positions, then the left position in the widget corresponds to position 0 in the Selector tag and the right position is 1. If the Selector tag has three positions, then middle corresponds to position 0, the left to 1 and the right corresponds to position 2.
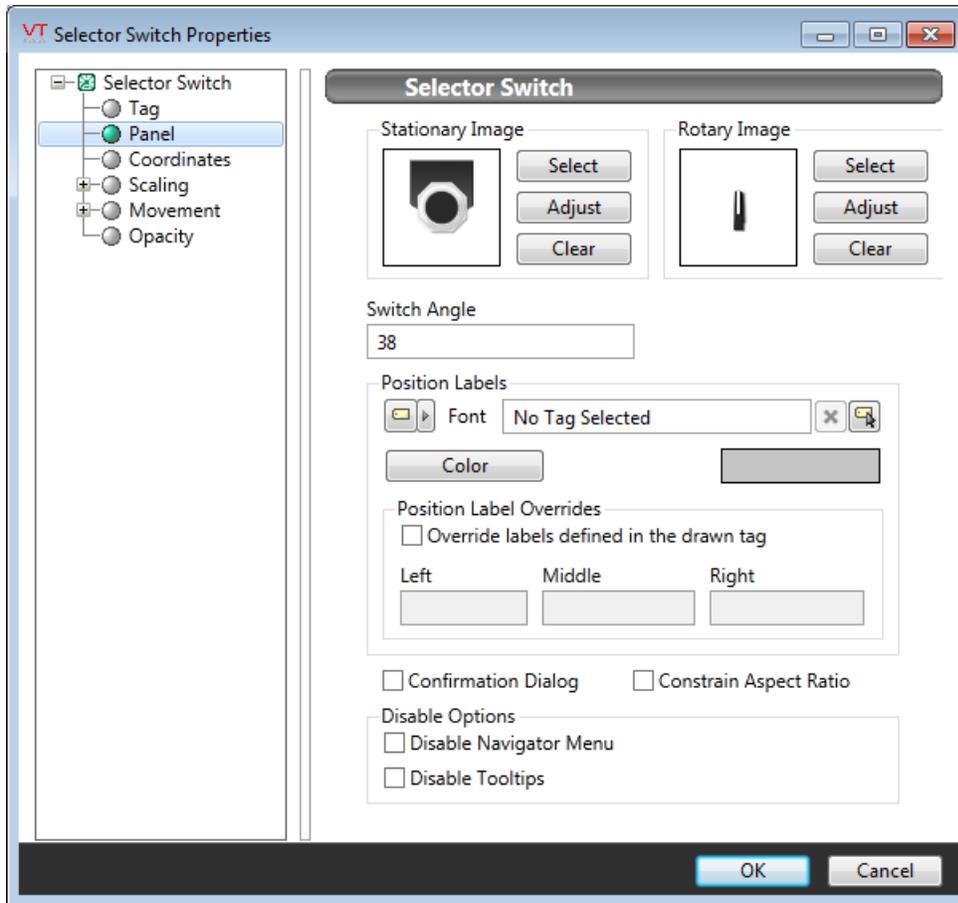
Stationary Image

Any image may be chosen for the base image, but three in par-
ticular are provided in the Selector Switches folder: Plate1a,
Plate1a-Large and Plate1b. (See example at the beginning of
this topic). The base image provides a background for the
switch and the labels.

Image Select

Opens the Select Image dialog, within which you can
browse for images, import new images, and select
the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Rotary Image

Any image may be chosen for the switch image. The Selector Switches folder provides three options: Switch1a, Switch1a-Large and Switch1b. This provides a control and indicator knob that will appear on the base image background. Variations of the image shown above can be found in the Selector Switches group of the images library and you are free to add your own images.

Switch Angle

This is the angle between each consecutive position of the switch. Any value from 0 to 120 may be set. As shown in the following images, an angle that is too small (5 deg, shown on the left) will be difficult to use, while an angle that is larger (120 deg, shown on the right) may cause the labels to be displayed outside the area of the base image.

Font

Select one of the font tags (or parameter linked to a font tag) that you wish to use to display the text. The display will use the default system font if you do not select a font tag. The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Color

Opens the Select Color Dialog from which you can select the color of the text.

Position Label Overrides

Set labels within the widget configuration that should be used in place of those configured in the linked tag.

Confirmation Dialog check box.

The confirmation dialog check box can be selected if you wish the user to be prompted for confirmation of the new value being set.

Constrain Aspect Ratio.

Prevents the image from being deformed when resized on the screen.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\

Basic Components\Button Controls\

*Related Information:*

...Choices for Operator Interaction

Set Analog Value Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control,
Counter, Selector Switch, Totalizer.
* Does not use the Stye Settings tag.
The Set Analog Value widget is used to create a button that operators
may click in order to write a specific value to an I/O address of the asso-
ciated tag. Several may be drawn in order to provide a range of value
choices. Note that the configured value is not shown as part of the wid-
get – if you would like it to appear (as shown in the following example)
the value must be drawn on the page as text.

Five Set Analog Value widgets, drawn on an image of a fan. The text and
the black backgrounds are not part of this widget.
The properties dialog for the Set Analog Value widget:
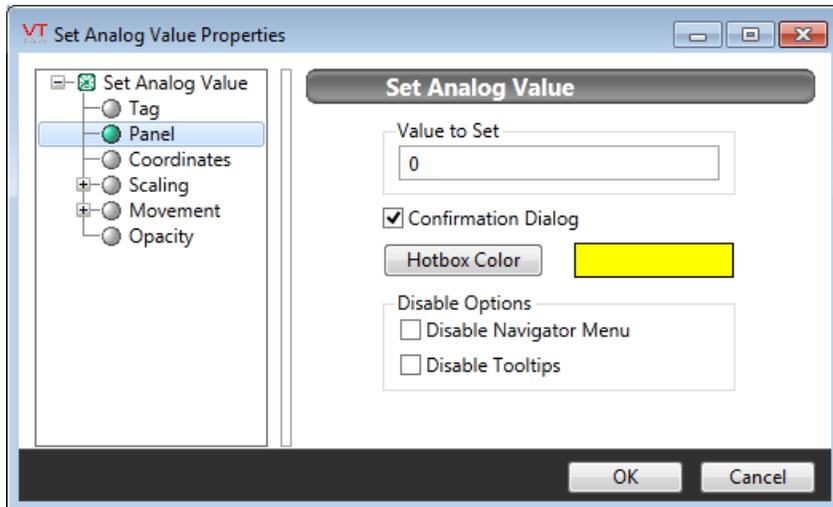
Value to Set

Set the value that will be written when the completed object is clicked. If the associated tag is an analog output, this may be any value that is legal for that tag. Otherwise, this value should match one of the configured states for the associated selector switch.
If this value is not set, the button will not work.

Query Changes

If set, the operator will need to confirm every value change made by this method.

Hotbox Color

Opens the Select Color Dialog from which you can select a color for the hotbox object. Since yellow hotboxes are often associated with page changes, you may want to select a distinct color.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

...Choices for Operator Interaction

Set Value Button Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control, Digital Input, Digital Status, Digital Output, Digital Control, Multi-Write, Pump Status, Report, Selector Switch, Trigger.
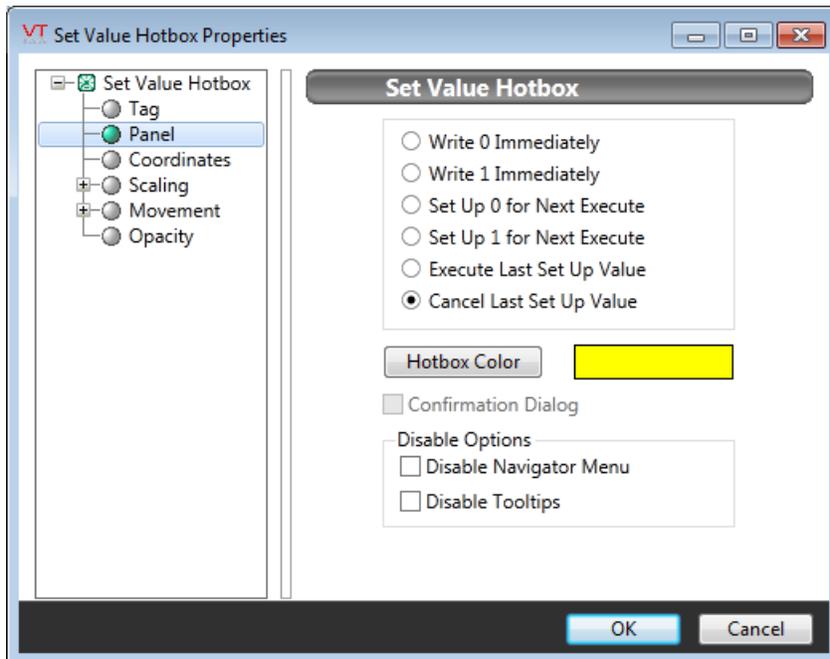* Does not use the Stye Settings tag.
The Set Value Button tag widget is used to create a button that operators can use to write either a 0 or a 1 to an address of an RTU or PLC.

Start Pump
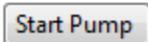
The button can be configured to either write a value immediately, or to set up a value for the next button press. Most commonly, this widget is used with a digital output or control tag and will be drawn twice for that tag: one instance will write a 0 and one instance will write a 1.
If the associated tag is configured to write the value with an x-second pulse, the State 1 label will be visible while the write is in progress. Otherwise, only the State 0 label will normally be seen.
If the label is to include an ampersand (&) and you are not using the legacy style, then the & character will need to be doubled in order to appear.
The properties dialog for the Set Value Button widget will look similar to the following. Note that the confirmation dialog option is available only when writing a 1 or 0 immediately, or when executing the last set up value.
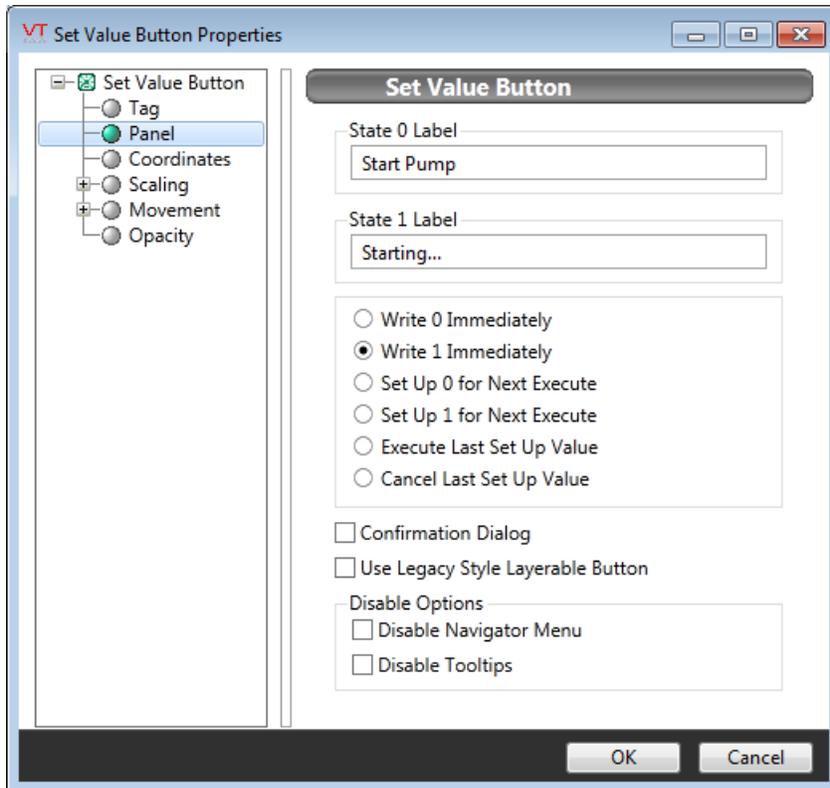
State 0 Label

Set the text that should be displayed on the button when the associated tag is in state 0. In most cases, this will should describe what will happen when the operator presses the button. (for example, "Run" or "Start").
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

State 1 Label

Enter the text that you wish to be displayed on the completed button when the associated tag is in state 1 (or a higher value). This should usually provide feedback to the operator to confirm that the button has been clicked (for example, "Running..." or "Starting...").

If the associated tag is a digital output or digital control and has been configured with a pulse duration (see the tag's Pulse Duration property on the I/O tab of its tag properties folder), the State 1 Label will be displayed for the length of the pulse duration. If the tag being drawn has been configured with a pulse duration of "0," then it is recommended that the State 1 Label match the State 0 Label.

Write 0 Immediately / Write 1 Immediately

Choose one of these radio buttons if you want the completed button to write a value of 0 or 1 to the PLC or RTU when it is clicked.

Set Up 0 For Next Execute / Set up 1 For Next Execute

Choose one of these radio buttons if you wish the completed button to set up a value of 0 or 1 to be sent to the associated equipment.
This will not result in the completed button sending a value to the PLC or RTU when it is clicked; rather, it will latch the button on, set the value in memory and wait for a command to execute this value. A second button should be configured with the Execute Last Set Up Value radio button selected.
When the second button (Execute Last Set Up Value) is clicked, the value in memory will be sent to the associated equipment, and the first button will be latched off.

Execute Last Set Up Value

The Execute Last Set Up Value radio button can be selected if you wish the completed button to transmit the last set up value to the associated equipment.

The Execute Last Set Up Value option requires at least two Set Value Button objects. The first button should be configured with either the Set Up 0 For Next Execute or Set Up 1 For Next Execute radio button selected. When the first button is clicked, it will be latched on, and the associated value (0 or 1) will be set up in preparation for a command to execute. When the second button (Execute Last Set Up Value) is clicked, the value will be transmitted to the equipment, and the first button will be latched off.

Cancel Last Set Up Value

The Cancel Last Set Up Value radio button can be selected if you with the completed button to cancel the last set up value to the associated equipment. A set value button with this option should be drawn whenever you have also drawn one with the Set Up … for Next Execute option set.

Confirmation Dialog

Available only when the button is configured to write a 1 or 0 immediately, or to execute the last set up value. When checked, the operator will be prompted to confirm the control action. The message in the dialog box will be a variation of the following example, depending on the control action selected. See also: Confirmation Prompts for Output Tags.

Use Legacy Style Button

If checked, the button will be shown using an older (resizable)

button style. The updated button will provide a more modern appearance to your pages but, it can only be resized horizontally.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Set Value Hotbox Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control, Digital Input, Digital Status, Digital Output, Digital Control, Multi-Write, Pump Status, Report, Selector Switch, Trigger.
* Does not use the Stye Settings tag.
The Set Value Hotbox tag widget is identical in function to the Set Value Button widget. The difference is only in its appearance in your applications. Where the Set Value Button widget creates a click-able button, the Set Value Hotbox creates only a rectangular hotbox marking an area where the operator can click. It is up to you to make it clear to the operator what will happen when the hotbox is clicked.
One way that this widget might be used is superimpose hotboxes over an image of equipment from your facility. The boxes should be located over the controls familiar to the operators.
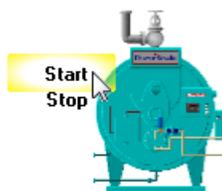
The properties dialog to configure the Set Value Hotbox:

Write 0 Immediately / Write 1 Immediately

Choose one of these radio buttons if you want the completed button to write a value of 0 or 1 to the PLC or RTU when it is clicked.

Set Up 0 For Next Execute / Set up 1 For Next Execute

Choose one of these radio buttons if you wish the completed button to set up a value of 0 or 1 to be sent to the associated equipment.
This will not result in the completed button sending a value to the PLC or RTU when it is clicked; rather, it will latch the button on, set the value in memory and wait for a command to execute this value. A second button should be configured with the Execute Last Set Up Value radio button selected.
When the second button (Execute Last Set Up Value) is clicked, the value in memory will be sent to the associated equipment, and the first button will be latched off.

Execute Last Set Up Value

The Execute Last Set Up Value radio button can be selected if you wish the completed button to transmit the last set up value to the associated equipment.

The Execute Last Set Up Value option requires at least two Set Value Button objects. The first button should be configured with either the Set Up 0 For Next Execute or Set Up 1 For Next Execute radio button selected. When the first button is clicked, it will be latched on, and the associated value (0 or 1) will be set up in preparation for a command to execute. When the second button (Execute Last Set Up Value) is clicked, the value will be transmitted to the equipment, and the first button will be latched off.

Cancel Last Set Up Value

The Cancel Last Set Up Value radio button can be selected if you with the completed button to cancel the last set up value to the associated equipment. A set value button with this option should be drawn whenever you have also drawn one with the Set Up … for Next Execute option set.

Hotbox Color

Opens the Select Color Dialog from which you can select a color to be displayed when the mouse pointer is within the area of the hotbox. To reduce confusion, you should consider making this a different color than that of the Page Hotbox.

Confirmation Dialog

Available only when the button is configured to write a 1 or 0 immediately, or to execute the last set up value. When checked, the operator will be prompted to confirm the control

action. The message in the dialog box will be a variation of the following example, depending on the control action selected. . See also: Confirmation Prompts for Output Tags.
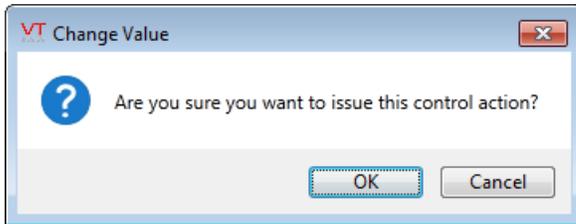
Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\Basic Controls\
Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

Show Stats Widget (ODBC)

Used by the ODBC Manager.
* Does not use the Stye Settings tag.
The Show Stats widget is used to place a button that will open the ODBC Statistics dialog. The button will be rectangular and will use your system color and font.
The only user-configurable option is to change the label that appears on the button.
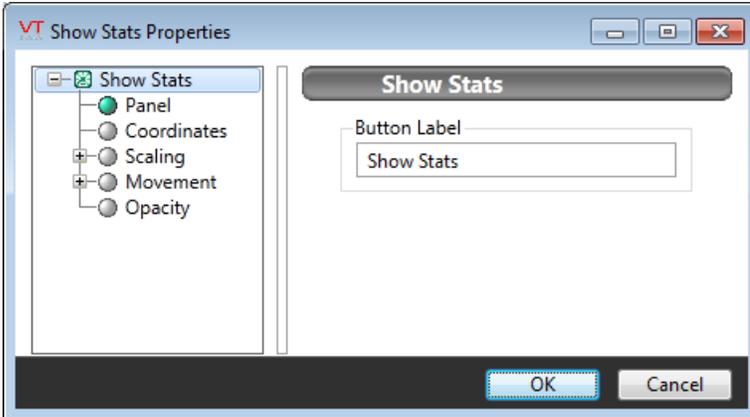
Widget Folders:

Analytics\Statistics\

Tools\ODBC Manager\

Related Information:

...Monitor the ODBC Interface – For further information on the ODBC Statistics Dialog.

Site Alarm List

Used by: Context, Site tags

Use the Site Alarm List widget to draw an alarm list that is pre-filtered to show only the alarms in child tags of the linked site or context.
It can also be configured to show all alarms in any tag having an Area property that matches the linked site. Or, it can filter for both child tags of the site, and all tags having a matching Area property.
An acknowledge button will be available for each entry, provided that the logged-on operator has the security privilege required to acknowledge these alarms.

If fields overlap, select a smaller font in the properties dialog.

The properties dialog for the Site Alarm List widget:
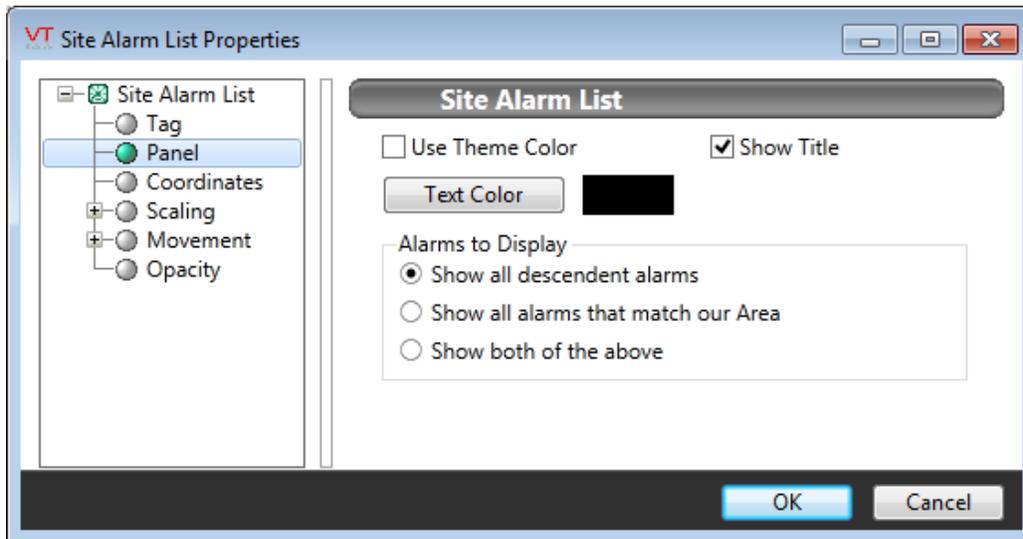
Use Theme Color

Choose whether to match the list to the application's theme coloring.

Text Color

Alarms will be shown against a background of the color configured for each alarm priority. Selected alarms are shown against a gray background. For each of these, a dark font tends to be most easily seen.

Show Title

Choose whether or not to display the title, "Alarms" above the list. The text of the title is not user-configurable.

Alarms to Display

Use these check boxes to control how the filter is applied. You can choose to display:

- Only alarms for child tags of this site or context.
- Only alarms whose area property matches the linked site or con-

text.

- Both of the above.

Site Details Widget

Can be drawn directly by the Context tag. Otherwise, the Site Details page is the default destination of the Site Summary and Site Draw widgets.
* Does not use the Stye Settings tag.

Related Information:
Site Details Page – Shows communication, alarm and I/O information for site and station tags.
Configuration options include the following:
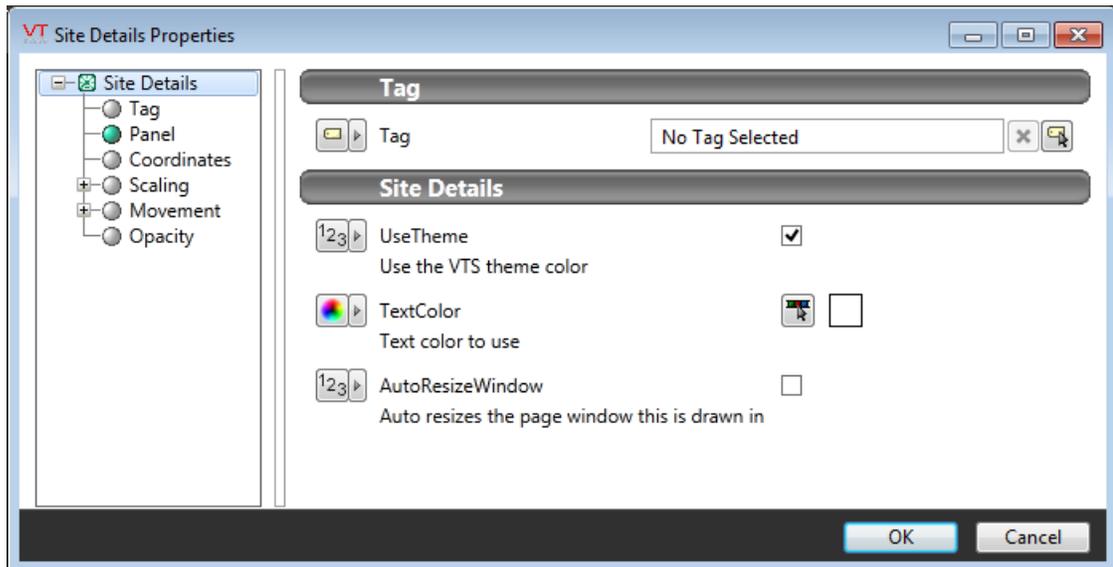
Use Theme

If checked, the color theme of the application will be applied to the Site Details page.

Text Color

Must be numeric, corresponding to any of the colors in the VTScada palette. Used for text within the Site Details page,

including labels of I/O tags. Defaults to 15, which is white.

AutoResizeWindow

When checked, the page containing this widget will auto-matically adjust upon opening to show only the area of the widget, plus any whitespace above or to the left.

> Note: This option should be used only when the Site Details widget is drawn on a pop-up window. Unexpected behavior will result when used in the application's main window.

Widget Folders:
Equipment\Lift Stations\
Tag Types\Organization/Containers\Context\

Site Draw Widget

Used by: DataFlow RTU, Polling Driver.
* Does not use the Stye Settings tag.
The Site Draw tag widget is used to draw station symbols that represent RTUs These display the status at the station being drawn, using two, concentric color indicators. The Site Draw may be configured to display a label or to show only the indicator circles.
A Site Draw legend is available in the Site Tools folder of the Widgets palette. Adding this to your page will help operators interpret the meaning of the color indicators.

Site Draw symbols, when clicked, will open an automatically-generated page containing details about the RTU and I/O tags attached to that

station. They can also be configured to open a page of your choice instead.

Note that the label is not considered part of the object's area when trying to select the object to be moved or edited. You must click on the circles in order to select the object.
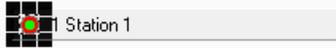
The properties dialog for the Site Draw widget:
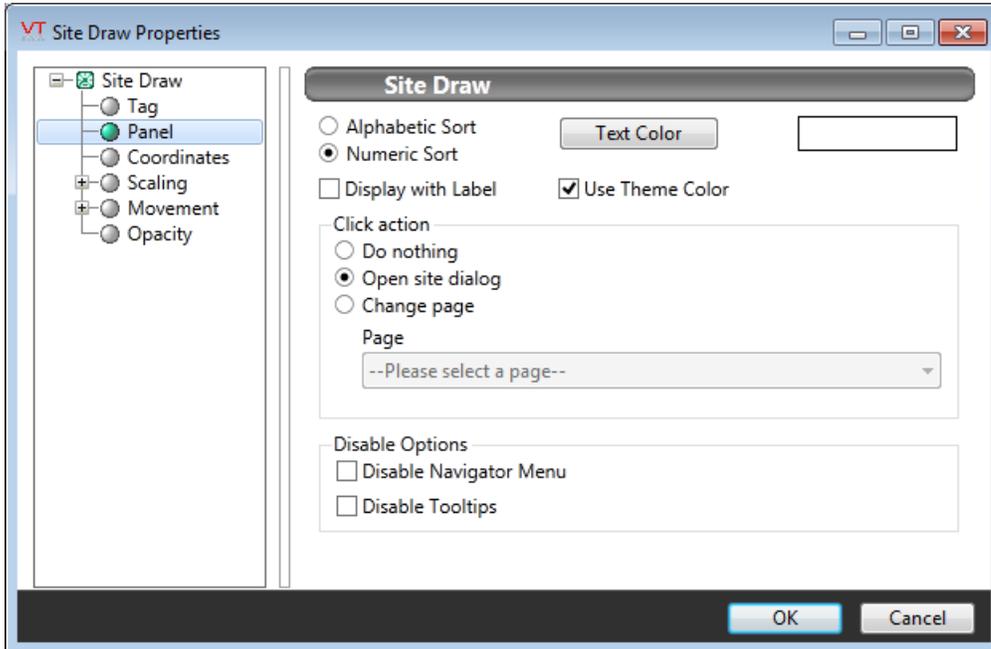
Sort Order (Alphabetical or Numeric)

Affects the display of the automatically created site dialog. The input and output tags associated with the station being drawn will be included in this dialog in either alphabetic or numeric order, by their description.

If an order has otherwise been specified by each input and output tag's Order property (on the tag's Order tab), the Sort Order will be ignored.

Display with Label

Select this option if you wish the completed station symbol to be drawn with a label showing the description of the tag being drawn.

Text Color

Select a color for the labels within the site draw page.

Use Theme Color

If selected, the site draw page will use the theme colors of your application. Otherwise, the standard colors of gray and black will be used.

Click Action

The usual action upon clicking a Site Draw dot is for the site dialog to open. This dialog-type page is automatically drawn by VTScada. As an alternative, you may choose to have no action occur, or you may select any page that you have created for the site. If using your own page instead of the site dialog, you should consider parameterizing that page so that it may display the tags relevant to the given server. See: Page Properties Parameters Tab.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Specialty\Mapping\
Tag Types\I/O Devices (Drivers)\PLCs and RTUs\Data Flow System RTU\
Tag Types\Communications\Link Management\Polling Driver\

Related Information:

...Choices for Operator Interaction

Site Icon Widget

Used by: MultiSmart Station, MPE Duplexer, MPE SC, Context tag, ScadaAce site tags.

* Does not use the Stye Settings tag.

The Site Icon Widget creates a color indicator that both displays the status of communications to the associated station and provides a click-able link to the built-in Station Page for the associated station tag. This is very similar to the Site Draw widget, and uses the same color system for its concentric rings.

The Properties dialog for the Site Icon widget:
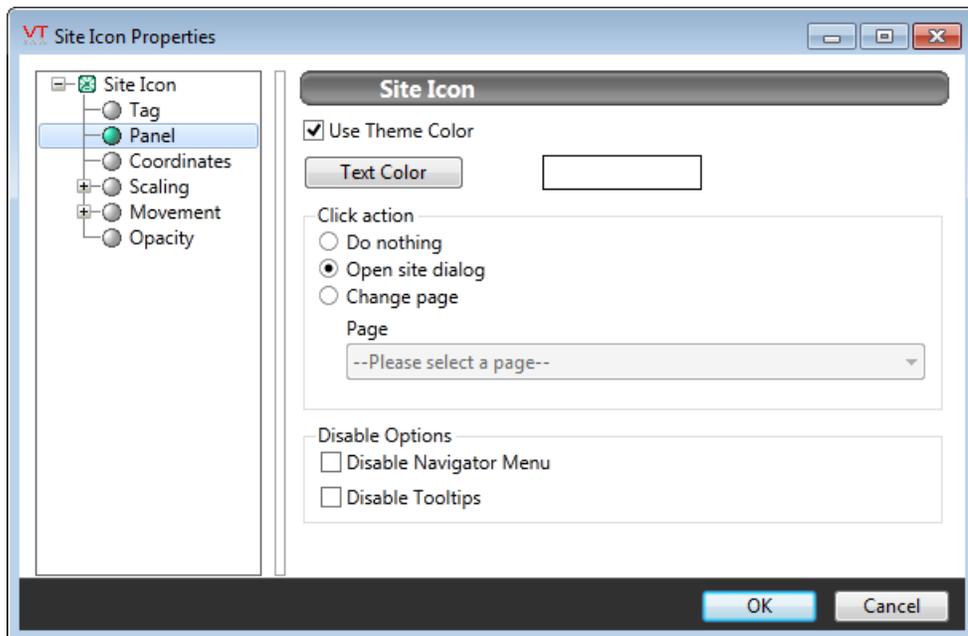
Use Theme

When selected, the station page that opens when the operator clicks on this icon will use the color theme configured for the application. Defaults to gray if not selected.

Text Color

Use to change the color of all text used in the associated Station Page. Text defaults to white if left unselected. Click to open the Select Color dialog.

Click Action

The usual action upon clicking a Site Icon dot is for the site dialog to open. This dialog-type page is built into your application. As an alternative, you may choose to have no action occur, or you may select any page that you have created for the site. If using your own page instead of the site dialog, you should consider parameterizing that page so that it may display the tags relevant to the given server. See: Page Properties Parameters Tab.

Disable Options

Disable selected operator-interaction features of this widget.

The Site Legend provides a guide to the meaning of each color used in the icon. The Site Legend can be accessed via the Site Tools folder as follows:

1. Open the Site Tools folder of the Widgets palette.
2. Drag the Site Legend onto the open page in the Idea Studio.
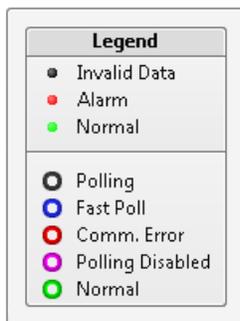
Widget Folders:

Equipment\Lift Station\
Basic Components\Specialty\Mapping\
Tag Types\I/O Devices (Drivers)\PLCs and RTUs\Data Flow System RTU\

Tag Types\Pump Station Controllers\MPE\MPE Duplexer Station\

Tag Types\Pump Station Controllers\MPE\MPE SC Station\

Tag Types\Pump Station Controllers\Multitrode\MultiSmart Station\

Tag Types\Organization/Containers\Context

Related Information:

...Choices for Operator Interaction

Add a Site Map to a Page

VTScada has built-in links to site maps, both within the menu, and within Sites Pages and Site Details pages. These are usually sufficient for most applications, but if you wish, you may add a Site Map to any page.

To add a site map to a page:

1. Open the Idea Studio and the page to which you intend to add a map.
2. Open the Site Tools palette (found within the Widgets palette section).
3. Locate the Site Map widget and drag it onto the page.
4. Provide parameters as appropriate. Parameters are described later in this topic.

Note that the map is drawn within its own window, which will have a higher z-order than the page it is being placed on. The result can be loss of part of the VTScada header or footer if the map is larger than the page.

The properties dialog options for a Site Map drawn this way, are as follows:

Parent

The station tag instance (and any child tags that also have valid latitude and longitude values) that is to be the primary target of this site map. All sites shown will be marked by a pin, but the root site will be further indicated by an animated dot.

Show All Sites

Choose whether this map instance should show only the root site, or if all sites visible in the map area should be shown. In either case, the pin marking the root site for the map is the only one highlighted by an animated bubble.

Use Theme

Choose whether the borders of the map should follow the VTScada color theme

Text Color

Specify the color to be used for the labels below the map.

Widget Folders:
Basic Components\Specialty\Mapping\
Tools\Site Tool Library\

Site Summary Widget

Used by: MultiSmart Station, MPE Duplexer, MPE SC, Context tag, ScadaAce Sites.
* Does not use the Stye Settings tag.
The Site Summary Widget creates a color indicator that both displays the status of communications to the associated station and provides a click-able link to the built-in Station Page for the associated station tag. It also provides relevant information about the underlying station tag. The full display mode option is shown here:

The Site Legend provides a guide to the meaning of each color used in the icon.

1. Open the Widgets palette of the Idea Studio.

2. Open the folder for any of the widgets that use this widget.

3. Drag the Site Summary onto the current page of the Idea Studio.

4. Link the widget to the appropriate tag.

5. Optionally, adjust the display properties.

Configurable properties available in the VGE for the Site Summary widget:
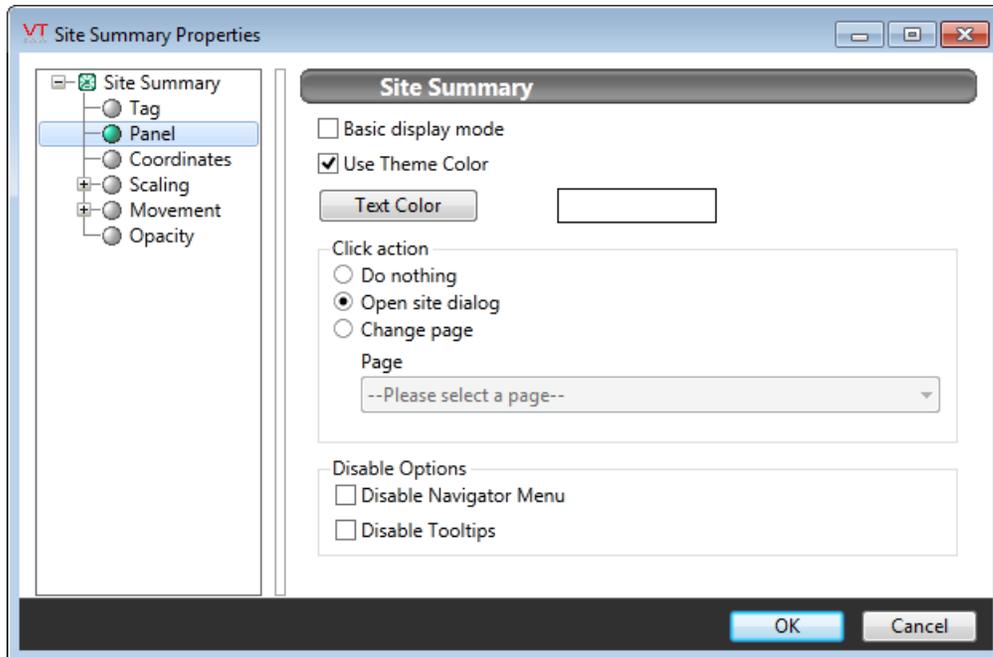
Basic Display Mode

When selected, only the site name will be shown on the label. Otherwise, the name, description and selected communication properties will be displayed.

Use Theme Color

When selected, the station page that opens when the operator clicks on this icon will use the color theme configured for the application. Defaults to gray if not selected.

Text Color

Use to change the color of all text used in the associated Station Page. Text defaults to white if left unselected. Click to open the Select Color dialog.

Click Action

The usual action upon clicking a Site Icon dot is for the site dialog to open. This dialog-type page is built into your application. As an alternative, you may choose to have no action occur, or you may select any page that you have created for the site. If using your own page instead of the site dialog, you should consider parameterizing that page so that it may display the tags relevant to the given server. See: Page Properties Parameters Tab.

Disable Options

Disable selected operator-interaction features of this widget.

Note: You may wish to add a legend to the page so that operators can understand what the color codes of the dot mean. The legend can be found in the Site Tools folder of the Widgets palette.
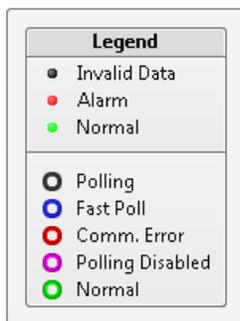
Widget Folders:
Equipment\Lift Stations\
Tag Types\I/O Devices (Drivers)\PLCs and RTUs\Data Flow System RTU\
Tag Types\Communications\Link Management\Polling Driver\
Tag Types\Pump Station Controllers\MPE\MPE Duplexer Station\
Tag Types\Pump Station Controllers\MPE\MPE SC Station\

Tag Types\Pump Station Controllers\Multitrode\MultiSmart Station\
Tag Types\Organization/Containers\Context

Related Information:

...Choices for Operator Interaction

Slider Widget

Used by: Analog Input, Analog Status, Analog Output, Analog Control,
Selector Switch.
* Does not use the Stye Settings tag.
The Slider widget is used to create an output control. An operator can drag the slider in order to set a value which will be output by the associated tag.

In the Design Studio, the range of motion is shown by a line. You will need to provide an image to provide scale for operators.
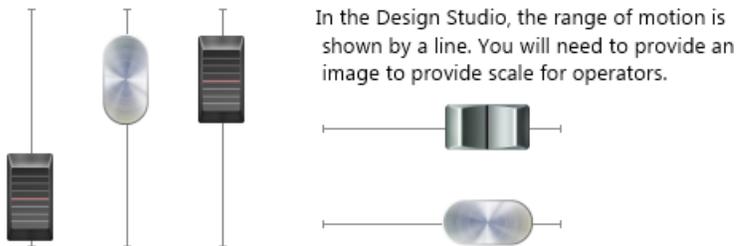
Any image may be used for the slider image, although images from the Sliders group or the Arrows group of VTScada images are recommended. The slider may be drawn vertically or horizontally and can be configured with the maximum value at either end of the range.
Feedback is provided to the operator while moving the slider, via a tooltip window. You also have the option of querying the operator for verification before any change is saved. If the slider is used to draw a Selector Switch, or an Analog Input or Analog Status tag, configured to write data, then a mismatch between the current value and the last value read will indicated by a blinking image.
To make the slider intuitive to use, it is recommended that it be placed on or near an instrument image that clearly shows the available range of motion, and provides scale markers for reference.
The properties dialog for the Slider widget:
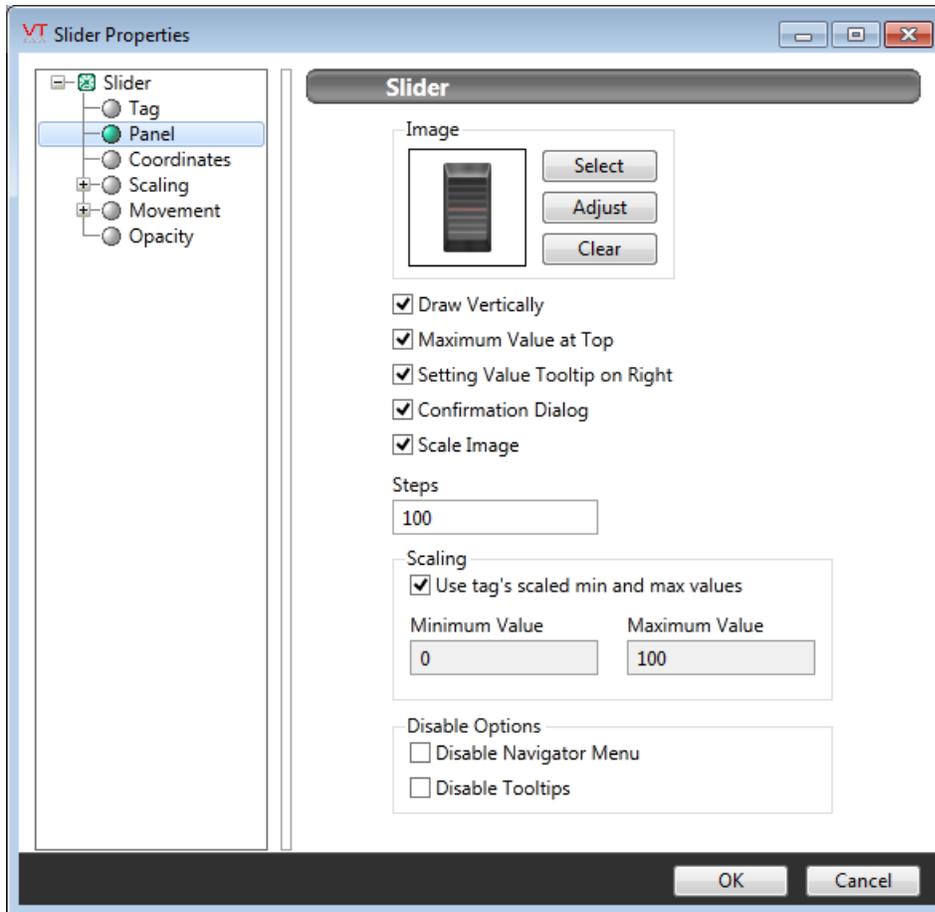
Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Draw Vertically

Select this option if you want the completed Slider object to operate vertically. If you would prefer the Slider to operate horizontally, leave this box unchecked.

Maximum Value at Top

When selected, moving the slider up (or to the right if Draw Vertically is not selected) will result in an increase of the value to write. Otherwise, values will increase towards the bottom or left as the slider moves.

Setting Value Tooltip On Right

When selected, a tool tip displaying the value to which the slider is being set, appears on the right of a vertical slider or above a horizontal slider. When not selected, the tooltip appears to the left of the vertical slider, below the horizontal slider.
This check box will be labeled Setting Value Tooltip On Top if the Draw Vertically check box is not selected.

Query Changes

If selected, the operator to be prompted for confirmation of any new value being set.

Scale Image

The Scale Image check box can be selected if you want the slider image to scale along with the overall widget. If not selected, the image will remain the same size and shape as the size of the overall slider is adjusted.

Steps

Sets the precision of the control, relative to the scaling value.

The default value enables the operator to select 100 possible values within the scale range. Note that operators may have difficulty selecting a value if you set a large number of steps in a short slider.

Scaling Options

Choose whether to use the tag's minimum and maximum scaling values or to set alternative scaling values that will be applied to the output value.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analog Controls\

Related Information:

...Choices for Operator Interaction

SMS Indicator Widget

Used by SMS Appliance tags.
* Does not use the Stye Settings tag.
The SMS Modem Indicator widget displays a graphic of an active SMS modem control, showing the name of the associated SMS Appliance tag. It is similar to the Modem Indicator Widget, but displays SMS events rather than a continuous indication of the modem status.
The indictor will display the color red and the target cellular number when a message is being sent.
When a message is being received, the indicator will display the color green and the cellular number of the device sending the message.
Other messages that may be displayed by the indicator include "Initializing," "Disabled," and "No Workstation".

A modem can be reset by clicking on the widget when the application is in operation mode. A modem can also be marked as failed or unavailable using the Disable option of the shortcut menu (right-click on the widget when the application is in operation mode and select the Disable option).
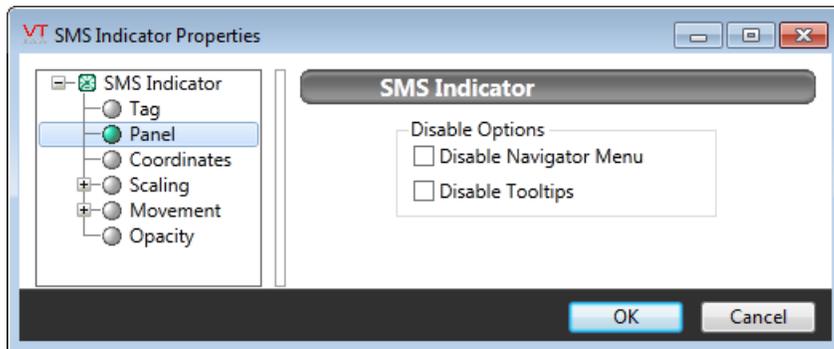
Disable Options

      Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Communications\Connection Status\
Analytics\Diagnostics\
Tag Types\Communications\Alarm Notification\SMS Appliance\

Related Information:
Choices for Operator Interaction

Equipment / Status Color Indicator

* Can be linked to a Stye Settings tag.
Used by: Alarm, Alarm Status, Analog Control, Analog Input, Analog Output, Analog Status, Calculation, Counter, Deadband, Digital Control, Digital Input, Digital Control, Digital Status, Function, History Statistics, Modem, Network Status, Pump Status, Rate of Change, Roster, Selector Switch, Serial Port, SMS Appliance, SQL Logger, TCP/IP Port, Totalizer, Trigger, UDP/IP Port,
Status color indicators can be found in the Equipment folder of the Widgets palette. These are images of equipment that take their color from a Style Settings tag, based on the tag to which they are linked.

The properties dialog for a Status Color Indicator widget:

Style

Optionally, select a Style Settings tag to control the colors shown by the light, in response to the linked tag's value and alarm state.

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators\

Related Information:

...Style Settings Tags

...Choices for Operator Interaction

String Droplist Widget

Used by: Analog Input, Analog Status, String I/O Tag

The String Droplist widget can be used to write one of a list of preset values to an output device. The write occurs when a value is selected in the list. Use of the confirmation dialog is suggested.
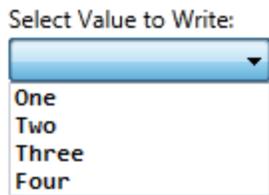
The properties dialog for the String Droplist widget:
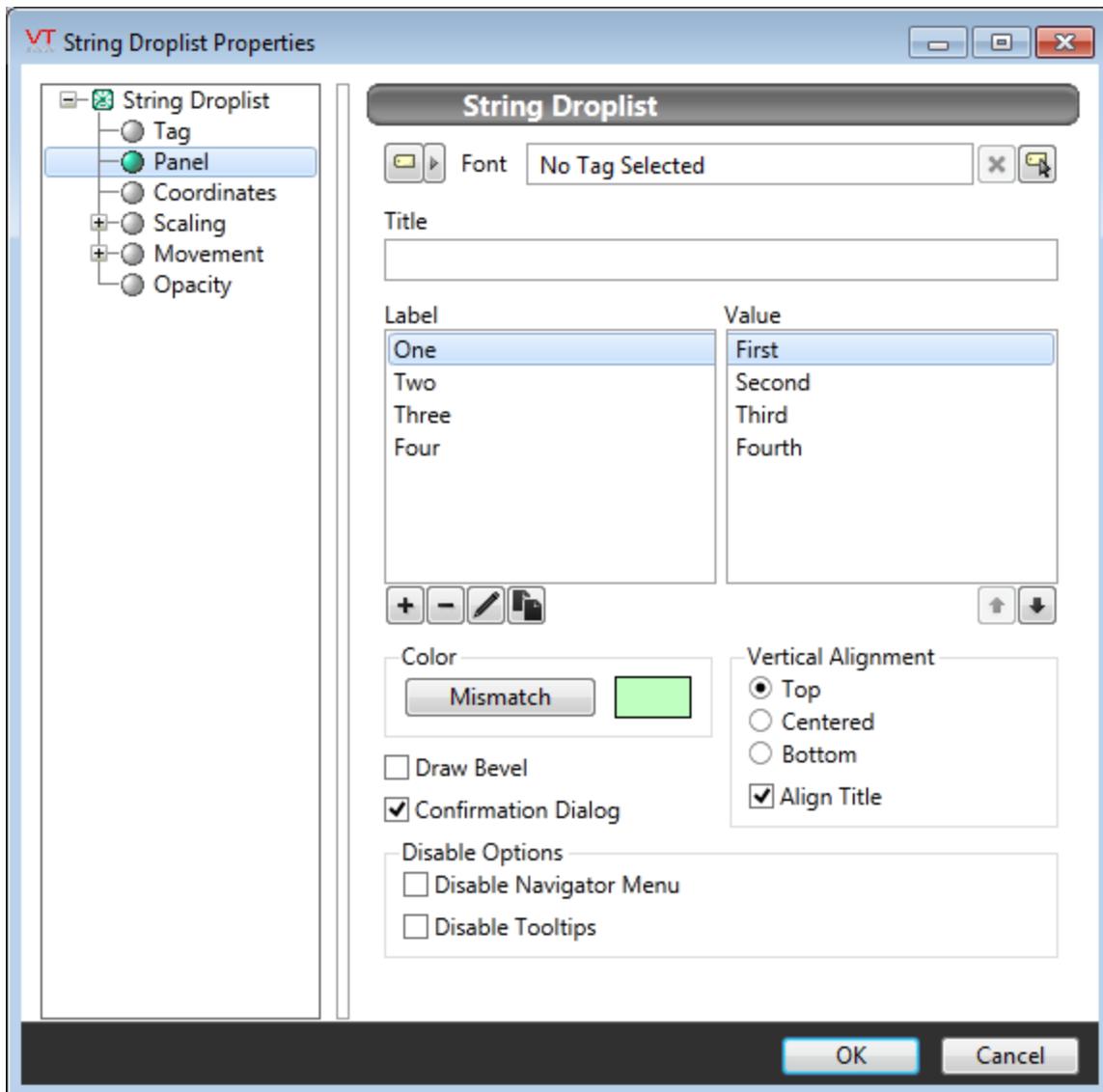
Font

Controls the display characteristics of the data entry field, but not the label. If you set individual characteristics using the Format ribbon, this will be an expression, combining those characteristics. Otherwise, select a font tag.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to

Title

Provide text that will tell the operator what to use this String Droplist field for.

Label and Value

The list of possible output values, and the label that is associated with each value. Values may be text or numeric, but must be constant. You cannot use tags or expressions for the values.
Use the buttons below the list to create or edit the values and labels. You can also double-click an entry to open it for editing. When creating a list of similar values, you may find it easier to copy and edit entries than to create each individually.
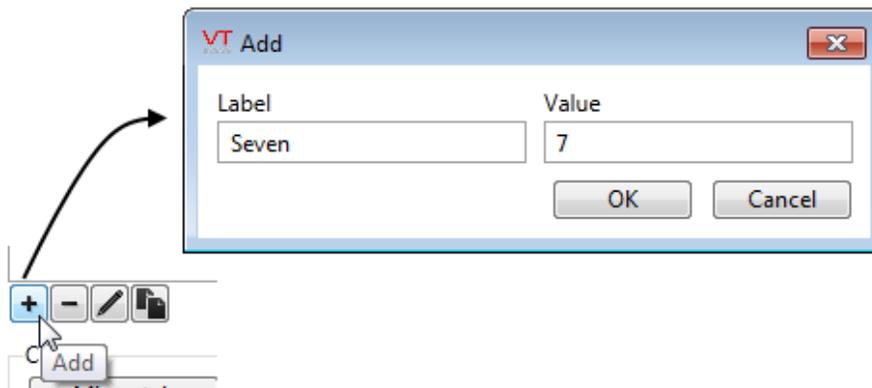
Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom.

Align Title

Choose whether the title should be included in the vertical alignment calculation. For example, if vertical alignment is top-aligned and the align title option is not selected then the title will extend above the widget's bounding box.

Mismatch Color

If the driver supports read and write operations at the same address, then when the last value written does not match the value being read, the widget will display the configured mismatch color.

Draw Bevel

Select whether or not the completed object should be drawn with a beveled border. If the Draw Bevel check box is selected, the completed object will be drawn with a border surrounding it.

Confirmation Dialog

When checked, the operator will be prompted to confirm the control action.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
String Entry Widget
Choices for Operator Interaction

String Entry Widget

Used by: Analog Input, Analog Status, String I/O Tag

The String Entry widget are used with String I/O tags that are configured to write values. Operators can enter the value to be written in the data entry field. The write operation will happen when focus leaves the field, whether by an Enter, a tab, or a click on some other widget. Characters can be masked for privacy.

The example is configured to show the optional title and bevel.

Value to write:
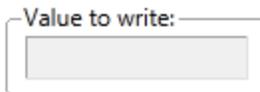
The properties dialog for the String Entry widget:
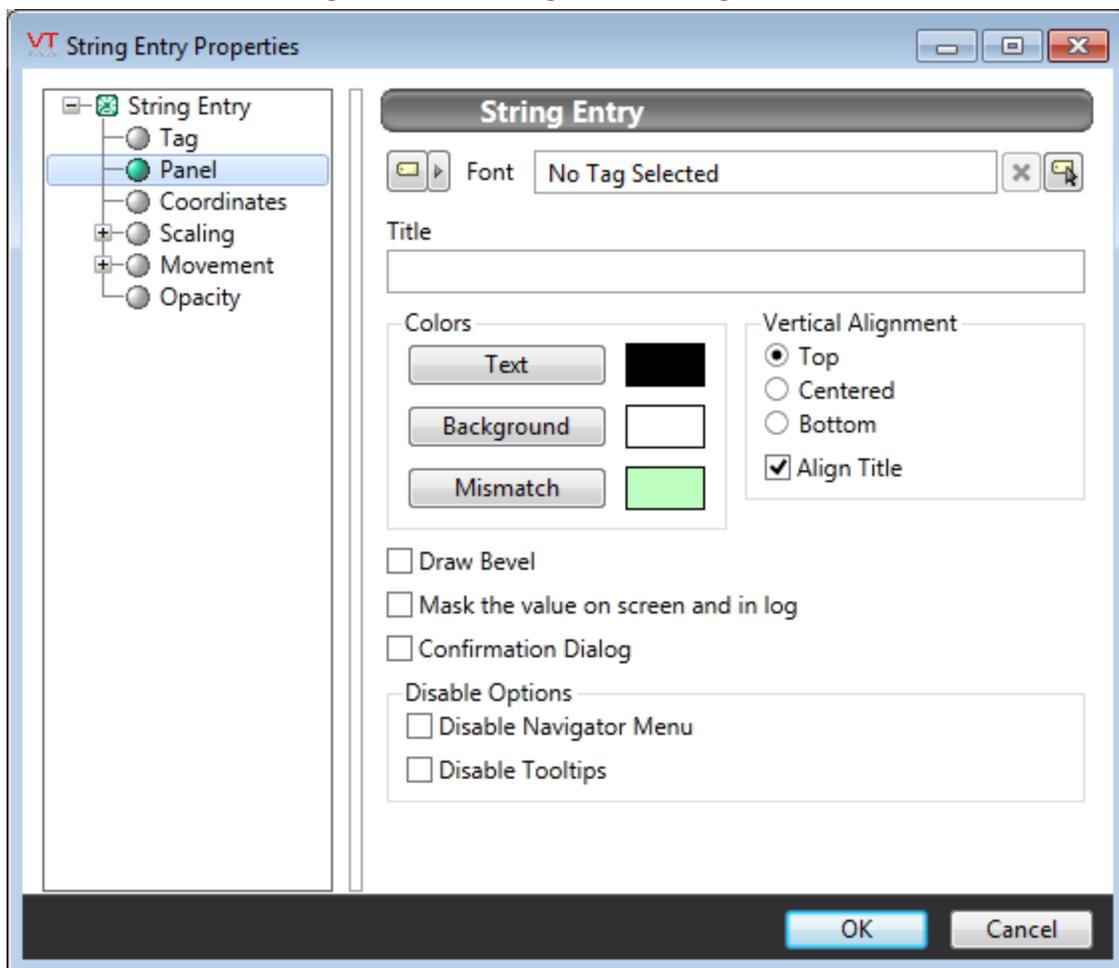
Font

Controls the display characteristics of the data entry field, but

not the label. If you set individual characteristics using the Format ribbon, this will be an expression, combining those characteristics. Otherwise, select a font tag.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Title

Provide text that will tell the operator what to use this String Entry field for.

Vertical Alignment

Locates the object vertically within its drawing area. May be one of Top, Centered or Bottom.

Align Title

Choose whether the title should be included in the vertical alignment calculation. For example, if vertical alignment is top-aligned and the align title option is not selected then the title will extend above the widget's bounding box.

Text Color

Select the color to be used for the input text. Does not apply to the label.

Background Color

Select the color to be used for the background of the input

field. Does not apply to the label.

Mismatch Color

If the driver supports read and write operations at the same address, then when the last value written does not match the value being read, the widget will display the configured mismatch color.

Draw Bevel

Select whether or not the completed object should be drawn with a beveled border. If the Draw Bevel check box is selected, the completed object will be drawn with a border surrounding it.

Mask Value

When checked, the value will be masked by asterisks rather than being displayed on the screen or in the event log. This option will most often be selected when the String Entry is being used to write a password to a hardware device, thus preventing other users from discovering the password.
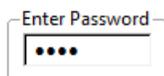
Confirmation Dialog

When checked, the operator will be prompted to confirm the control action.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:
String Droplist Widget

Numeric Entry Widget
Choices for Operator Interaction

Tag List Widget

Used by: Context Tag, Analog Statistics tag, Digital Statistics tag, Station tags. Optionally, by user-created tag types.
* Does not use the Style Settings tag.
The Tag List widget provides a highly configurable means of displaying any or all of the child tags attached to a selected parent. If no parent tag is selected, then root of the tag hierarchy is considered to be the parent and any set of tags in the application can be displayed in the list, subject to the configuration of the filtering options.

Note: The Analog Statistics tag is composed of a number of History Statistics child-tags. The Digital Statistics tag is composed of Counter, History Statistics and Totalizer child tags. All of these child types hold numeric (analog) values.

Use this information in the Tag List configuration when selecting the widget and the types to display using that widget.

The Tag List will display each type of child tag using any of the standard widgets available for that type. For example, the Analog Status tags in the list may be displayed using a Meter4 widget while the Pump Status tags are shown using an Image Change widget.
The Tag List can be filtered and sorted to select and organize which tags will be included. Also, the list itself may be configured to control where the widgets are placed within it and how it will handle multiple pages when there are too many tags for the configured display area.
The steps shown here outline the general procedure to configure a Tag List Widget. Detailed steps for each part of the procedure are provided in following topics.

1. Determine the type of the parent tag for which you want to draw a Tag List.
2. Open the Widgets palette within the Idea Studio.

3. Open the folder for the selected parent tag type.

4. Drag a Tag List from the palette onto the current page.

5. Link the Tag List to the parent tag.

6. Open its Properties dialog.
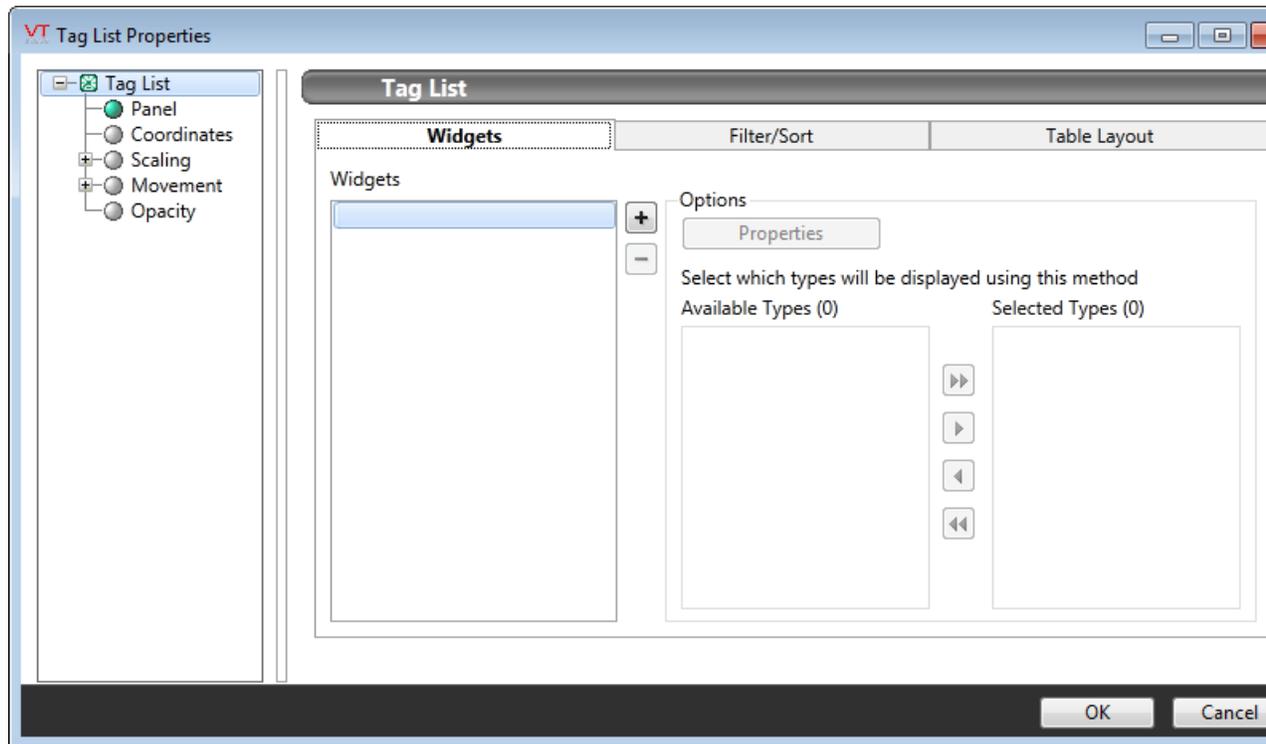
7. For each of the tag types to be included in the display, select the tag type, then select which widget will be used to display that type in this list.
Note that you are not limited to types for which tag instances exist. You can configure the Tag List so that tags that are added later and that match the selection filter will be displayed automatically.

3. [Optional] Many widgets can be used by more than one type of tag. If other tag types are to use the same widget chosen in step 2, select those types from the Available Tag Types list and click the arrow button to move them to the Selected Tag Types list.

4. [Optional] Adjust the display options of the selected widget by clicking the Properties button.

5. For the remaining tag types that you wish to include in the Tag List, repeat steps 2 through 4.

6. Click the Filter/Sort tab.

7. [Optional] Restrict which tags in the structure will be included in the list by using the Filtering options.

8. [Optional] Sort the tags shown in the Tag List display.

9. Click the Table Layout tab.

10. Adjust the appearance of Tag List as required.

11. Click on OK.

Widget Folders:

Basic Components\Multi-Variable\

Tools\Standard Library\

Tools\Report Tools\

Tag Types\Organization/Containers\Context

Related Tasks:

Access the Tag List Widget

If you select a Context tag to draw, the Tag List is the default widget.

1. Select a Context tag in the Tag Browser.

2. Click Draw, or right-click on the tag and select Draw.

3. Select the Tag List.

4. Click on the page to locate the lower-right corner of the list.

A Tag List can be drawn for any parent tag, or for all tags in the application.

1. Open the Idea Studio.

2. Access the Widgets palette.

3. Expand the Tools folder, then the Standard Library folder.

4. Drag the Tag List to the page.

5. Link to a parent tag, or open the Properties dialog to select a tag type and a list of widgets.

Next task: Select widgets for the list.

Select Widgets for the List

1. Click the plus sign beside the Draw Methods list.
   The Widget Selection dialog opens.
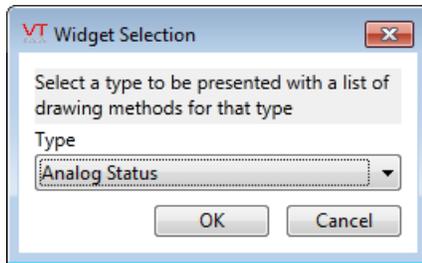
A widget is chosen by first selecting the tag type to be drawn. With that information, VTScada can provide a list of the available widgets for the chosen tag type.

2. Select the appropriate tag type from the drop-down list.
   This should be the next tag type that you intend to display in the Tag List widget.
   A list of the available widgets for that tag type will be displayed. Only one method may be used in the Tag List for each tag type.

3. Select the appropriate widget.
   The Widget Parameters dialog will open. This is a condensed version of the properties dialog that would normally be displayed when using the selected widget.

4. Set the display parameters that will be used by all instances of this widget in the Tag List display.
   At a minimum, this will include the standard three operator controls: Disable Trend, Disable Navigation or Disable Tooltip. Leave as zero if you want each widget in the list to respond normally to the pointer hovering or clicking.
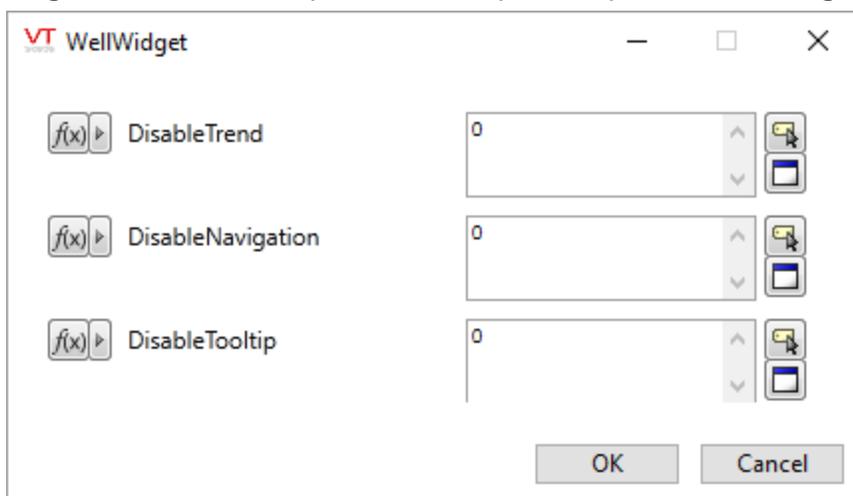
5. Click OK.
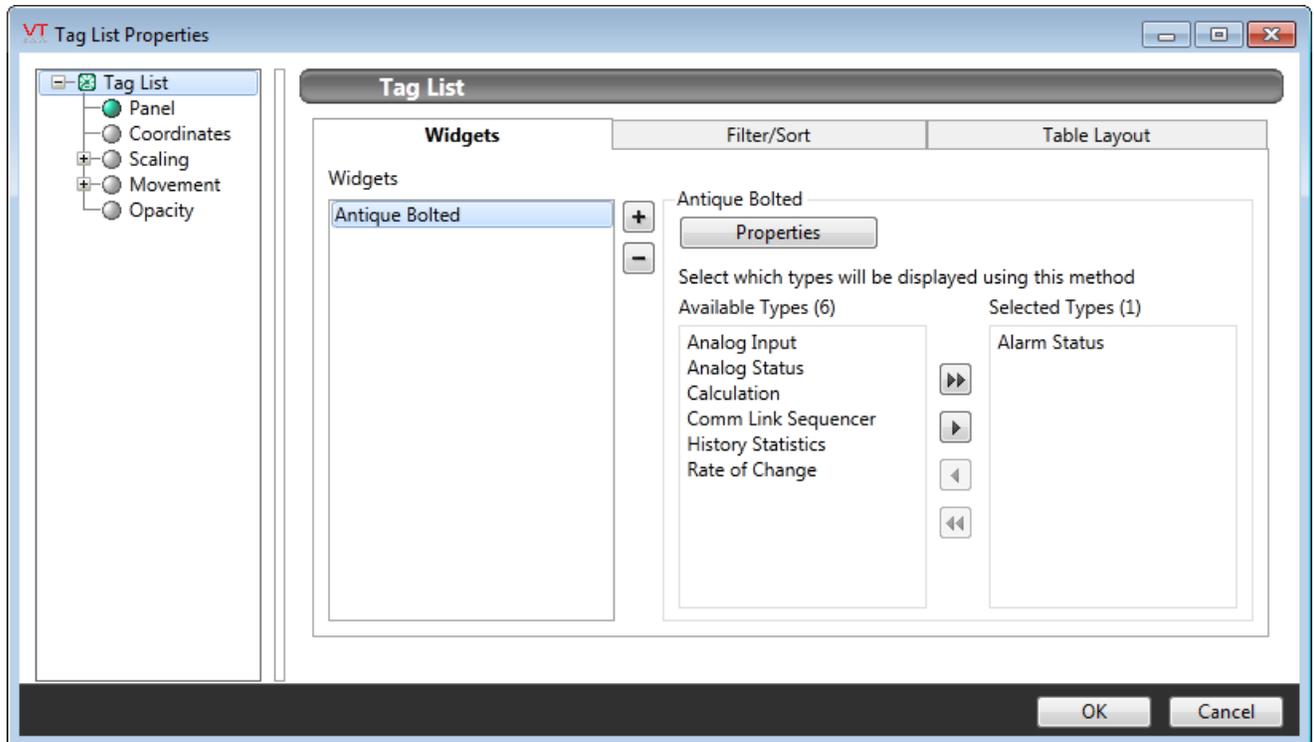
The tag type that you used to select the widget will be displayed using this method in the Tag List. This is the tag type shown in the Associated Tag Types list.

It is possible that other tag types could also be displayed using this widget. These are the items in the Tag Types Available list.

Next task: Filter and sort the display.

Filter and Sort the Tag List Display

Unless otherwise filtered, all the tags in the parent-child tag structure, that have had a widget selected for their tag type, will be included in the list. You may use the filter options to limit the list to only those tags that are of interest to you.

The first filtering option is "Immediate Children Option". This is checked by default, restricting the display to only those tags one level below the selected parent.

If deselected, then all the tags in the tag structure (that have an associated widget from the previous tab and that are not otherwise excluded by the configuration of the filter) will be included in the Tag List.

Name Filter:

Using the wildcard character, *, enter a pattern to restrict the Tag List to only those tags whose Name property matches the pattern.
Example: "Pump*"   – all tags whose name begins with the letters "Pump".

Area Filter:

Using the wildcard character, *, enter a pattern to restrict the Tag List to only those tags whose Area property matches the pattern.
Example: "*zone" – all tags whose area property ends with the letters "zone".

Expression Filter:

Using the VTScada Expression language, you can create a filter that restricts the Tag List based on tag property or combination of properties. For example: \Units == "FT". This will limit the selection to those tags whose engineering units are "FT".
Note that, if filtering for a value you must use \Root\Value. All other properties can be referenced by simply \PropertyName, but \Value must be preceded by \Root.

Sort the Tag List Display
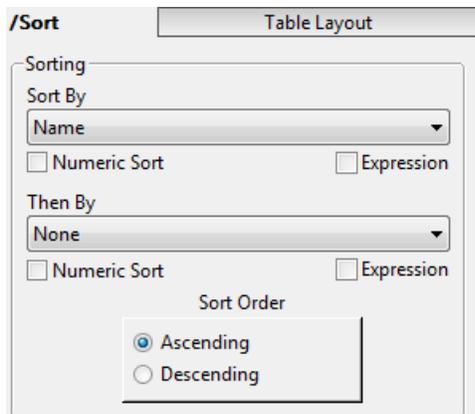
By default, the items in the Tag List will be sorted by name. Using the controls shown here, you may sort the list by other parameters, and may also perform a sub-sort. For example, you might sort the tags by area, then within each area sort by name.

You can also choose whether to sort the list alphabetically or numerically, and in ascending or descending order.

The available fields for sorting are Name (tag name), Area, Tag Type and Draw Method. You can also create an expression that returns a string or a number, in order to sort based on any tag property. Examples of expressions for sorting include, \Value, \Description, and \Units.

Next task: Configure the tag list layout.

Configure the Tag List Layout
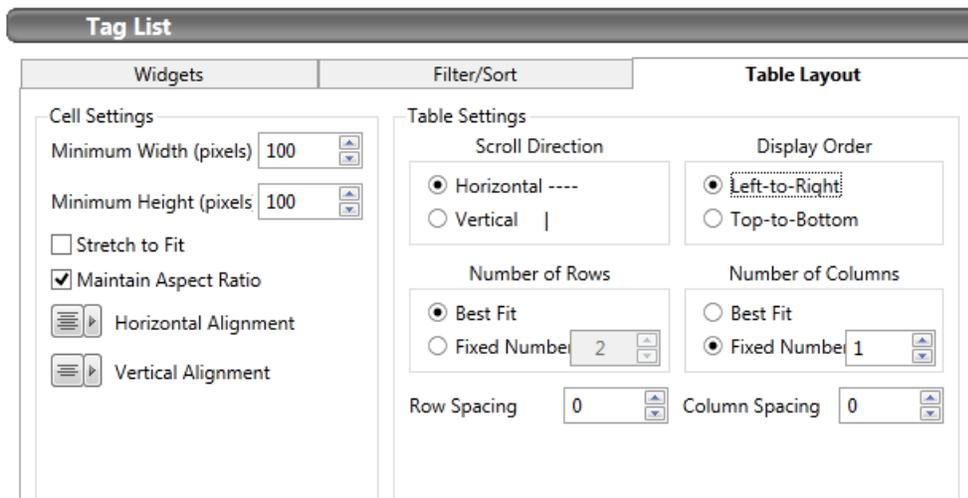
Cell Settings

Each tag in the list, displayed using the selected widget, counts as one cell.
You may set the following options for the cells:

Minimum Width and Height

Measured in pixels. Sets the smallest size that each cell may be. Cells may be as large as required. A widget will not necessarily fill its cell unless you have selected "Stretch to fit".

Stretch to fit.

The widget in each cell will be scaled to fit the cell size.

Maintain Aspect Ratio.

If stretched, determines whether the widget will be stretched equally in width and height.

Horizontal and Vertical Alignment options.

Sets the widget's position within its cell, in the event that the widget does not fill the cell. The default options are to center the cells within the Tag List, both horizontally and vertically. Some users may prefer a table that is left-aligned with entries that start at the top, filling downwards.

Table Settings

Scroll direction

If there are more tags to display than will fit within

the area of the Tag List, a scroll bar will be added and the operator can page through the tags. This option controls whether the list scrolls from top to bottom or from left to right.

Display Order

For the tags that fit within the Tag List's area, the order of display – left to right or top to bottom, are controlled by this option. This option is affected by the sorting order selected in the previous tab.

Number of Rows and Number of Columns.

Sets how many rows and columns of cells are to be shown in the Tag List's area. Best fit will take the size and number of cells into account to display multiple rows or columns of cells if there is room.

Row Spacing and Column Spacing.

Adds a margin, measured in pixels, between each row or column, should there be more than one row or column in the displayed tag list.

Adjust the Appearance of the Selected Widget

Select a widget from the list of those to be used in the Tag List, then click the Properties button.
A simplified version of the properties dialog for that widget will open. The contents will vary by widget. (Example showing a Numeric Value widget.)
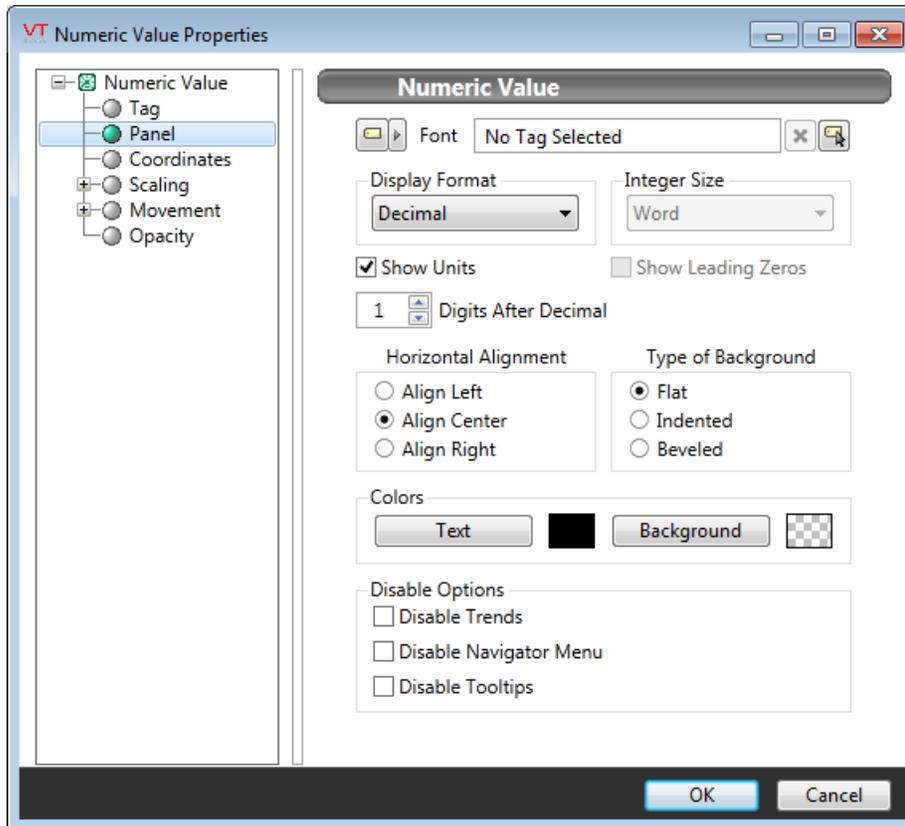
For specific instructions, refer to the chapter in this manual for the wid-
get you are configuring.

Text Change Widget

Used by: Drivers, Alarm, Alarm Status, Analog Input, Analog Output, Ana-
log Status, Analog Control, Calculation, Counter, Deadband, Digital
Input, Digital Output, Digital Status, Digital Control, Function, History
Statistics, Modem, Network Status, Pump Status, Network Status, Rate
of Change, Roster, Selector Switch, Serial port, SQL Logger, Totalizer.
* Does not use the Stye Settings tag.
The Text Change tag widget is used to represent the associated tag's
value as a text string that changes (text or color) based on the tag's
value. While this widget is available to analogs, it is most commonly used
with digital and other tags that have clearly defined states (values from 0
up to 3).

Motor Stopped   Motor Running

> Note: "2 state" versus "4 state". Not all tags will have all the states shown in these examples. Alarms and single-bit digitals have only state 0 and state 1 while 2-bit digitals also have states 2 and 3. The alarm state applies both to tags that have built-in alarms and to tags that have alarms attached to them. The alarm state will be active as long as the corresponding alarm is unacknowledged, regardless of whether the alarm is active.

The properties dialog for the Text Change widget:

Font

Select one of the font tags (or parameter linked to a font tag)

that you wish to use to display the text. The display will use the default system font if you do not select a font tag.

The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

> Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

State x Text

Provides input fields where you can type the text to be displayed when the tag is in the corresponding state.

State x Color

Opens the Select Color Dialog, from which you can choose a color to be displayed when the tag's value is in the corresponding state.

Invalid Text

Provides the same function as State x Text for the case when the tag's value is Invalid.

Invalid Color

Provides the same function as State x Color for the case when the tag's value is Invalid.

Horizontal Alignment
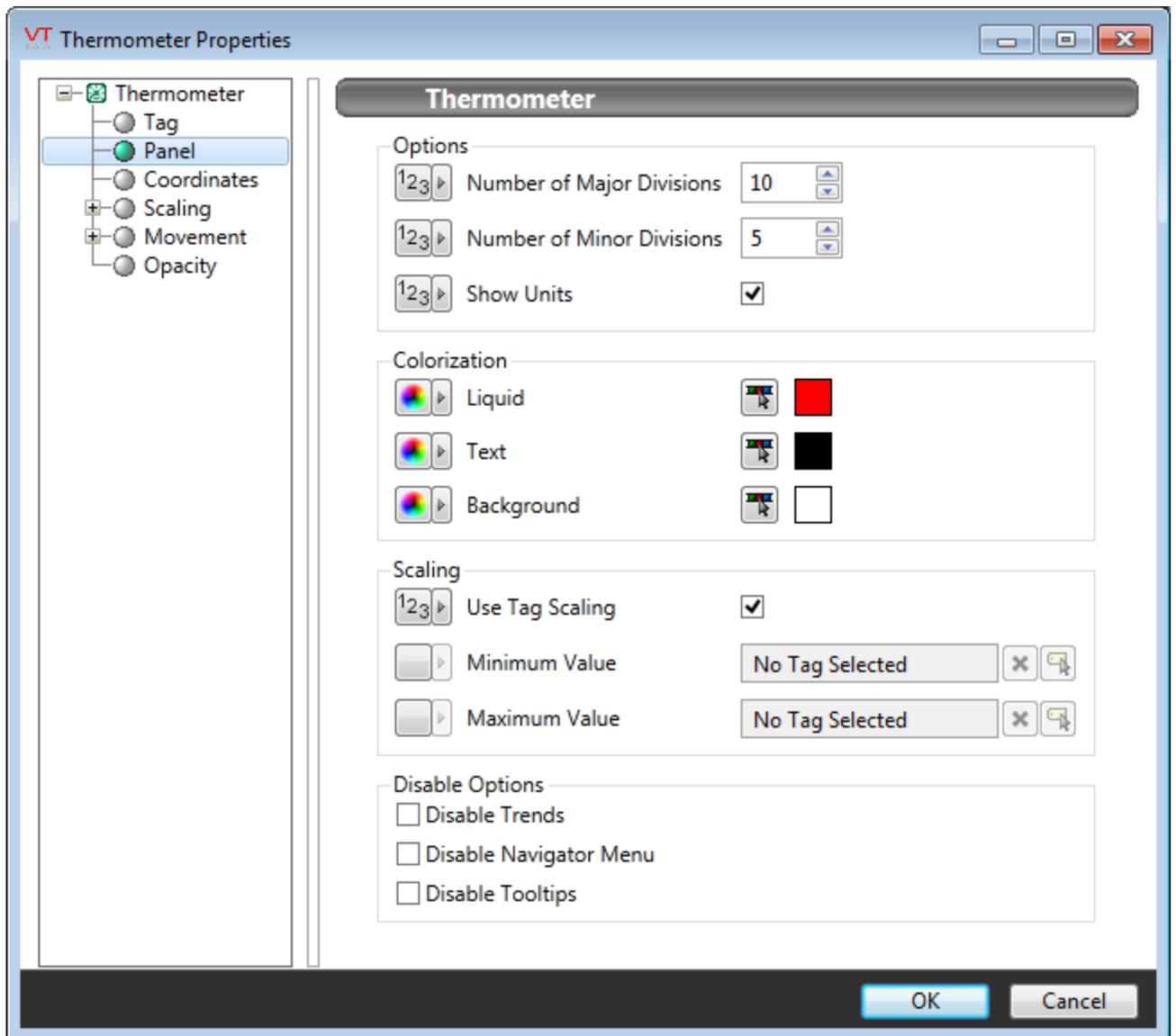
Configure the text to be left, center or right-aligned within the widget's bounding box.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Indicators\Basic Indicators\
Basic Components\Text\

Related Information:

...Choices for Operator Interaction

Thermometer Widget

Used by: Alarm Status, Analog Input, Analog Status, Calculation, Comm Link Sequencer, History Statistics, Rate of Change.
* Does not use the Stye Settings tag.

The Thermometer widget represents the associated tag's value as a photorealistic alcohol thermometer.

The properties dialog for the Thermometer widget:

Tag

The Tag field displays the name and description of the tag that is associated with this widget. You can use this to select a different tag or parameter to use as the data source for this widget.

Options

Set the number of major divisions and minor divisions within each major division. Defaults to 10 and

5.

Choose whether to include the text used for engin-
eering units within the gauge. Defaults to show
units.

> Note: You can add a degree symbol to the engin-
> eering units when configuring the tag by holding
> down the Alt-key while typing 0176. Release the
> Alt-key before typing the C or F.

Colorization

Use the color selector dialogs to set the color of the
liquid, text and thermometer background. This wid-
get does not use the Style Settings tag.

Scaling

If Use Tag Scaling is checked, the minimum and max-
imum range of the meter will be controlled by the
tag's scaled minimum and maximum values.
Otherwise, you can override those values to set your
own scale range. For example, assume that you have
a tag for which the minimum and maximum scaled
process values are 20 and 80 respectively. Drawing
this tag with a minimum value of 0 and a maximum
value of 100 will result in a meter that shows a full
range of numbers from 0 to 100, but whose needle
will move only between 20 and 80.

Disable Options

Disable selected operator-interaction features of this widget.

Related Information:

Choices for Operator Interaction

Toggle Polling Button Widget

Used by: Polling Driver, DataFlow Systems RTU
* Does not use the Stye Settings tag.
Use this widget to give operators the ability to enable or disable polling without needing to open the Polling Driver's properties dialog. Operators must possess either the Modify Tags privilege or the Manual Data privilege in order to change this setting.

While polling is disabled, no communication will be sent or received through the station. To switch from a timed poll to continuous communication, use the Fast Poll option of the Polling Driver.
The properties dialog for the Polling Button object:
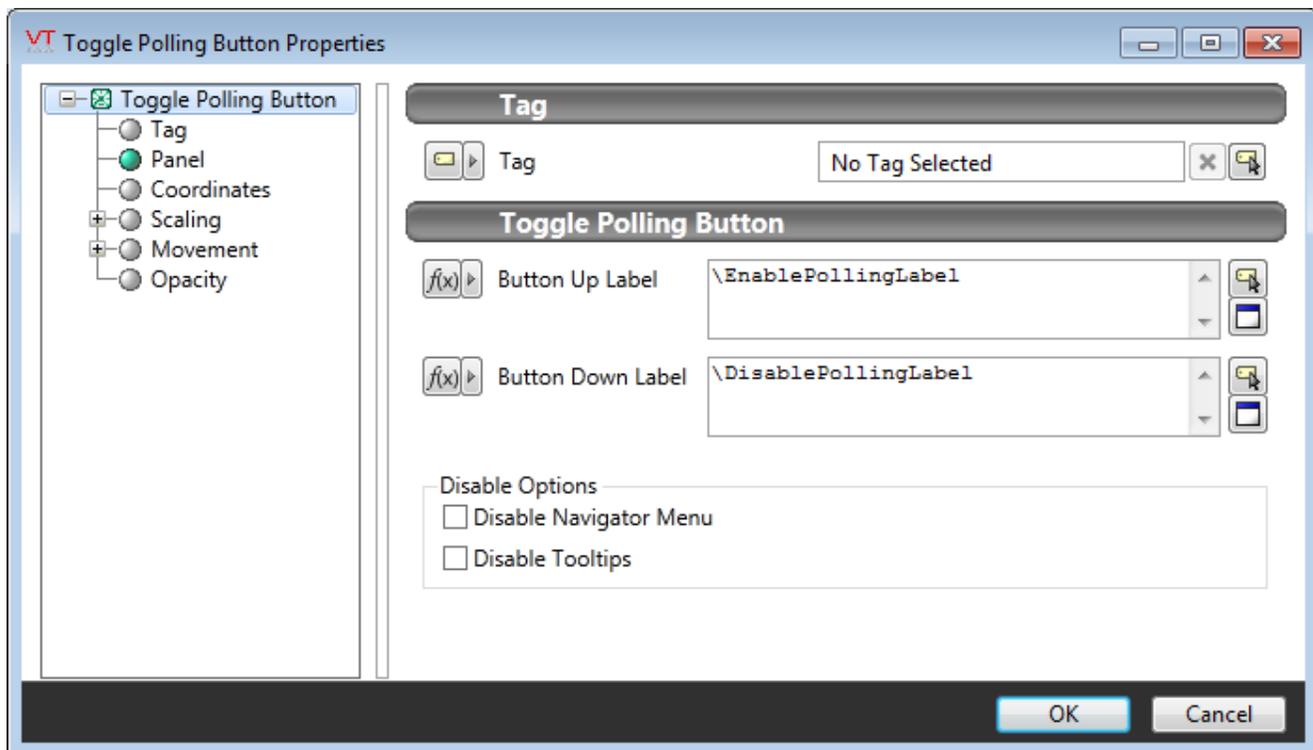
Button Up Label

The label displayed when polling is disabled, and a click on this button will enable it. The default value is a link to the

application property, "\EnablePollingLabel", which defaults to "Enable Polling". You may provide any text or expression you prefer.

Button Down Label

The label displayed when polling is enabled, and a click on this button will disable it. The default value is a link to the application property, "\DisablePollingLabel", which defaults to "Disable Polling". You may provide any text or expression you prefer.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Tag Types\Communications\Link Management\Polling Driver

Related Information:

...Enable Polling Checkbox Widget

...Choices for Operator Interaction

Toggle Switch Widget

Used by: Selector Switch.
See also: Illuminated Toggle Switch Widget
* Does not use the Stye Settings tag.
The Toggle Switch widget, one of the native widgets for Selector Switch tags, is a control method used to output either of two possible values. The values will be those configured for the first two of the three positions of the selector switch, 0 to 1.
A variety of switch images are available in the Selector Switches images group. A few of the available choices are shown here (for every On image there are matching images for the Off state and the Invalid state, drawn in the same style).

The color and image choices work the same way as an Image Change Widget. In addition to the images from the Selector Switches image group, you can use any image that has a color indicator region defined within it (pump and motor images commonly include these indicator regions). This will allow you to use an equipment image as a toggle button in your application. You could also choose to not use any image, and simply place the rectangular drawing area over a region of active orange (RGB 255, 160, 0) to create a custom control toggle.

To indicate to the operator that the image is a button, a hotbox is drawn around it. A hotbox is commonly understood to define an area that can be clicked upon as a control action.

In the event that the position feedback from the selector switch indicates that the actual position is not the same as the requested position, the border of the image will blink using the mismatch color. You should select a color that stands out clearly on your page background, but red is not advised since it is commonly understood to mean that an alarm has been triggered.

The switch works by toggling between states 0 and 1 whenever the operator clicks on the button. Should the tag be in an invalid state (which will not be unusual when a selector switch is first created) then it is not clear whether the first click should move the switch to the state 0 or the state 1 position. For this reason, a dialog will prompt the operator for an initial position, using the terms you provide in the Selection Dialog area. Selecting the Confirmation Dialog option will force the operator to confirm each click on the button by selecting OK or Cancel from a dialog.
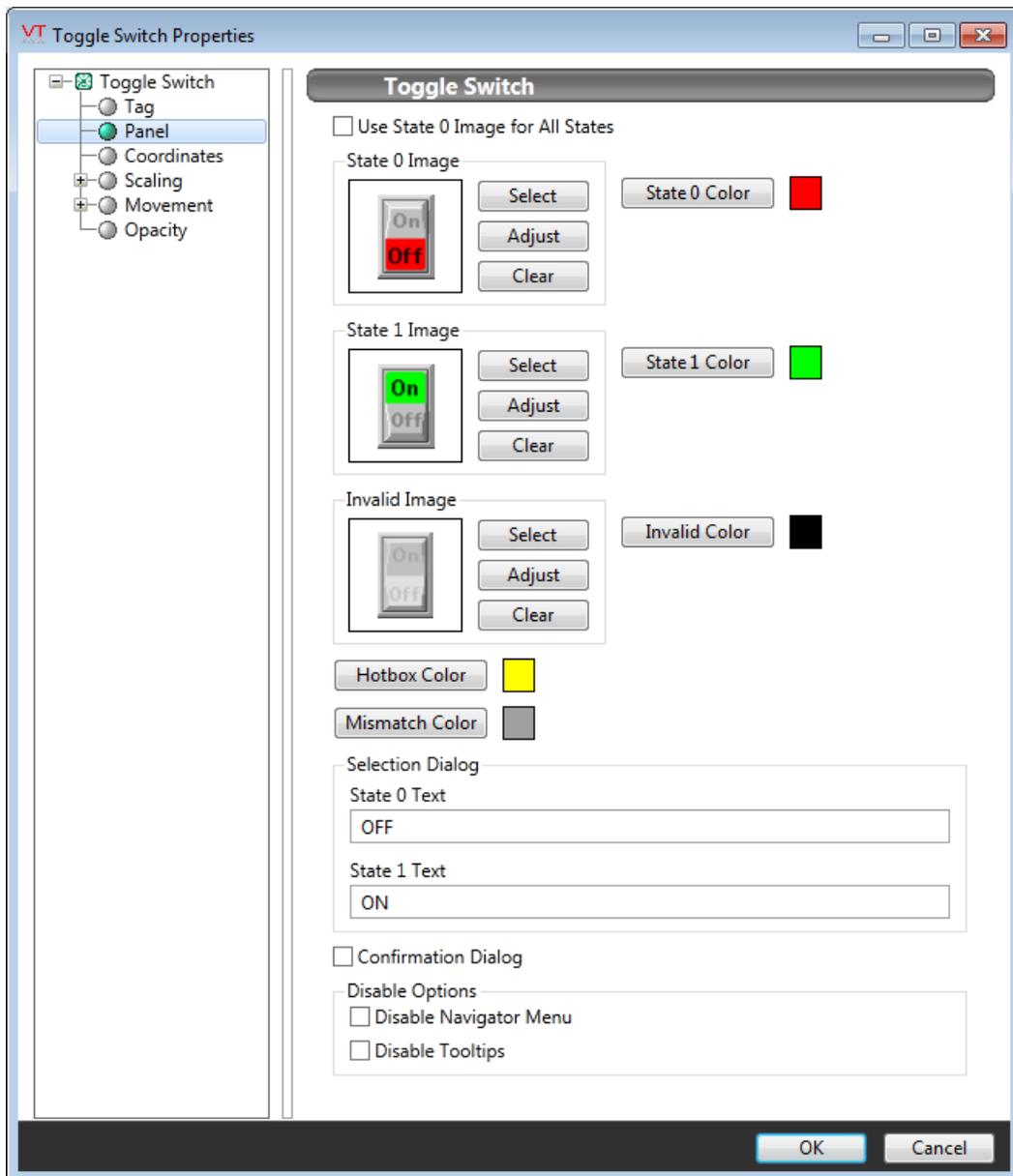
Use State 0 Image For All States

Check this box if you wish to use the image selected for State 0 for all states of the tag.

This option is often used when the selected image's indicator area (the small rectangle at its base) is to change color according to the state of the tag.

State x Color

Opens the Select Color Dialog, from which you can choose a color to be displayed when the tag's value is in the corresponding state.

State x Image

Image Select

Opens the Select Image dialog, within which you can browse for images, import new images, and select the image to use.

Image Adjust

Opens the Adjust Image dialog box, which can be used to change the color and other display characteristics of the image. See: Adjust Image Dialog

Image Clear

Remove the selected image and do not select a replacement.

Invalid Color

Provides the same function as State x Color for the case when the tag's value is Invalid.

Invalid Image

Provides the same function as State x Bitmap for the case when the tag's value is Invalid.

Hotbox Color

When the operator moves the mouse pointer over the Toggle Switch image, a hotbox will appear to help indicate that this is a control that can be clicked. Use this option to select the

hotbox color.

Mismatch Color

If the associated Selection Switch tag detects that the actual position of the switch is not the same as the position that the operator requested, a box of the mismatch color will blink around the switch image to indicate a problem. The color selected for this should provide a good contrast from the page background, and should also be a color that the operator will associate with a problem condition.

Selection Dialog (State 0 Text and State 1 Text)

If the selector switch starts from an invalid or unknown position, it must determine which state the operator intends, the first time that it is clicked. The Selection Dialog will be displayed with the text you enter for State 0 and State 1 to prompt the operator upon first click.

Confirmation Dialog

Select this option if you want the operator to have to confirm each control action from this object. See also: Confirmation Prompts for Output Tags.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Buttons & Switches\

Related Information:
...Choices for Operator Interaction

Top Bar Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Deadband Control, Function, History Statistics, Script, Selector Switch, SQL Logger.
* Does not use the Stye Settings tag.
The Top Bar widget is used to display a tag's value as a vertical bar, the length of which increases from the bottom towards the top as the tag's value increases. When the tag's value is at a minimum, the bar will be shown as a thin line at the bottom of its range. When the tag's value reaches its maximum, the widget will be displayed as a rectangle, filling the range.

at minimum   at maximum   being edited
                            at 54%
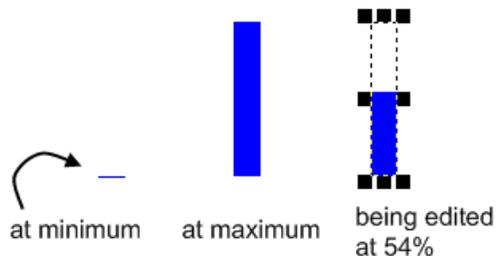
The properties dialog for the Top Bar widget:

Set Color

Opens the Select Color Dialog from which you can select a color for this object.

Low Scale Range and High Scale Range

In most cases, these fields will be display-only, showing the Scaled Process Data Min and Scaled Process Data Max values configured for the associated tag.
These fields become editable when the widget is used to display a function or Calculation tag. The most common use is to provide an output range of 0 to 100 (%) given a data source whose numbers do not match these values (perhaps ranging from 20 to 80, for example). Set the low scale range to the value expected from the data source when it is at its minimum and the high scale range to the value expected from the data source when it is at its maximum.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Gauges\Basic Analogs\
Basic Components\Bars\

Related Information:

...Choices for Operator Interaction

Two Color Bar Widget

Used by: Alarm Status, Analog Input, Analog Output, Analog Status, Analog Control, Calculation, Comm Link Sequencer, Counter, Deadband, Digital Control, Digital Status, Function, History Statistics, Network Status, Pump Status, Rate of Change, Script, Selector, Totalizer, Trigger, SQL Logger.
* Does not use the Stye Settings tag.

The Two Color Bar tag widget is used to display tag data using an animated, multi-colored bar. Its appearance is similar to the Color Fill widget with the addition of a second color that will be displayed when the value goes above a set point. A common use of this second color is to provide clear notice that a value is above its safe limit.
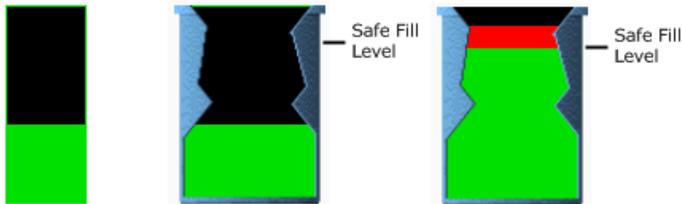
Examples showing the Two Color object, the same object behind a tank cut-away image and again with a value exceeding the set point.

Despite the name, four colors are used to fully configure the Two Color Bar object:

- The value of the tag is displayed in one color, the Bar Color, shown as green in the example.
- The portion of the tag's value exceeding a user-defined setpoint value is displayed in a second color. This is the High Color, shown as red in the example.
- The background of the object's drawing area is displayed in a third color: the Unfilled Color.
- If you would like this object to be visible only where it overlaps a defined area, you may set the Changed Color to that of the defined area. Any portion of the Two Color object not over that color will be invisible. This feature can be used to build your own tank cut-aways.

The properties dialog for the Two Color widget will look similar to the following. There must be a Filled Color, but you are free to leave the Changed Color and Unfilled color properties unselected.
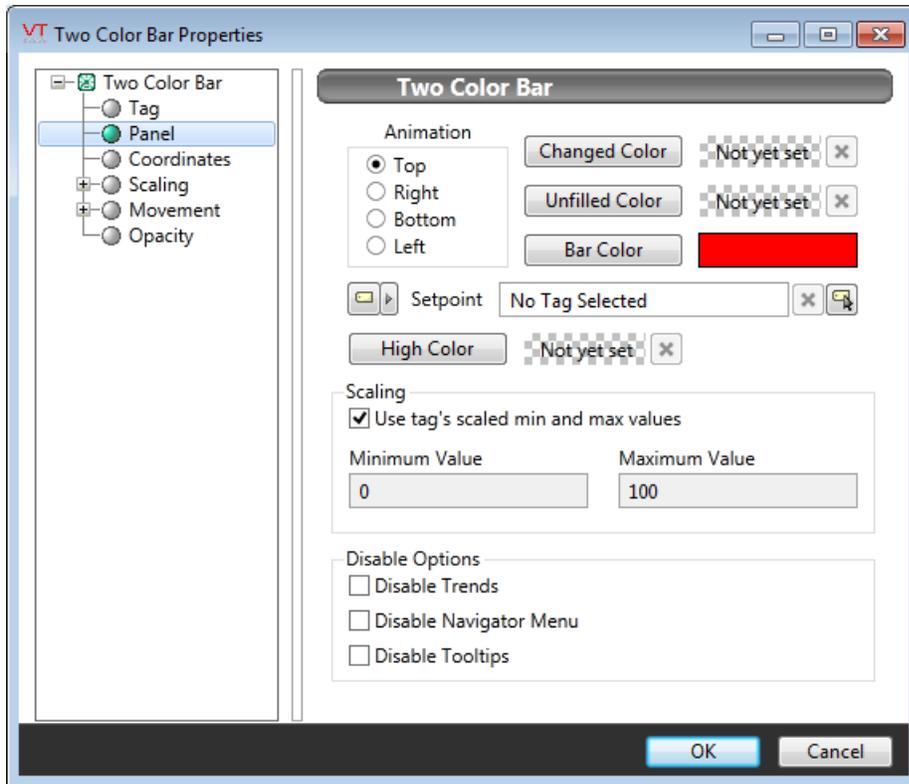
Animation

Select one of Top, Right, Bottom or Left to set the direction that the bar will grow with increasing values.

Changed Color

If set, the object will only be visible when placed above an area of this color. The button opens the Select Color Dialog from which you can choose a color. Select transparent in order to stop using the changed color. Click the X button to remove the color if you change your mind.

Bar Color

Opens the Select Color Dialog from which you can choose a color to that will represent the tag's value.

Unfilled Color

Opens the Select Color Dialog from which you can choose a color to be displayed as the background for the bar color.

Setpoint

Use this field to define the point, above which the tag's value will be shown using the High Color rather than the Bar Color. May be set using any of a constant value, an expression, a second tag's value or a page parameter.

High Color

Opens the Select Color Dialog from which you can choose a color to be displayed for the portion of the bar showing the value above the set point.

Use Tag's Scaled Min & Max Values

In most cases, it is best to select this option. When the tag is at its minimum scaled output value, the bar will not be visible and when the tag is at its maximum scaled output value, the bar will fill the area of the widget.

Choosing not to use the tag's scaled min and max values enables the Minimum Value and Maximum Value controls which you can use to scale the bar length relative to the tag's output scaled values. The scaling you set will apply only to this one instance of the tag widget. This option is normally used when the object represents tags such as Calculation and Function that do not have their own scaled values.

Unlike widgets such as the Top Bar, the bar for the color fill will be visible only within the area originally drawn for this object, regardless of the scaling you define.

Minimum Value and Maximum Value

The Minimum Value and Maximum Value fields are used to scale the color fill bar relative to the tag's scaled output values. For example, given a tag with scaled output ranging from 0 to 100:

Setting the widget's scale range from 0 to 200 results in a bar that reaches only half the length drawn when the tag is at its maximum value.

Setting the widget's scale range from 0 to 50 results in a bar that reaches its full length when the tag is still only half way to its maximum value.

Unlike widgets such as the Top Bar, the bar for the color fill will be visible only within the area originally drawn for this object, regardless of the scaling you define.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Basic Components\Bars\
Basic Components\Color Indicators\

Related Information:

...Choices for Operator Interaction

...Color Fill Widget

Totalizer Widget

Used by: Pulse Input tags

Note: This widget is *not* used by the Totalizer tag.

* Does not use the Stye Settings tag.
The Totalizer widget calculates and displays the combined total of the value of the Pulse Input tag being drawn within a user-defined period of

time. Each time the user-defined period of hours has cycled, the Total-izer object will automatically reset to 0 and will begin to recalculate.

The Totalizer tag widget does not calculate the average value of the tag; rather, it increments the new value to the former value each time the value changes. For example, if the value of the Pulse Input tag changes from 27 to 30 to 32 to 21 within the user-defined period of time, the value displayed by the Totalizer object will be 110. It should also be noted that the Totalizer tag widget rounds values off before totaling them (for example, 23.62 will be rounded off to 24 and then added to the total value).

The properties dialog for the Totalizer widget:

Font

Select one of the font tags (or parameter linked to a font tag)

that you with to use to display the text. The display will use the default system font if you do not select a font tag.

The parameters button is enabled only if the tag is drawn on a page that includes a font parameter, or if the tag is part of a widget and the associated widget editor is open.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Type of Border

Examples of the three border types are shown at the beginning of this topic.

Colors (Text and Background)

Opens the Select Color Dialog from which you can select the color of the text.
The background color against which the text will be displayed can also be defined.

Restart Period

Set to the number of hours (or fractions of an hour) after which the widget should reset to 0 and begin totaling again.

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:
Analytics\Statistics\

Related Information:

...Choices for Operator Interaction

Vertical Button Widget

Used by: Selector Switch.
* Does not use the Stye Settings tag.
The Vertical Button widget, one of the native widgets for Selector Switch tags, is a control method used to output one of three possible values. The values will be those configured for the three positions of the selector switch, 0 to 2 in order from top to bottom.

Note: If the underlying selector has only two positions, then only the top and the middle button will be functional. The operator may click the bottom button, but this will have no effect on the switch position or output. Due to the confusion that this may cause, vertical buttons should be used only with three-position selector switches.

Color is used to indicate that either a mismatch in state has occurred, or that an alarm is active as shown in the following examples.

Vertical buttons in a normal operating state (A), showing a state mismatch (B) and showing an active alarm (C). (See: Selector Switch Tags, for a discussion of mismatch conditions.)
The following image shows the properties dialog where you can set the properties of the object.
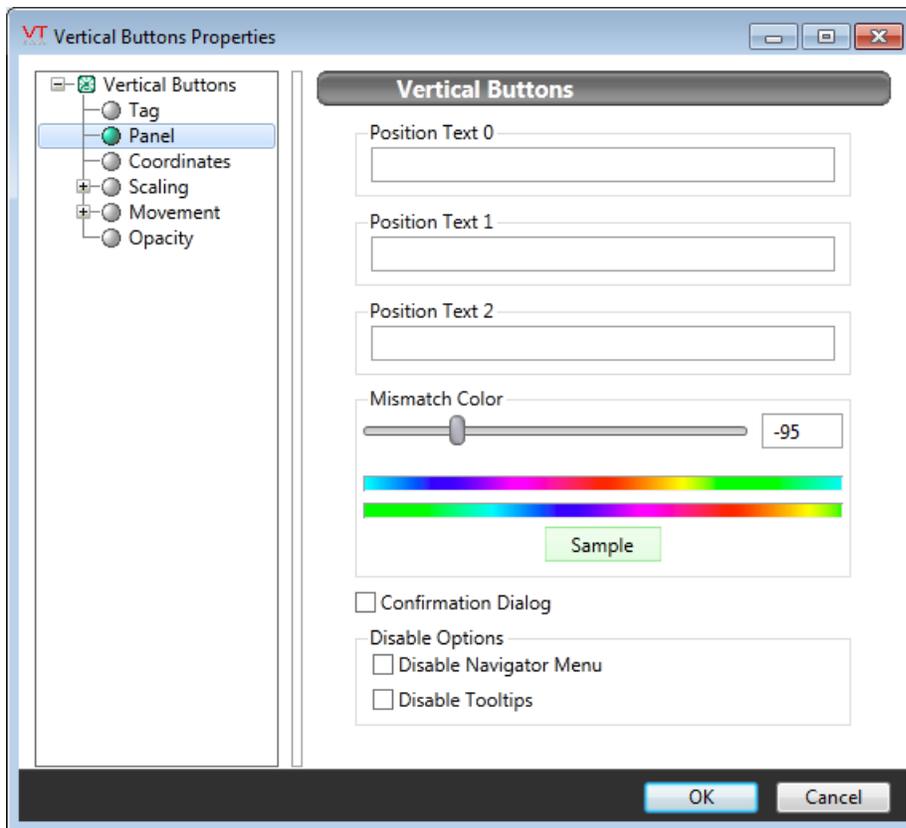
Position Text n

Set the labels that will appear on the buttons. The buttons are drawn from top to bottom and match the Position 0 through Position 2 settings of the associated selector switch.

Mismatch Color

Use the slider to set the color that will be superimposed on a button in the event of a position mismatch in the selector switch. A sample image displays the result as you adjust the slider.

Confirmation Dialog

If selected, the operator will be prompted to confirm every change made by clicking on the buttons. See also:

Confirmation Prompts for Output Tags.

An example of the confirmation dialog:

Disable Options

Disable selected operator-interaction features of this widget.

Widget Folders:

Buttons & Switches\Basic Controls\

Basic Components\Button Controls\

Related Information:

...Choices for Operator Interaction

VTScada Graphics Library

Images provide visual context for a page. They may be maps, pho-tographs or drawings of equipment, flow charts or anything else. A selec-tion of photo-realistic background images of water, metal, stone, and more are available, as are lighting overlays. Note that these can be scaled or tiled to fill an area.

Note: Images have no "Link" option to tie them to a tag. They are essen-tially static, although any property (movement, scaling, etc.) can be animated by linking it to a tag value or to an expression.
The Style Settings tag has no effect on images and should not be linked to any property.

Add an image to a page: (Method 1)

1. Open the Idea Studio and open the page where the image should be added.
2. Open the Images palette.

3. Click on a group to view the images within it.
   If required, click the home button at the top of the folder list to return to the top level and select a different folder.

4. Drag an image from the palette to your page.

5. Adjust the size, location and other properties of the image as required.
   Most tools are available through the image format ribbon.
   Advanced configuration details are available by opening the Properties dialog of the image.

Add an image to a page: (Method 2)

1. Open the page in the Idea Studio.

2. Open Windows Explorer and find an image on your computer.

3. Drag the image from Windows Explorer to the page in the Idea Studio.
   The Imported Images folder of the Images palette will be revealed, and the image will be imported to your project and added to that folder.

4. Click OK in the New Menu Item dialog to accept the defaults. (Or, edit as needed before clicking OK.)

5. Adjust the size, location and other properties of the image as required.

Troubleshooting:

- The image is too big / too small.
  All images have an initial size, which may be wrong for your application. After the image has been placed, zoom in or out on the canvas until you can see the edges easily, then select the image and drag the corner grips to resize it.

Related Information:

...Draw Widgets - Display tag values or provide operational controls.

...Draw Shapes - Provide context to widgets using geometric building blocks.

...Ribbons - A description of the tools in each formatting ribbon.

Next Steps:

...Select and Group - Before changing the appearance of an object, you must first select it.

...Align – Tools and techniques for well-organized pages.

...Import Images – Add your own images.

...Image Formatting – Change how images look.

Images Palette / Select Image

Images are static graphics in the sense there is no "Link" option to tie them to a tag. They are used to represent background views, equipment, instruments, etc., providing context for the widgets that are linked to tags.
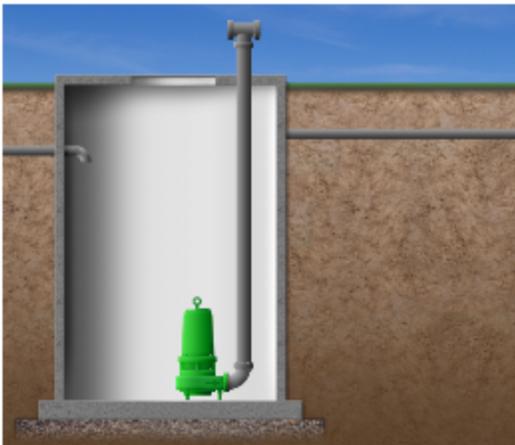
In this example, the pump is a widget that is linked to a tag and changes color to match the tag's state.
The wet well is an image, providing visual context for the pump widget, but not linked to any tag.
The contents of the default image palette match the Select Bitmap dialog. Either may extended by adding new palette groups or image folders. Additions are specific to your application, not to VTScada in general.
The palette is used when creating new images. Click a group to view the images within it, then drag images onto the page canvas. Click once on an image to return to the parent folder.
The Select Image dialog is used when changing an existing image.
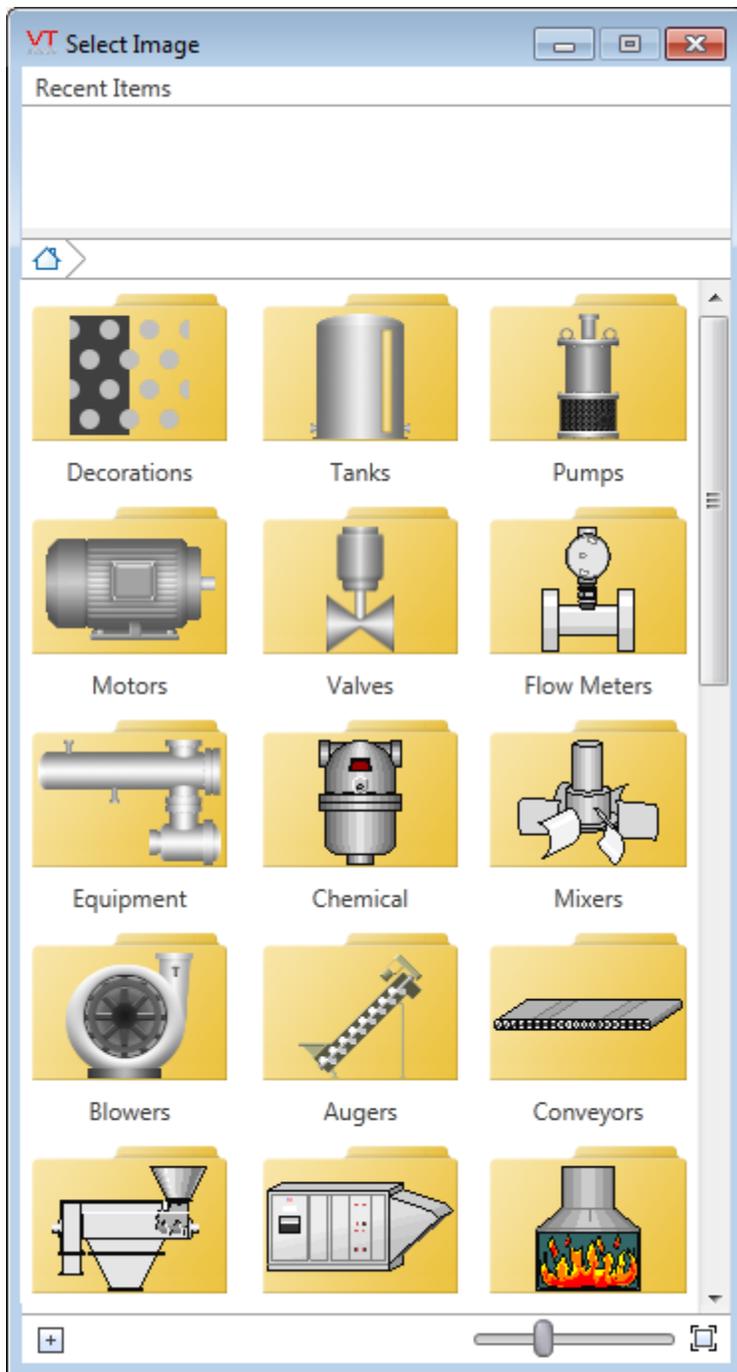
Folders can be opened with a click. A click on an image will select it, completing the Select Image operation.

New folders can be created, and new images imported using the buttons at the bottom of the Select Image dialog.

Related Tasks:

...Select and Group – Before changing the appearance of an object, you must first select it.

...Align – Tools and techniques for well-organized pages.

...Import Images – Add your own images.

...Image Formatting – Change how images look.

Import Images

Even though several thousand images are included with every copy of VTScada, it is likely that you will want to use your own images from time to time.

> Note: Images are imported into the open folder of the palette. You should keep your palette organized by opening the folder where the image is to be imported, or by creating new folders for your images.
> • You can create new folders for your imported images using the same process (described following) as for importing images. New folders are automatically opened.
> • There is a limit to the size of the image that can be imported. No image may measure more than 10,000 pixels on either side.

Folders can be created using the same dialog box that opens when importing an image, as described in the following steps.
Palettes and the items within them such as images and folders, are organized and stored in Menu Tags. The following steps will therefore refer to menu items. The same result can be obtained by starting with the Tag Browser.

To import an image: (Method 1)
1. Open the Images palette of the Idea Studio.
2. Open a folder for the image to be imported.

3. Click the Import Image button.
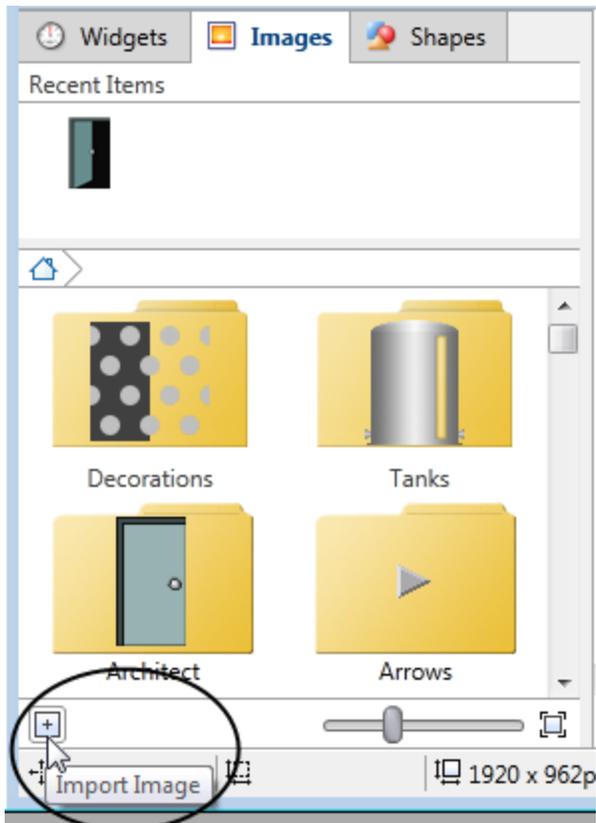
The New Menu Item Properties dialog will open.
It is at this point that you could choose to create a new folder before repeat-
ing the steps to import images to that folder.

4. Click, Image in the New Menu Item Properties dialog.
The Import Image dialog will open.

5. Browse for and select the image to be imported.
The New Menu Item Properties dialog will display a preview of the image, and
its properties.

6. Click OK.
The image will be added to the open folder.

To import images or a folder of images: (Method 2)

1. Open the Images palette of the Idea Studio.

2. Within the palette, open a folder for the image or images to be imported.

3. Open Windows Explorer and find the image or images that you want to
import.
Note that you may select multiple images, or a folder of images.

If you select a folder, VTScada will attempt to create a matching folder within the palette before importing the images.

4. Drag the image from Explorer to the palette.
   The New Menu Item Properties dialog will open. In most cases, there is no need to change any parameters within this dialog.

5. Click OK.
   The image will be imported, and added to the current folder at the bottom of the list.

> Note: Method 2 works equally well if you drag the image to the editing canvas rather than to a palette folder. The difference is that the image will be saved to the folder, "Imported Images" instead of a folder of your choice.

Remember that you can edit the Images menu within the Tag Browser, to organize the palette as you see fit.

Troubleshooting:

- The image is not visible after being imported.
  It may be necessary to scroll through the images of the current folder to find the one you imported.

- The image is in the wrong folder.
  Use the Tag Browser to reorganize the palette.

Related Information:

...Adding New Menu Items

Decorations

As a convenience and due to history, the decorations folder can be found in both the Widgets palette and the Images palette. Decorations are images that you add to a page to enhance other objects. Examples include lighting overlays, tiles & textures, folders, frames and tooltips. The following example illustrates how decorations such as tiles and light-ing overlays can enhance your application
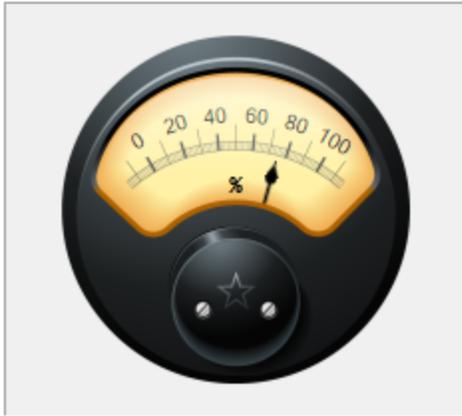
Antique gauge on a plain border
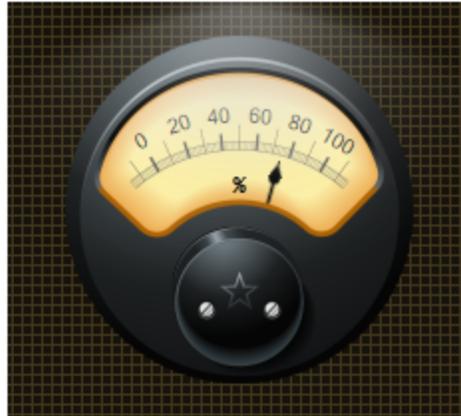
The same on a tiled grid,
with a spotlight fade decoration

Related Information:

...Border

...Grid

...Folders

...Frame

Border

Borders are most often used to provide a visual indication to operators that a set of controls have some common purpose. Borders can appear raised or sunken relative to the rest of the page, and can be transparent or shaded.

You can find the Border image in the decorations folder of the Images palette.

The properties dialog for a border:
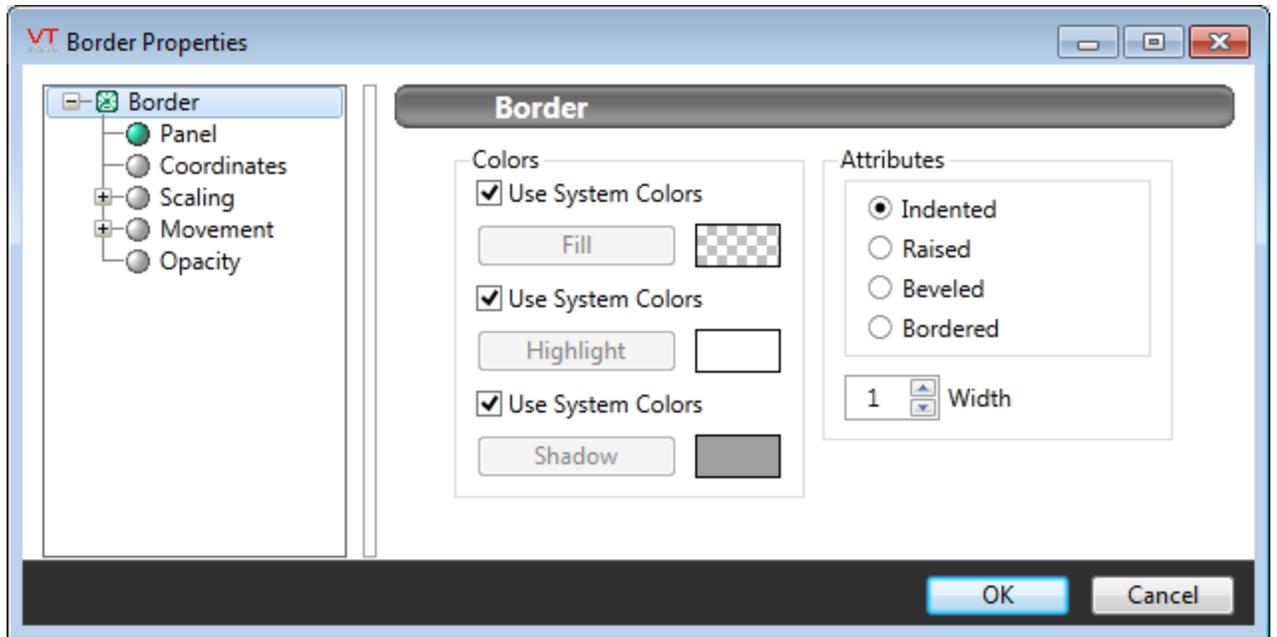
Colors

By using the default option of System Colors, you can make your borders look similar dialog boxes used by your operating system.
The fill is the background color of the border's area. The high-light and shadow are used to make the border appear raised or sunken by simulating light shining on one edge or the other.
If you choose your own fill color, it is recommended that you use a darker shade of the same color for the shadow and a lighter shade of that color for the highlight.

Attributes and Width

Selection option for the visual effect that will be applied to the border. The difference is in the choice of the order of the shadow and highlight lines around the fill area.

Related Information:

...Frame

...Rounded Rectangle

Grid

The grid shape creates a rectangular area of any color, visually divided into smaller rectangles by lines.
The background and the lines can be given any color you prefer, or either can be made transparent. Lines may be solid or dashed.
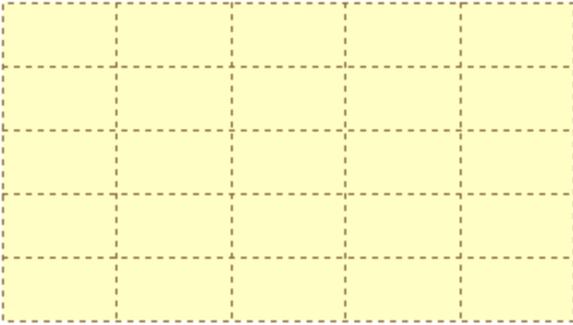
You can find the Grid image in the decorations folder of the Images palette.
The properties dialog for the grid:
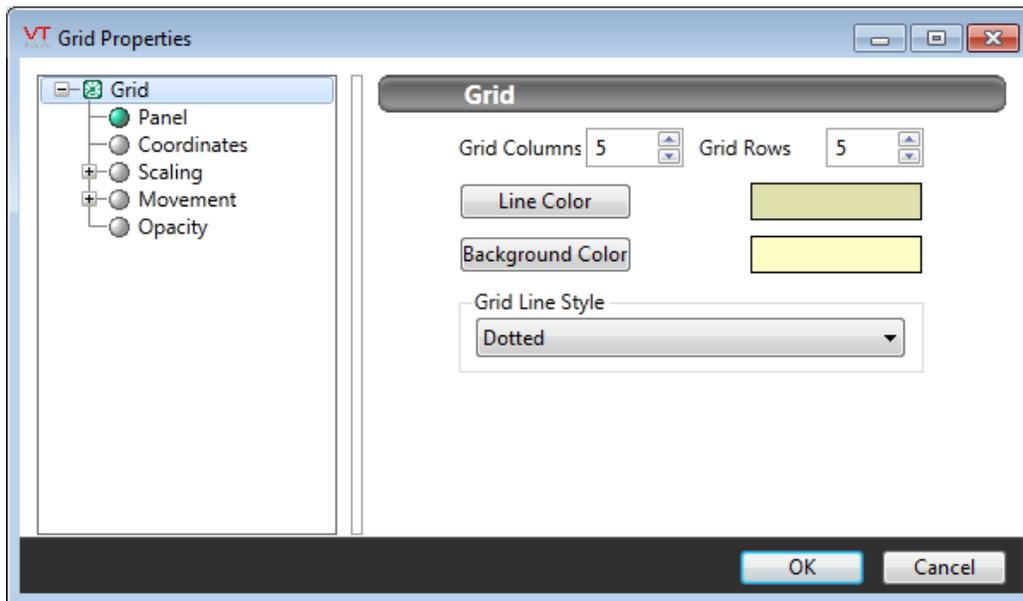
Columns and rows

Control over how many cells will be included in the grid. Cells are always uniform in size.

Colors and Line Style

Click to open the Select Color dialog for each part of the grid.

Six line styles are available, including invisible.

Folders

The Folders widget creates a user-interface object with multiple tabs; each tab has a separate display of system monitoring and control objects. The tabs are used to select which frame is to be displayed. Three styles of folder are provided in the palette, but these are all variations on the same widget. (See Tab Style, later in this topic.)
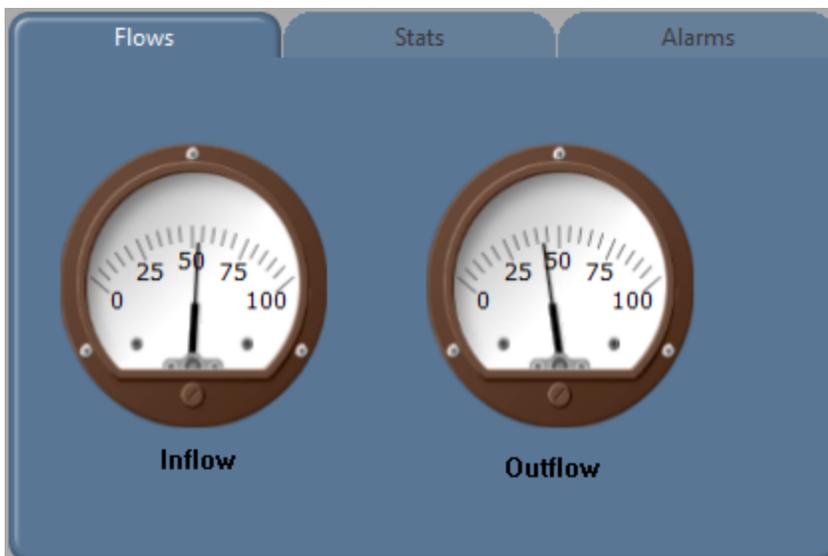
A folder may have from one to fifteen tabs. You have extensive control over the appearance of the folder and the tabs within it.

The content for each tab's frame must come from a pre-existing object in your application. These may be pages, tag drawing methods, or library drawing methods (normally, User Draw methods). You do not draw objects on a folder as you would on a page.

There are two parts to a folder's configuration: The first defines the overall style of the folder. The second controls the content and the appearance of the individual tabs.

Define a Folder's Overall Style

The Style tab of the folder configuration:
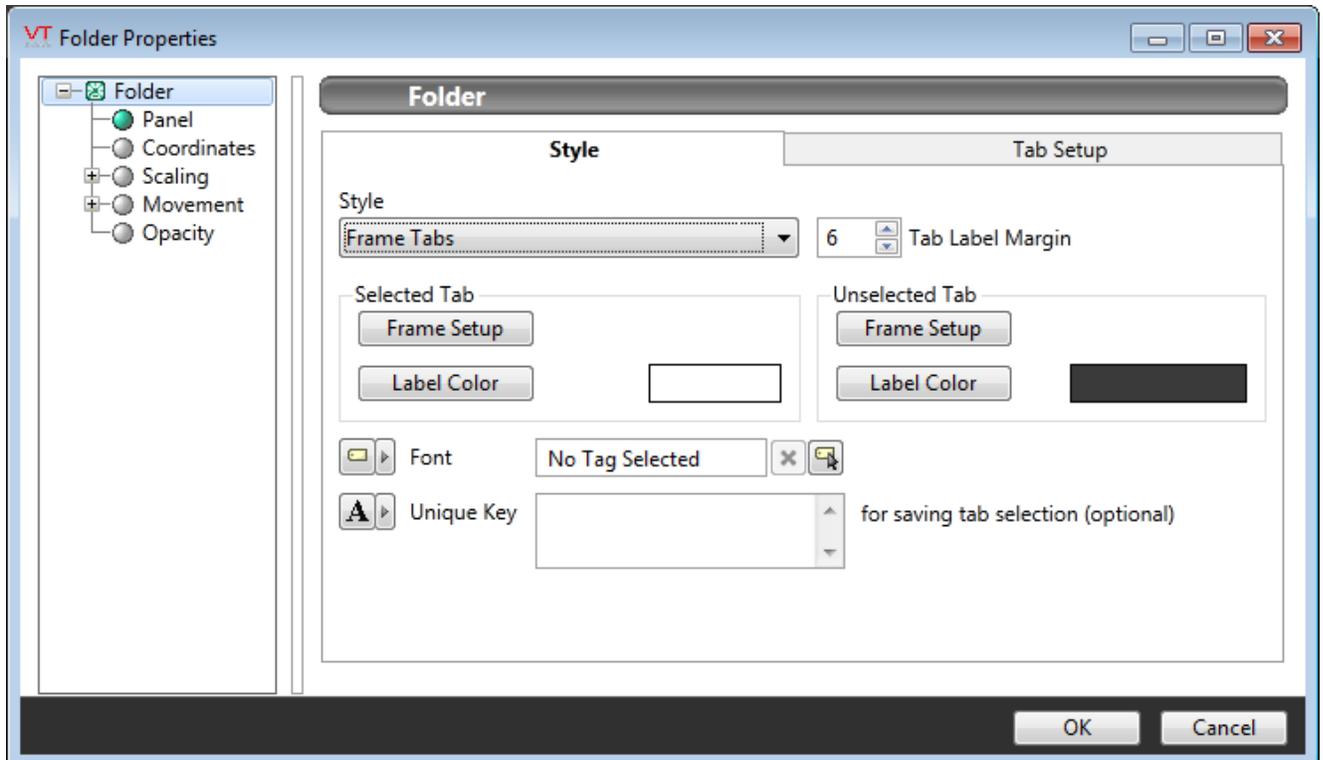
Tab Style

The tabs may use any of the following three styles (shown here using default colors).
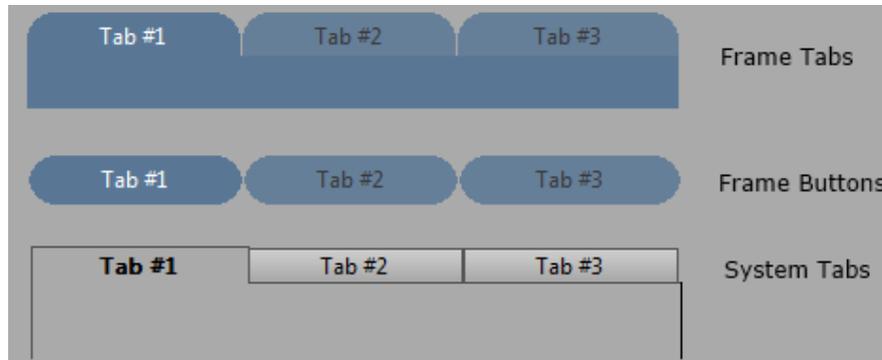
Tab Label Margin

Sets the number of pixels above the tab labels. A large margin value may increase the height of the tabs, but will not increase the width – if necessary, the label will be truncated to provide the required margin within a given tab width. The overall folder width, divided by the number of configured tabs, controls the width available for each tab.

Selected Tab's Frame Setup

The shape of the tabs themselves and the area below the selected tab are controlled by the Frame Setup selection. If either Frame Tabs or Frame Buttons are selected for the overall style, then you may further refine the shape by selecting one of the six built-in frame designs. See Frame for details. The System Tabs style does not allow for further refinement of the frame.

Selected Tab's Label Color

Opens the Select Color dialog, from which you can choose the color, in which the selected tab's text will be displayed. To emphasize that a tab is selected, it is best to choose a color with high contrast from the tab's color.

Unselected Tab's Frame Setup

Provides the same set of options as for the Selected Tab's Frame Setup. Use care with this selection: by choosing a raised frame for the selected tab and a flat or sunken frame for the unselected tabs, you can provide a clear visual message to the operator to help them see which tab is selected. Choosing a frame for the unselected tab that is a completely different style from that of the selected tab may cause confusion.

Unselected Tab's Label Color

Sets the color of the text to use for the labels on the unselected tabs. This should have a low level of contrast against the tab's background color in order to indicate that the tabs are not selected, yet still have enough contrast to be easily read.

Font Style

Select a Font tag to define the appearance of the tab labels.

Note: Note: While it is possible (and very easy) to modify the appearance of the text within this widget using the Format ribbon of the Idea Studio, you will find it much easier to manage a group of similar controls by defining a font tag and selecting that for each. Changes to all can then be made by adjusting the properties of a single font tag.

Unique key for saving selected tab

Variables such as dialog positions and sizes are normally retained per-user and per-module. What this means for the folder is that the "Last viewed tab" would be saved, but shared across all folders.

By providing a unique key to this parameter, each folder drawn in an application can save its own last viewed tab. If used, the

key may be any phrase such as, "Main Station Display" or "Treat-ment Plant Panel".

If the folder is used in a parameterized page, it will be neces-sary to have a different key for each version of the page that you open. Therefore, this property may be linked to a page parameter or expression.

Define a Folder's Tabs

The number of tabs, the label for each, and the content of each tab's frame are defined with the Tab Setup.
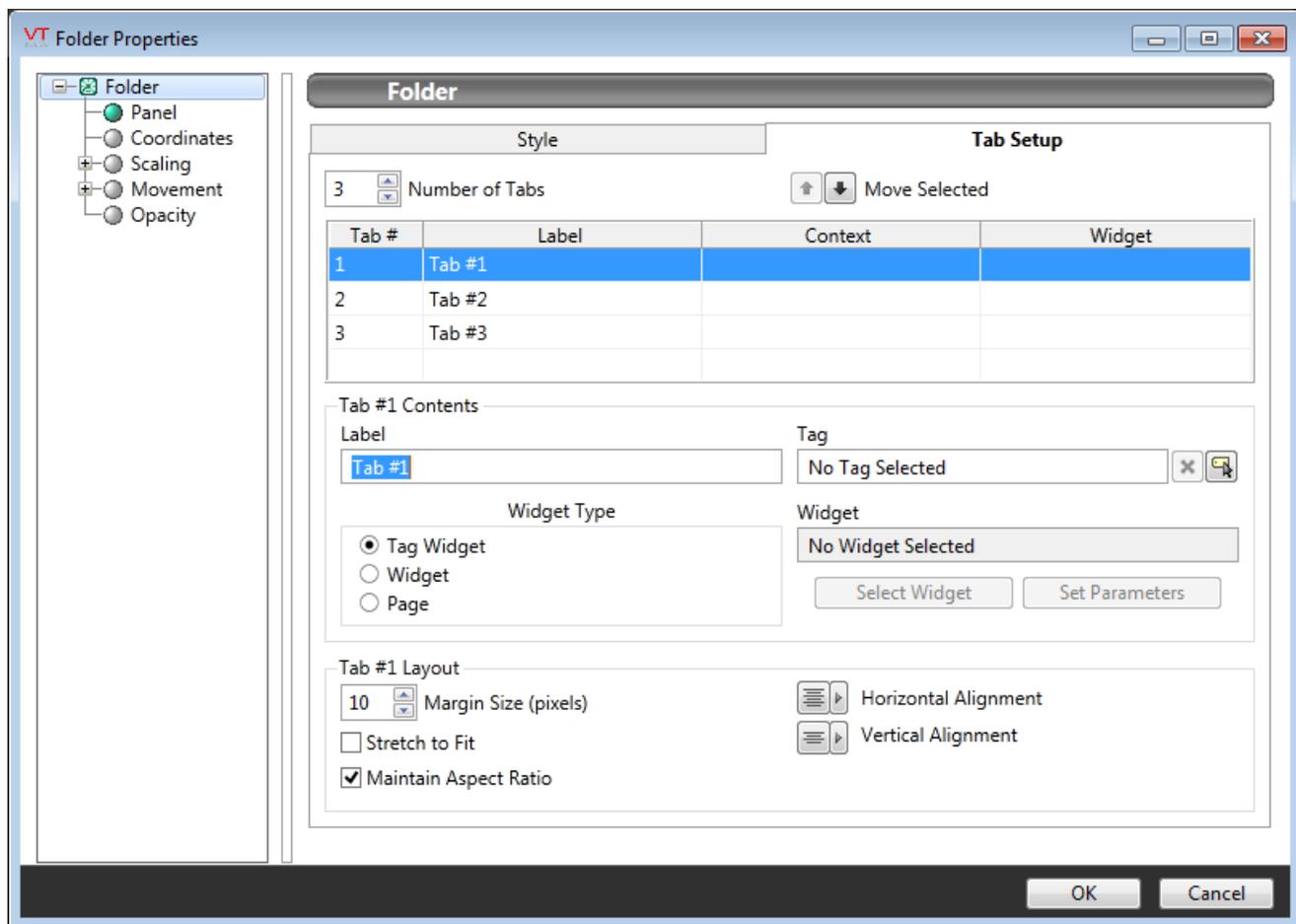
The table in the center of this panel displays a summary of each tab's con-figuration and can be used to re-order their appearance in the folder.

Use the controls above or below the table to modify the tabs.

Number of Tabs

May be any value from one to fifteen. This only changes the number of tabs displayed; tabs are not added to or deleted from the folder. Thus, if you configure four tabs, then drop the number to three, you will have hidden the fourth tab, not deleted it.

Move Selected

Tabs may be re-ordered. They are displayed left-to-right across the folder in the order that they appear from top to bottom in the table.

Tab Layout

Provides control over how the contents of the tab fit into the folder size.

- The margin, measured in pixels, will ensure a set amount of blank space around the contents. If the area of the folder plus the margin is smaller than the selected content, then the content will be scaled down to fit.
- If Stretch to Fit is selected, the folder content will be expanded to fit the available size.
- The Maintain Aspect Ratio option controls whether the content may be deformed when being stretched or shrunk.
- Finally, the Horizontal Alignment and Vertical Alignment options control how content that is smaller than the folder will be placed within the available space.

Tab Contents

For the selected tab, you may provide the text for the label and the contents of the tab's frame.

Any one of three types of object may be used for the tab's contents: Tag Widget, Library Widget or Page. The specification of the contents depends on the content type. See following examples.

Steps to use a Tag Widget:

1. Select Tag Widget as the Widget Type.
   The Context field will be enabled.
2. Select the tag type in order to filter for its available widgets.
   For example, if you select an Analog Status tag, you can then choose from any of the Analog Status widgets.
3. Click the Select Widget button to choose from the available representations of this tag.
   If this widget uses parameters, you will be prompted to set those parameters for this instance.
   You may later use the Set Parameters button to change those selections.

Steps to use a Library Widget:

1. Select "Widget" as the Widget Type.
   The Context field will be enabled.
2. Click the Select Widget button.
   A palette will open. You may use any folder.
3. Select the Widget that you want to show on the tab.
   If this widget uses parameters, you will be prompted to provide set those parameters for this instance.
   You may later use the Set Parameters button to change those selections.

Steps to use a Page:

1. Select Page as the Widget Type.
   The Context field will be set to "Pages" and disabled.
2. Click the Select Page button to open a menu of all the pages in this application.

Any page may be used, but in general, it is best to use a page that was designed to fit the area of this folder.

3. Select the Page that you want to show on the tab.
   If this page uses parameters, you will be prompted to set those parameters for this instance of the page.
   You may later use the Set Parameters button to change those selections.
   Note that, the page's background image will not carry through to display in the tabbed folder.

Related Information:

...Frame

...Border

Frame

A frame is similar to a border, except that it provides a much greater range of formatting options. Frames are used in the construction of Folders.

Three frames.
You can find the Frame image in the decorations folder of the Images palette, and in the Standard Library sub-folder of the Tools folder in the Widgets palette.
The properties dialog for the frame shape:
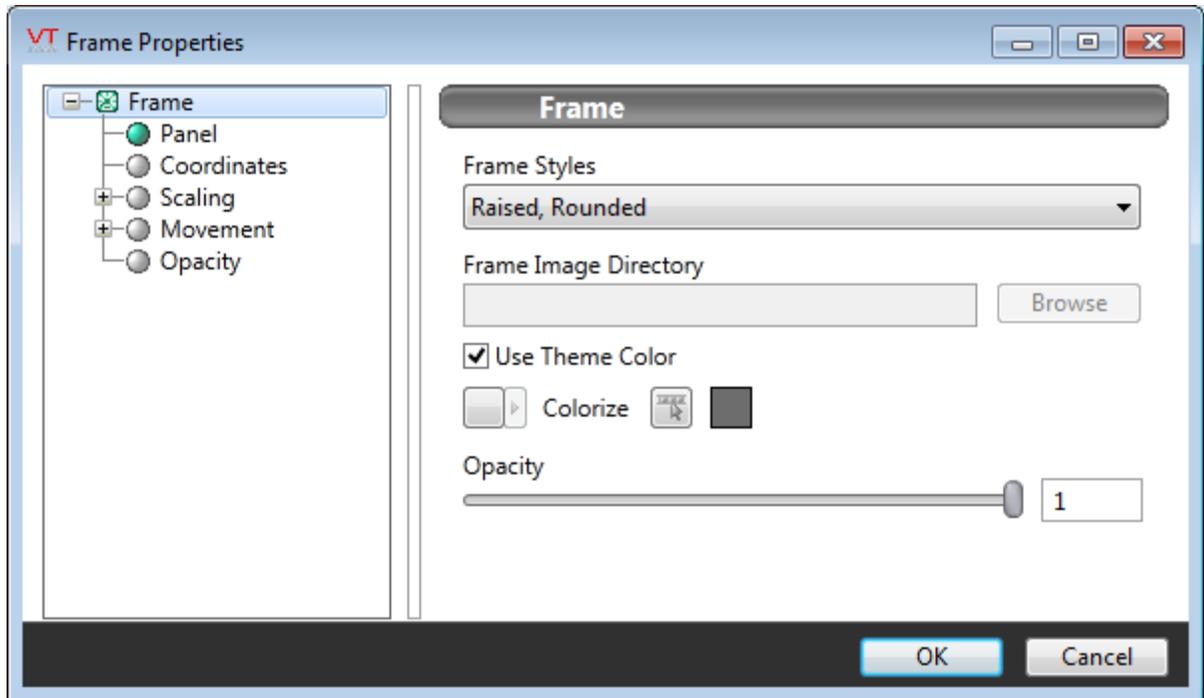
Frame Styles

Set the overall appearance of the frame by selecting from one of the nine built-in options, or choose your own frame image. Images for the built-in frames can be found in sub-folders of C:\VTScada\VTS\Resources\Frames.
Each folder has nine images, setting the appearance of each corner, each side and the fill. Use these as examples if building your own frames, but add the new folder to your application's Bitmaps folder. Do not add or change files in the VTScada folders.

Use Theme Color

When checked, the Color Options will be adjusted for you such that your frame will take the appearance of whatever color theme is in effect for the application.

Colorize

Enabled only when the option, Use Theme Color is deselected. Colors may be set explicitly using the Select Color dialog, or they can be linked to tag value, page parameters, linked tag property or expression.

Opacity slider

Controls the overall visibility of the frame. An Opacity of one is fully visible and a value of zero is fully transparent.

Related Information:

...Border

...Folders

...Rounded Rectangle

Lighting Overlays / Tiles & Textures

These are images, used and configured the same as pump, motor and valve images.
Lighting overlays are often used to enhance other objects. (Examples can be seen in many of the gauge widgets, and in the Dashboard A sample page.) Tiles and textures are often used as a background fill. (All of the Dashboard sample pages make use of a textured background.)

Related Information:
VTScada Graphics Library

Rounded Rectangle

A rounded rectangle is one with a user-specified radius for the corners. Each corner can have its own radius value.
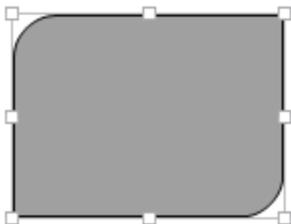
The fill and the border can be independently configured.

The Properties Dialog for the Rounded Rectangle:
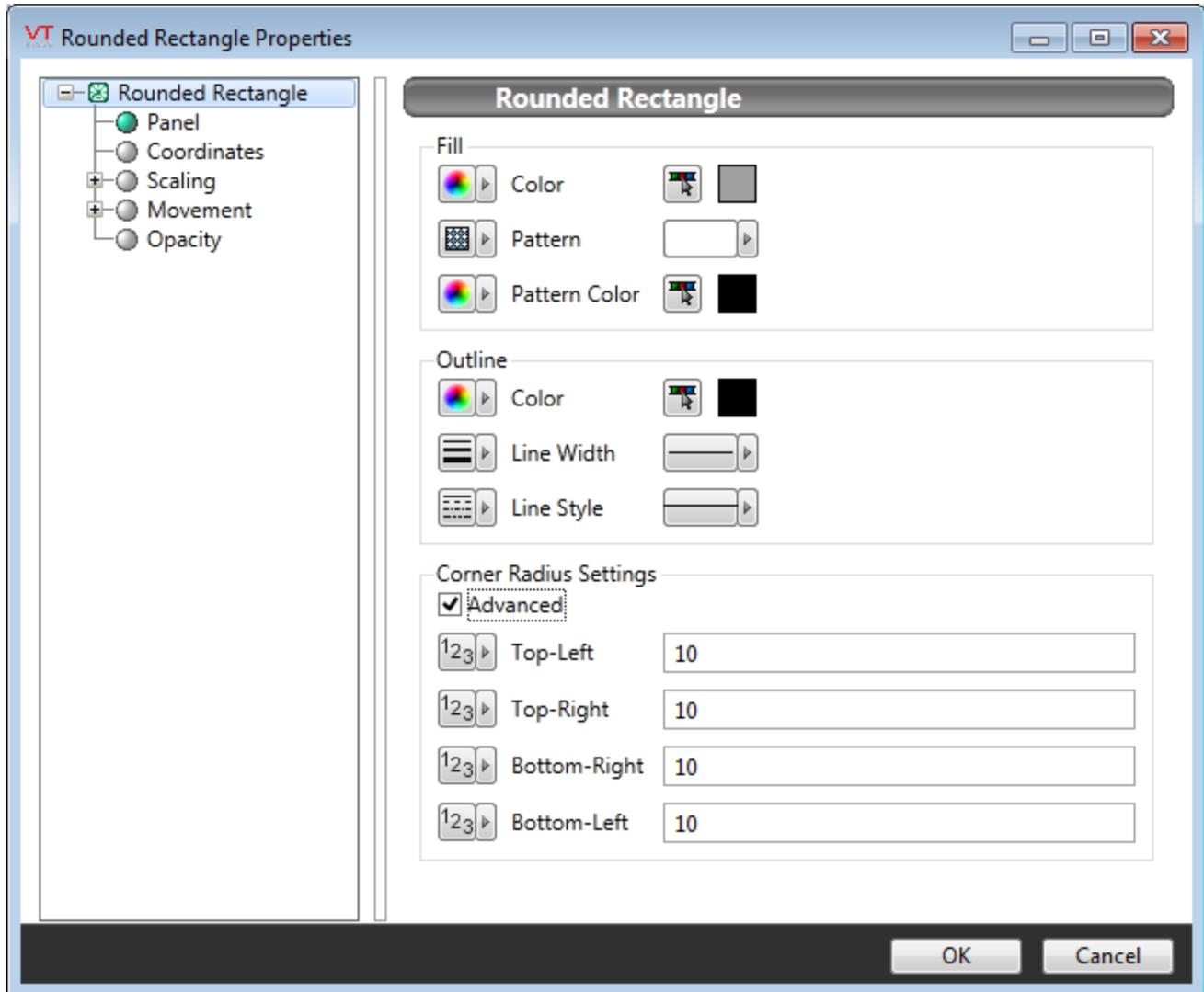
Fill

    Color

        Use the Select Color dialog to choose the color of
        the rectangle's interior. Set the color to transparent
        (0% opaque) to remove the fill.

Pattern and Pattern Color

Select any of the 25 predefined fill patterns supplied with VTScada.
For each of the sample patterns, the black lines or dots represent the portion of the pattern that will be affected by a choice of pattern color. The space between the lines or dots of the pattern will be filled with the base color chosen for the fill.

Outline

Color

Use the Select Color dialog to choose the color of the outline border. Set the color to transparent (0% opaque) to remove the fill.

Line Width

Measured in pixels, use this control to set the width of the outline boundary. Changes to the line width do not alter the size of the rectangle; additional pixels are added inside the outline, overlapping the fill.

Line Style

Select any of the five line types supplied with VTScada.

Corner Radius Settings

If the basic mode is selected, one data entry field is used for

all four corners. Select the advanced mode in order to set independent values. Note that basic mode will be selected by default if the rectangle is square.

Each radius is measured in pixels. Use a value of zero for a right-angle.

Values less than zero or greater than 999 cannot be entered directly, but will be accepted from expressions or tag values.

Setting a radius value less than zero or greater than the dimensions of the rectangle can produce interesting (and potentially useful) shapes.

Related Information:
Frame
Border

Tool Tip Widget

Not linked to any tag.
* Does not use the Stye Settings tag.
The Tool Tip widget is used for adding extra information to a screen that should not be displayed unless requested. The widget is drawn as a rectangular area. When an operator moves the pointer over the area, the information that you configure will be displayed.

The properties dialog for the Tool Tip widget:
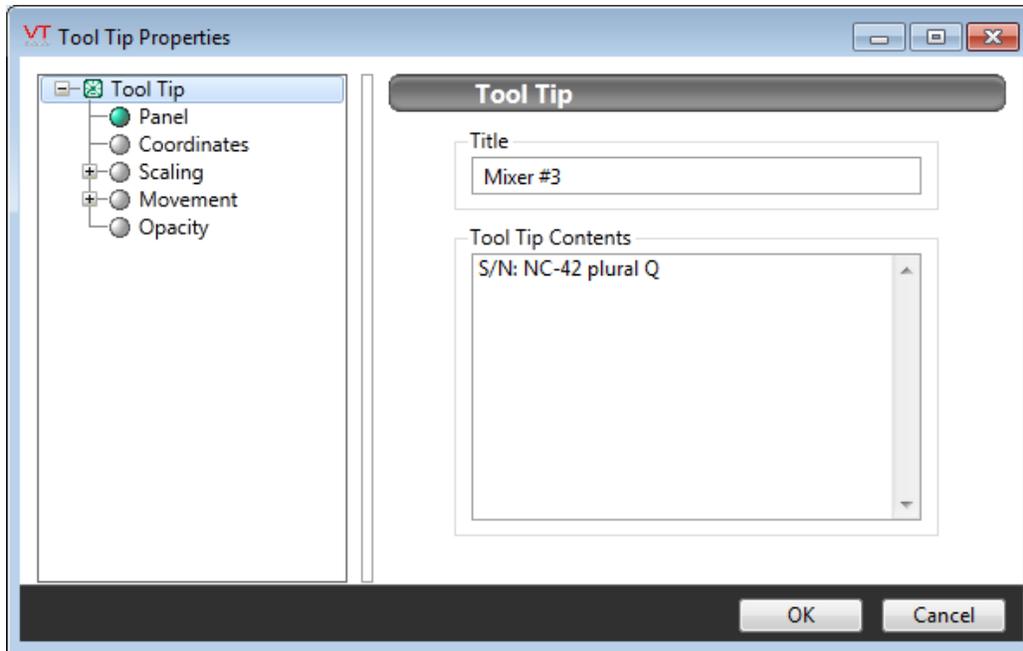
Title

Text to be displayed in bold above the body of the tool tip.

Tool Tip Contents

Usually, the greater part of the information to be displayed.

Both the title and the contents are optional. You may choose to display only the title in bold, or only the contents.

Widget Folders:
Basic Components\Specialty\Misc\
Tools\Standard Libary

Site Legend

Not linked to any tag
* Does not use the Stye Settings tag.
The Site Legend widget is a static image, used to provide a reference for the colors in a Site icon. While not technically a widget, it is included with this group since it is so often used on pages that include Site icons.
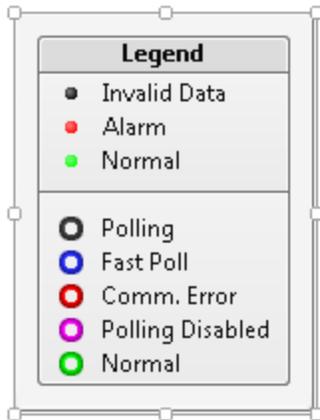
There are no configurable properties other than those common to any widget (scale, movement, opacity).

Widget Folders:
Basic Components\Specialty\Mapping\
Analytics\Diagnostics\
Tools\Site Tools Library\

Draw Shapes

Shapes are geometric objects. Like images, they are used to provide visual context for a page. Shapes include rectangles, ellipses, stars and more. All shapes have an outline and most have a fill, both of which can be configured independently as to style, width, and color. You can set either the outline or the fill to "none", removing that part of the shape from the display.

> Note: Shapes have no "Link" option to tie them to a tag. They are essentially static, although any property (movement, scaling, etc.) can be animated by linking it to a tag value or to an expression.
> The Style Settings tag has no effect on shapes and should not be linked to any property.

To add a shape to a page:

1. Open the Idea Studio and open the page, to which you want to add a shape.

2. Open the Shapes palette.

3. Drag a shape from the palette to your page.

4. Adjust the size, location and other properties of the shape as required.
   A format ribbon is available for each widget type.
   Advanced configuration details can be accessed by opening the Properties dialog of the widget.

Troubleshooting:

- The image is too big / too small.
  All images have an initial size, which may be wrong for your application. After the image has been placed, zoom in or out on the canvas until you can see the edges easily, then select the image and drag the corner grips to res-ize it.

Related Information:

...VTScada Graphics Library - Provide context to the widgets by including relevant images.

...Draw Widgets - Display tag values or provide operational controls.

...Ribbons - A description of the tools in each formatting ribbon.

Next Steps:

...Select and Group - Before changing the appearance of an object, you must first select it.

...Align - Tools and techniques for well-organized pages.

Building Blocks

All of the objects in the Shapes palette are considered building blocks. Each can be scale d, stretched, rotated, colored, and more after being dragged to a page.
All building blocks have an outline, and all except for arcs have a fill. The appearance is controlled by tools in the formatting ribbon, or by each shape's properties dialog.
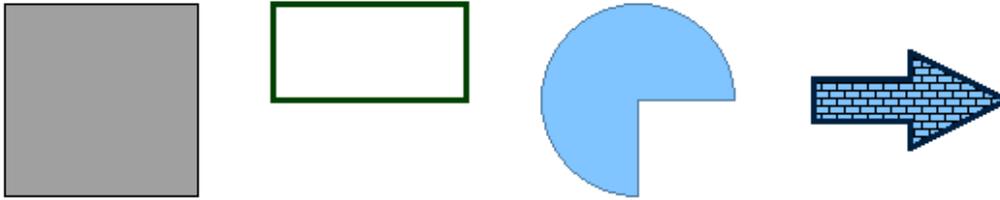
A variety of shapes, with a variety of outlines and fills.

Note: Shapes have no "Link" option to tie them to a tag. They are essentially static, although any property (movement, scaling, etc.) can be animated by linking it to a tag value or to an expression.
The Style Settings tag has no effect on shapes and should not be linked to any property.

Once drawn on a page, you can adjust the appearance of a shape using either the ribbon tools or the shape's properties dialog. The properties dialog for a rectangle, with the Advanced options opened is shown in the following example. All shapes will have a similar properties dialog.
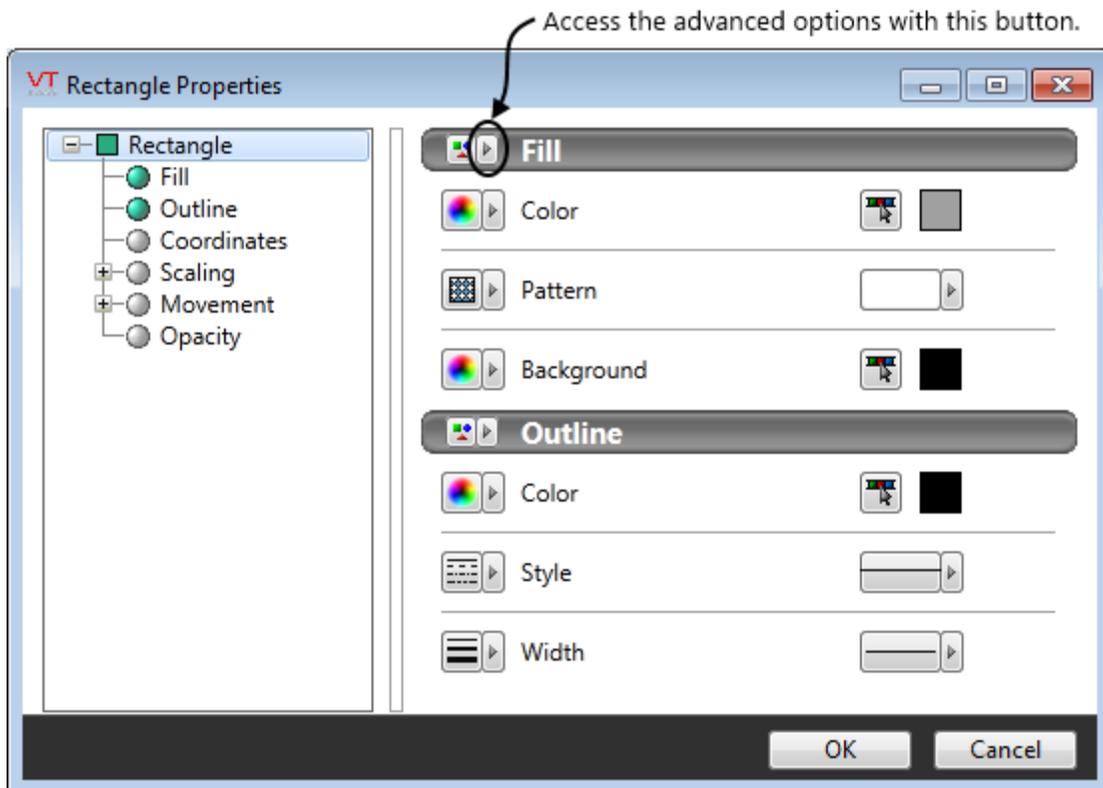
Fill – Color, Pattern, Background

Only color will be available unless you expand the advanced options. Click to open the Select Color dialog.

Click the pattern option to view a selection of shapes that can be used for the fill: brick, zigzags, etc.

Each shape is drawn using lines of the current color selection, against a background color of your choice.

Outline – Color, Style, Width

Only color will be available unless you expand the advanced options. Click to open the Select Color dialog.

The style option can be used to select the pen style, (solid or dashed), that will be used.

The default width is one pixel, but may be increased to a maximum of 30.

Lines

Lines can be drawn in any color, and have any width from one to thirty pixels. By default, each new pipe will be black, and have a width of one pixel. Lines differ from pipes in that they do not have a gradient shading.

As you draw a line, each segment will snap to a multiple of fifteen degrees. To release the snap feature and draw at any angle, hold down the Ctrl key on the keyboard as you move the mouse.

Multiple segments that lie along a straight line will be merged into a single segment.
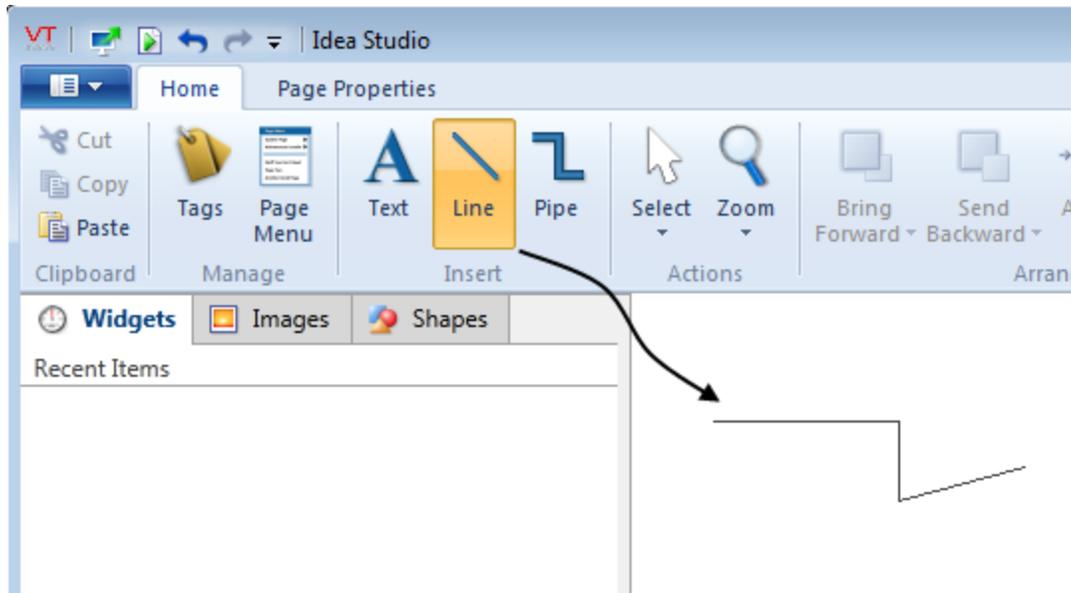
To draw a line:

1. Open the Home ribbon in the Idea Studio, and click the Line button.
2. Click once on the page, where the line is to begin.
3. Move the mouse and click again to mark the end, or the next corner.
4. Continue, adding as many segments as required.
   To remove points while drawing, press either the backspace key or the delete key, as many times as needed to remove vertices in the reverse order that you added them.
5. To stop, right-click after placing the final endpoint of the line.

Change a line's color and width:

Having drawn a line, you can adjust both the color and the width using the Outline button in the formatting toolbar. Note that lines do not have a fill property. Selecting the option, No Outline from the menu is the same as setting the outline color to be invisible.
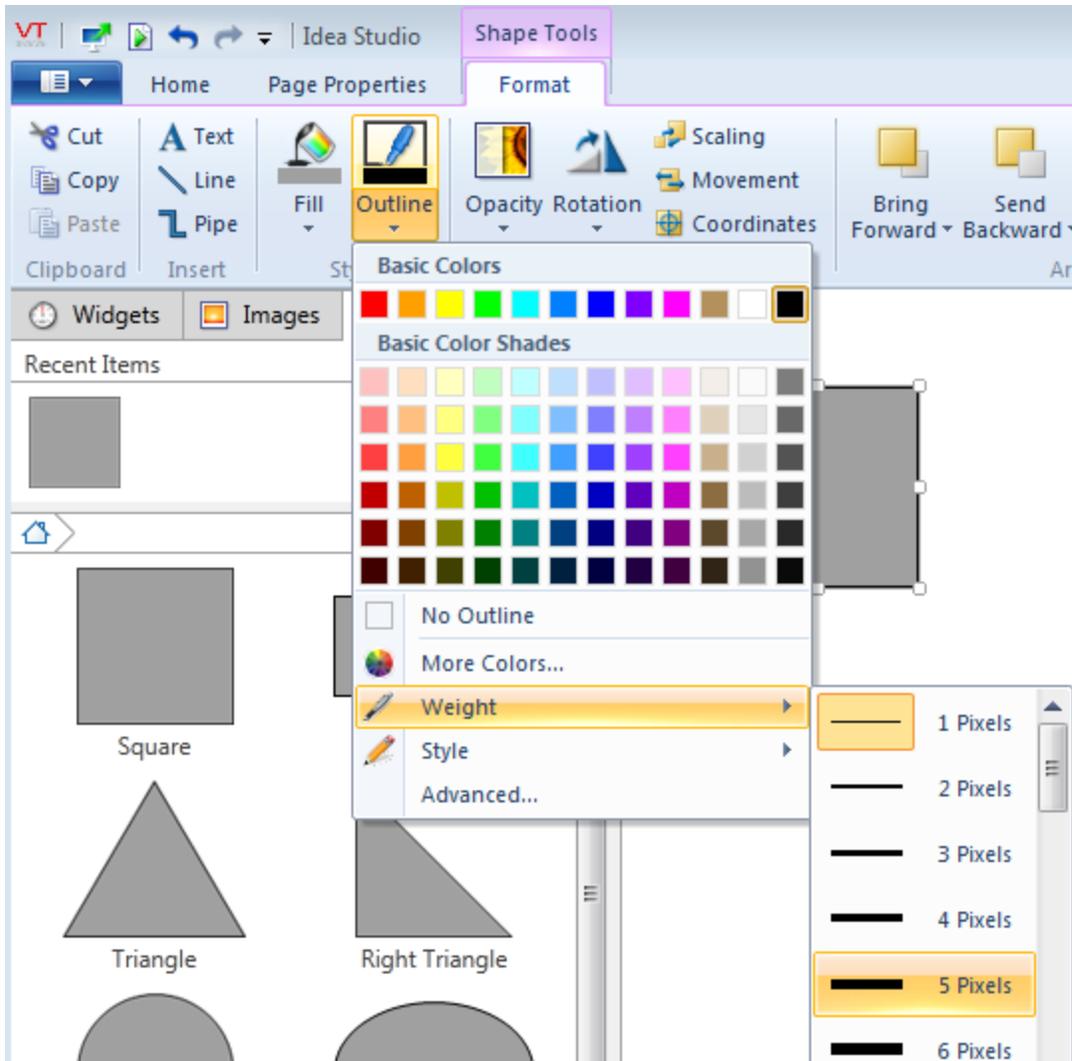
Relocate a line:

A selected line has two types of handle: a white square at each endpoint and a blue dot in the middle of each segment.

- Drag an endpoint handle to move that corner or end. If you straighten a path by dragging a corner into line between connecting corners, the handle will be removed.

- Drag a middle dot to add a new corner.

- Click and drag between handles to move an entire segment. Any connecting segments will stretch with the segment that was moved.

- To move the entire line, select it then tap the arrow keys to nudge the object one pixel at a time.

Related Information:

...Pipes

Pipes

Pipes can be drawn in any color, and have any width from one to thirty pixels. Each new pipe uses the color of the last pipe you drew in the current session.

As you draw a pipe, each segment will snap to a multiple of fifteen degrees. To release the angle snap feature and draw at any angle, hold down the Shift key on the keyboard as you move the mouse. Like other objects that you can draw, pipes respond to alignment points with other objects. Hold down the Shift key to release alignment snap.

Pipes do not have rounded corners, but after drawing a pipe you can add images from the extensive collection of pipe fittings, available in the Images palette.
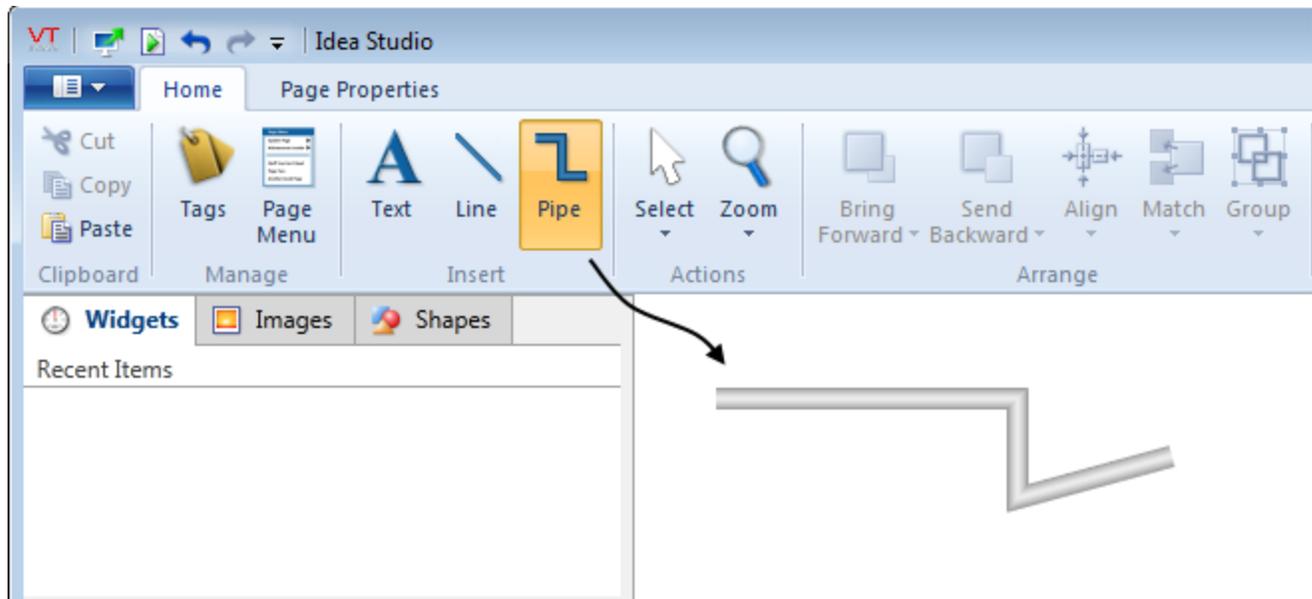
To draw a pipe:

1. Open the Home ribbon in the Idea Studio, and click the Pipe button.

2. Click once on the page, where the pipe is to begin.

3. Move the mouse and click again to mark the end, or the next corner.

4. Continue, adding as many segments as required.
   To remove points while drawing, press either the backspace key or the delete key, as many times as needed to remove vertices in the reverse order that you added them.

5. To stop, right-click after placing the final endpoint of the pipe.

Change a pipe's color and width:

Having drawn a pipe, you can adjust both the color and the width using the Outline button in the formatting toolbar. Note that pipes do not have a fill property. Selecting the option, No Outline from the menu is the same as setting the outline color to be invisible.
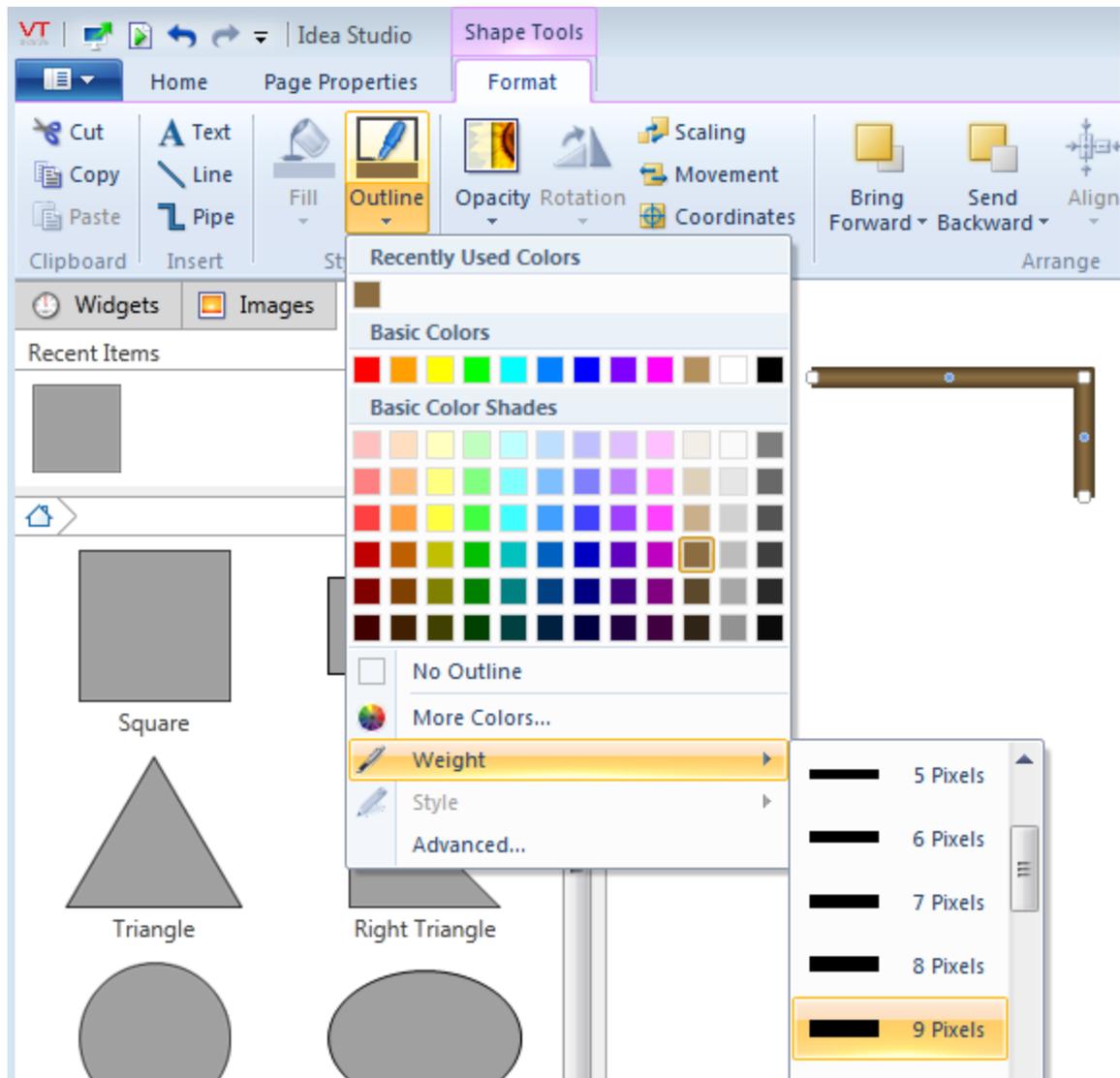
Relocate a pipe:

A selected pipe has two types of handle: a white square at each endpoint and a blue dot in the middle of each segment.

- Drag an endpoint handle to relocate that corner or end.
- Drag a middle dot to add a new corner.
- Click and drag between handles to move an entire segment. Any connecting segments will stretch with the segment that was moved.
- To move the entire pipe, select it then tap the arrow keys to nudge the object one pixel at a time.

Related Information:

Draw Text

Text is an essential part of any display. Titles, instructions, and labels are all required to help operators use the display screens that you build. While you may use the ribbon to format each label that you write, you are advised to make use of styles instead. By creating a small set of styles and applying them to the text that you create, you avoid the need to constantly return to the formatting commands, and you ensure that each type of label, title or instruction block will have the same appearance, exclusive of color, background, alignment ,and opacity.

To add single-line text:

1. Open the Idea Studio and open the page, to which you want to add text.
2. Ensure that the Home ribbon is selected.
3. Click the Text button in the ribbon.
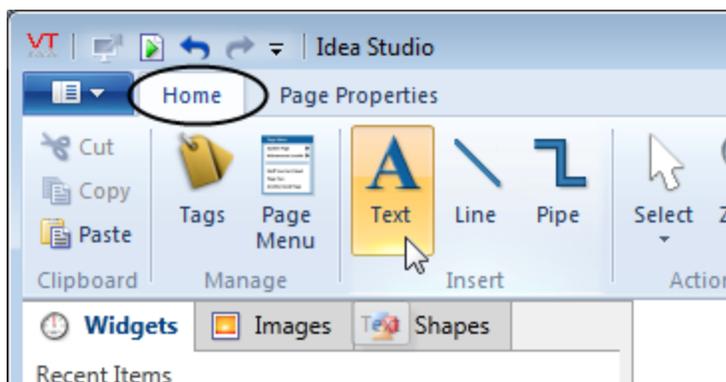
4. Click on the page, near where you plan to put the text.
   A selection box will mark the location on the page and the Edit Text dialog
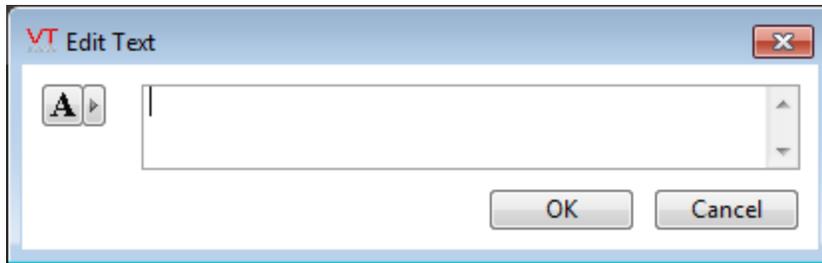
5. Type a single line of text, then click OK.

   Note that you may also select a page parameter or, if working within a tag widget, link this text to a property of the linked tag.

   The dialog can be re-sized as needed. Lines will wrap as required to fit the display space.

> Note: While it is possible to press <Enter> while typing, multiple lines in the Edit Text dialog will be merged together into a single line on the screen. Spaces will not be added to replace the line breaks.

To add multi-line text:

1. Open the Idea Studio and open the page, to which you want to add text.
2. Open the widgets palette.
3. Click to open the Tools folder, within the Widgets palette.
4. Click to open the Standard Library, within the Tools folder.
5. Drag the Multi-Line Text widget onto the page.
6. Open the widget's properties dialog.
7. Type the multi-line text in the field provided.
8. Adjust the display properties as required.

> Note: The text will wrap within the bounding box as required to fit, and it will use the line breaks that you type.

Troubleshooting:

- Line breaks in the Edit Text dialog were ignored.
- Words that were on two lines in the Edit Text dialog have been merged into a single word.

   This is by design. If multiple lines of text are required, use the Multi-Line Text widget.

- When stretching the bounding box on the page, the text appears to be deformed.
  This is only a preview while re-sizing the bounding box. The text is controlled by its font and style definition and will not be scaled or stretched with its bounding box.

Related Information:

...Multi-Line Text Widget – Reference details for the Multi-Line Text widget.

...Text Formatting – Adjust the appearance of your text.

...Font Tags – Where font and other style information is stored.

Select and Arrange

It is usually the case that you will want to adjust things after you draw them on a page. They might need to be aligned with other objects, nudged a bit up or down, or have their display order changed so that they do not "hide" behind other objects.

In all cases, before you can change anything drawn in the Idea Studio, you must first select it

.

Select and Group

To arrange objects on a page, change their appearance, or link widgets to tags, you must first select them. There are many ways to do this. To treat the selection as a single object for the purpose of alignment, etc. you may choose to create a temporary group using the objects.

Note: The Multi Image Widgets panel may be configured to open automatically whenever a multi-image widget is selected. This may be inconvenient at times. You can disable this behavior within the Multi Image Widgets panel.

Click on an object.

A left-click selects the object. A right-click selects the object and opens the related context menu.

Any other objects that were already selected will be deselected. You can use this to cycle through objects that overlap.

Ctrl-Click and Shift-Click.

Both are the same. Hold the Ctrl key or the Shift key down while clicking on objects. Adds or removes objects to/from a selection set. Other objects that were already selected remain part of the selection set.

Note: Tip: Use this technique to choose the anchor[1] object within a selection set. The most recently selected object will be the anchor object.

Click on an empty spot and drag the cursor over one or more objects.

Sometimes, referred to as a sweep select. A right-to-left selection gathers only objects completely inside the window.

Right to left window.
Only the blue pump will be selected.
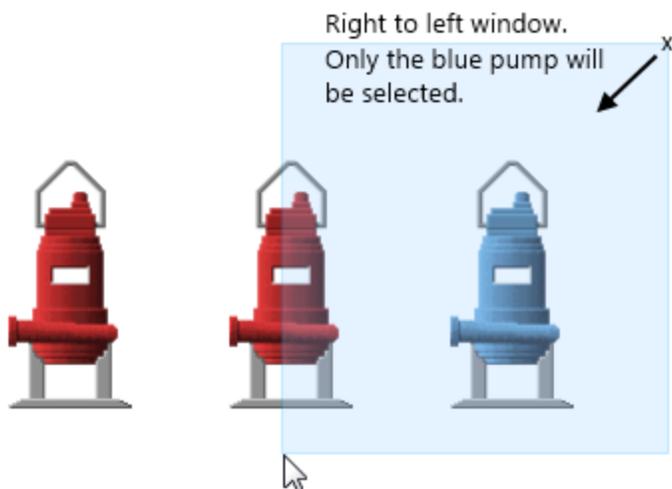
A left-to-right selection gathers every object it touches.

---

[1]One object within a selection set, indicated by a bold border. Tools to align and to match properties will use this as the reference object.

Left to right window. Both the red and the blue pumps will be selected.
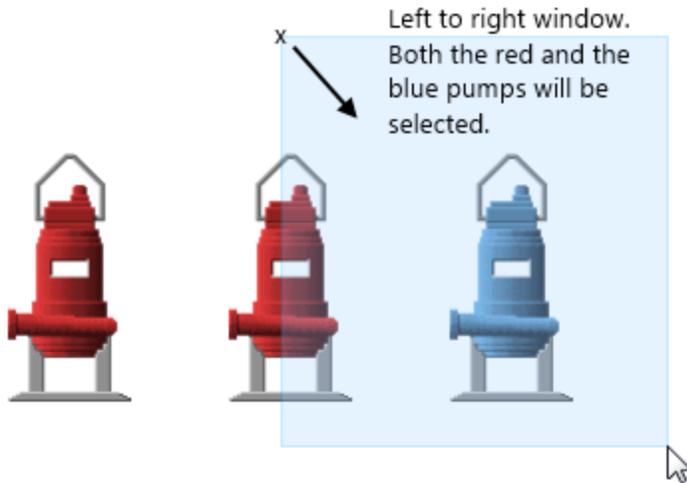
The Shift or Ctrl keys work with a window in the same way that they work with individual clicks. Objects will toggle between selected and deselected without any effect on the selection set outside the window area.

Press Ctrl-A on the keyboard
All objects on the page will be selected.

Use Select All in the home ribbon
This is the same as pressing Ctrl-A on the keyboard.

Use "Select of this type" in the home ribbon
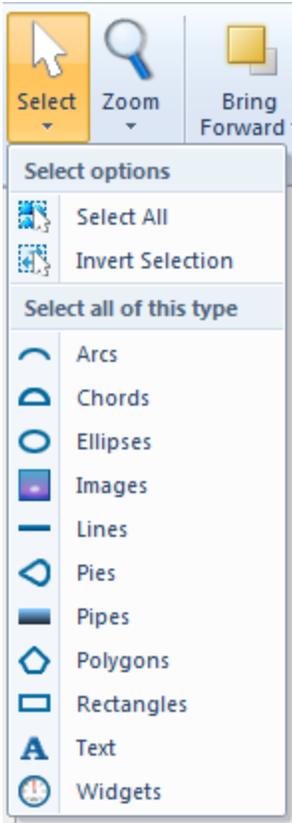
Selects all objects of the chosen type.

Use Invert Selection from the home ribbon.
In a crowded page, you may find that it's easier to select the objects that you don't want. If so, you can select those and use Invert Selection to swap them for the objects that you did want to select.

Groups

A group is a collection of objects, joined together for the purpose of selection and editing.

Groups do not have names and are not added to any of the palettes. You can copy and paste groups, but there is no list of groups to select from.

Use groups as an extension of the selection tools. Groups are very helpful to use when editing your pages. They allow a set of objects to be treated as a single item. In a crowded page, it is much easier to click on one part of a group to select the whole than to try to select each and every component.

Formatting commands apply to a group as a whole, but are somewhat more limited for the group than for the component objects. For example, a group cannot be rotated, even if the component parts could be.

Three shapes

One group
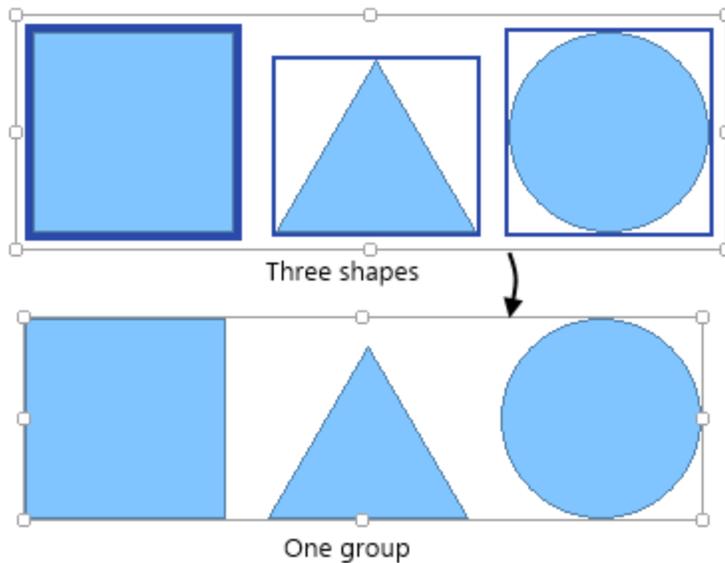
In essence, a group is a page drawn on another page. If you click the Edit command for a selected group, it opens as a page, where the individual components may be modified, and new parts added. For advanced developers, each group is stored in a .SRC file, using exactly the same format as found in a page or widget's source code file.
A group may be ungrouped as easily as it was first assembled. An "Ungroup" command can be found in both the toolbar and the right-click context menu for each group.

Snap Lines

The Snap Lines tool in the Format toolbar is used for two purposes:
You can enable or disable alignment snap with a click.
You can set the colors of the alignment lines and the default spacing using the Advanced mode.
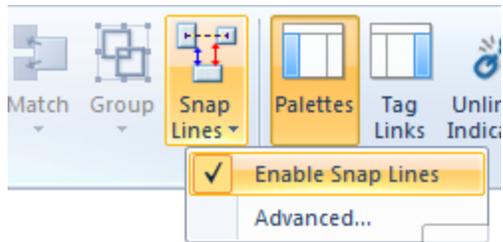
Note: When Snap Lines have been disabled, all automatic alignment features will be disabled, including proximity snaps to existing objects. Alignment guides will not appear.

The advanced mode opens the Snap Line Options dialog:
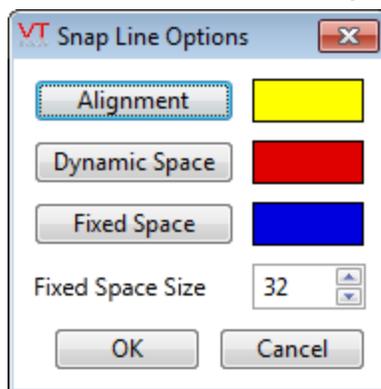
Alignment refers to the dashed line that indicates edge or center alignment.

Dynamic Space arrows appear when you are adding objects to an array and the new object is nearly at the same spacing of existing objects.

The Fixed Space is a user-controlled, default distance for adding new objects relative to existing objects.

Related Information:

...Align

Display Order
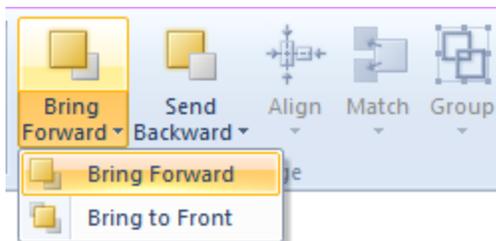
The toolbar commands, Bring Forward and Send Back, can be used to change which object appears to be in front of other objects. Referring to the image,

similar options exist for Send Backward as those shown for Bring Forward.

The most recently drawn object will always appear to be on top of other objects. This remains true as it or other objects are moved about the page.

When changing display order, you can choose to send objects all the way forward or back, or you can move them one step at a time, relative to other objects on the screen.

Green triangle at back   One step forward   Brought to front
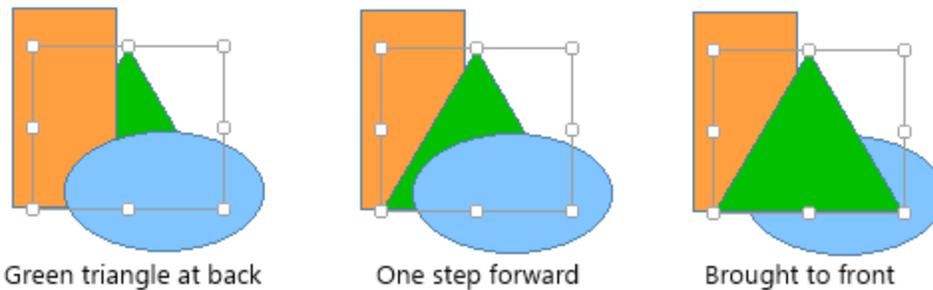
If there are several objects in the selection set, they will retain their display order relative to each other, and will be brought forward or sent back as a group relative to other objects on the screen.

Related Information:

...Align – Organize the display.

...Match Properties – Make other objects match the one that has the correct properties.

Align

This topic covers both alignment while dragging and alignment of objects within a selection set.

The procedures apply to Widgets, Images, Text, and Shapes.

Note: To disable alignment snap temporarily, hold the Ctrl key down while dragging. Use the Snap Lines tool to enable or disable alignment snaps.

Align while dragging

1. While you drag an object across a page, alignment lines will automatically appear, matching an edge or center of the object you are dragging to the edge or center of existing objects.
2. When aligned with two existing objects, or more that are equally spaced, red distance alignments will indicate when the object being dragged is at an equal spacing with respect to the existing objects.
3. Click when the alignment line indicates that the object you are dragging is properly aligned with an existing object.
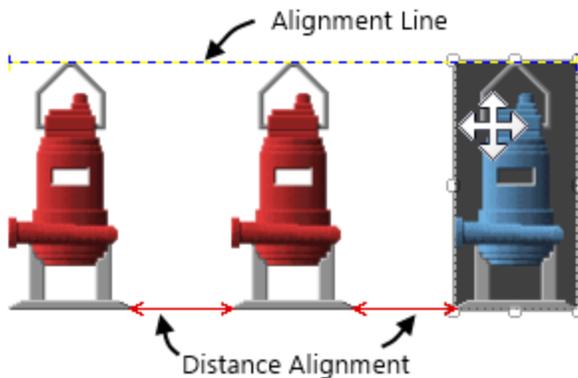
Notes:

- Automatic alignment works both horizontally and vertically.
- Alignment snaps work for the top, bottom and center of any object.
- When a group of objects is being moved, alignment snaps refer to the overall bounding box of all selected objects, not for individual objects within the group.

Troubleshooting:

- There are too many objects – alignment lines appear everywhere.
  Zoom in until fewer objects are visible.
- Alignment lines are not appearing as expected on existing objects.
  It is likely that the existing objects have been grouped. Alignment lines treat the group as a whole, rather than finding the individual elements within the group.

Align Selected Objects

Tools are provided to align a set of selected objects along a common edge or center. You can also space the selected objects equally, either horizontally or vertically.

1. Select two or more objects.
   One object in the selection will have a bold bounding rectangle. It is the anchor / reference point, to which the other objects will be aligned.

2. To select a different object as the anchor, hold down the Ctrl key and click it once to deselect, then again to re-select.
   The most recently selected object will be the anchor.

3. Click Align to open the menu of alignment options.

4. Click on the desired alignment option.
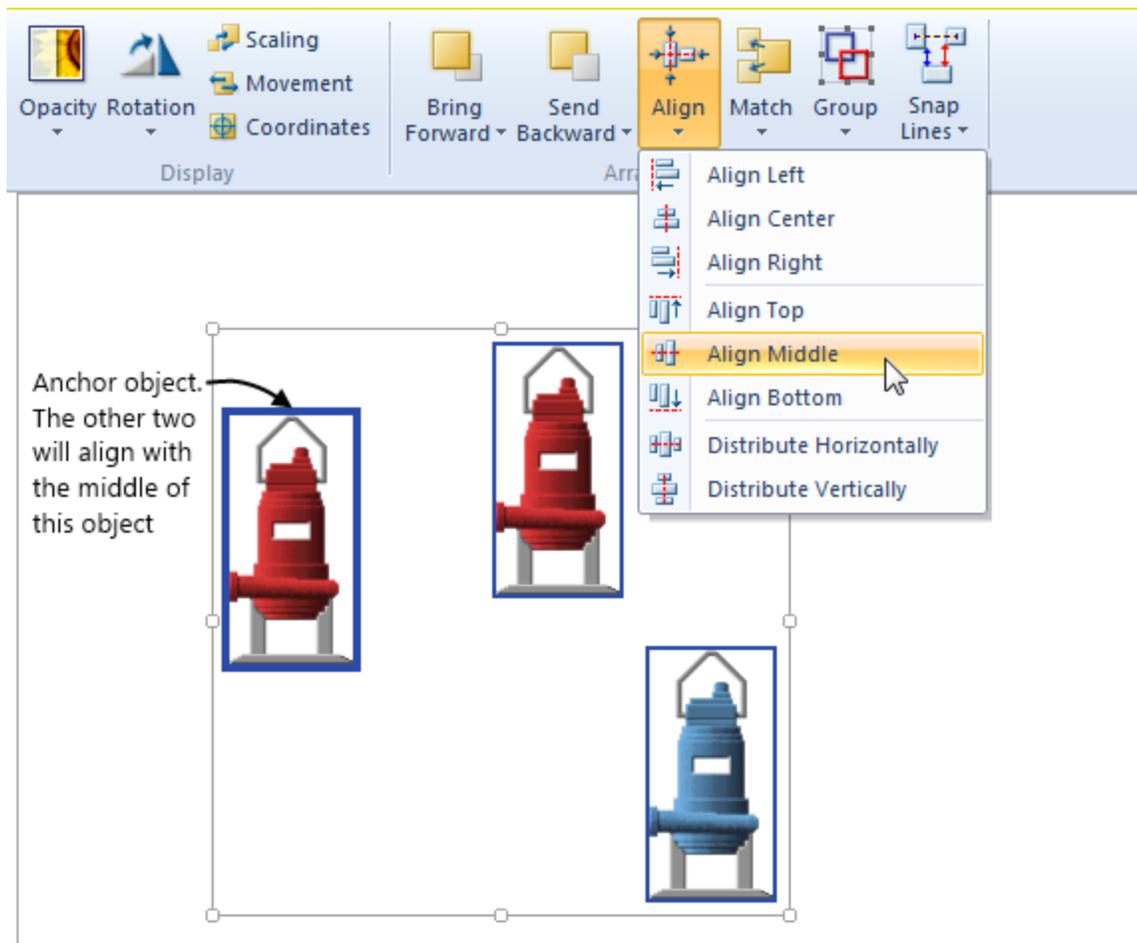
Troubleshooting:

- Unwanted objects are selected.
  A mouse-click will select each object in turn, whose bounding box is under the location of the click. If Ctrl is held down then all the objects will be selected with a single click.
  Use a window or crossing selection, or use Ctrl to deselect unwanted objects by clicking in an area covered only by their bounding box.

- Alignment did not happen as expected.
  It is likely that the selection includes objects that have been grouped. Alignment actions treat the group as a whole, rather than finding the individual elements within the group.

Related Information:

...Snap Lines – Enable or disable alignment snap lines. Control colors and default distances.

...Select and Group – Choose which objects to edit.

...Match Properties – Make other objects match the one that has the correct properties.

Move and Re-Size by Nudging

You will often want to move an object by only one or two pixels. Or, you may want the object to be a few pixels larger. Movement over such small distances can be difficult to control using only the mouse, even if you first re-size the display using the zoom feature of the Idea Studio. The solution is to use the arrow keys to nudge selected objects.

To move an object by nudging:

1. Select one or more objects.
2. Tap the arrow keys on the keyboard.
   The select object(s) will move by one pixel for each arrow key tap, in the direction indicated by the arrow key that you use.
3. Finish the action by pressing the Enter key, or by left-clicking anywhere on the Idea Studio canvas.

To resize an object by nudging:

1. Select one or more objects.
2. Hold the Alt key down while tapping the arrow keys on the keyboard.
   Hold both Alt and Shift to lock the aspect ratio while the object is being re-sized.
   The select object(s) will be re-sized[1] by one pixel for each arrow key tap, in the direction indicated by the arrow key that you use.

3. Finish the action by pressing the Enter key, or by left-clicking anywhere on the Idea Studio canvas.

[1]The base point of every object is its upper-left corner. It will be re-sized relative to that point. Therefore, it is always the right edge or the lower edge that will move in response to each nudge.

Re-sizing a pipe or line by nudging means changing its width.

Related Information:

...Movement

...Scale

Coordinates

Any object can be sized or positioned using its coordinates. Dragging or stretching an object on the page will change the values shown in the coordinates dialog. The values refer to the object's bounding box which is always rectangular, regardless of the shape of the object.

Width and Height are measured relative to the top, left corner of the object's bounding box. Changing width and height will also change the values for the right and bottom coordinates. If you change the coordinates for any of the edges, the width and height will not remain the same unless the difference between opposite sides remains the same.
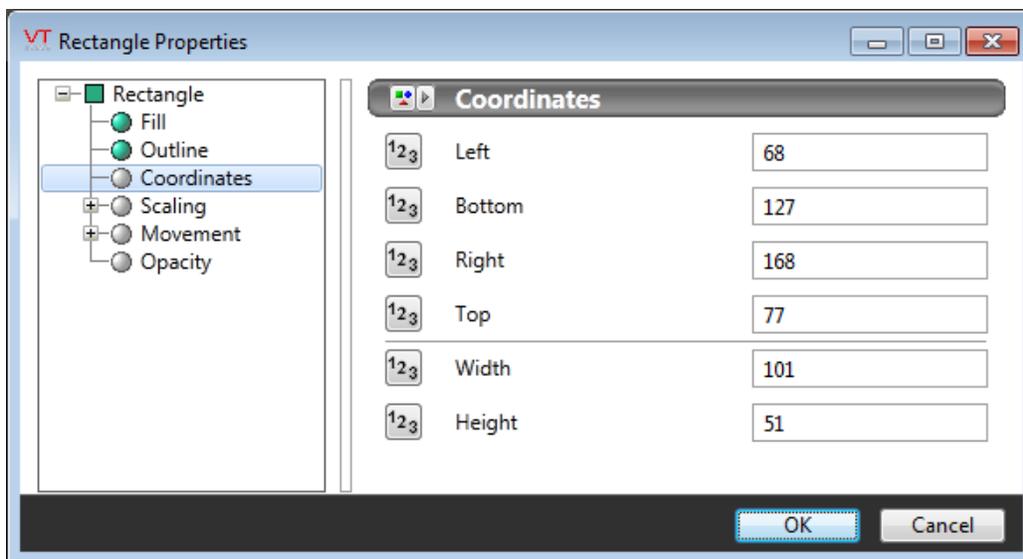
Formatting Tools

The Format ribbon of the Idea Studio is visible only when one or more objects are selected. A different format ribbon exists for each type of object, but the tools shown in the following image are common to most [1] ribbons.
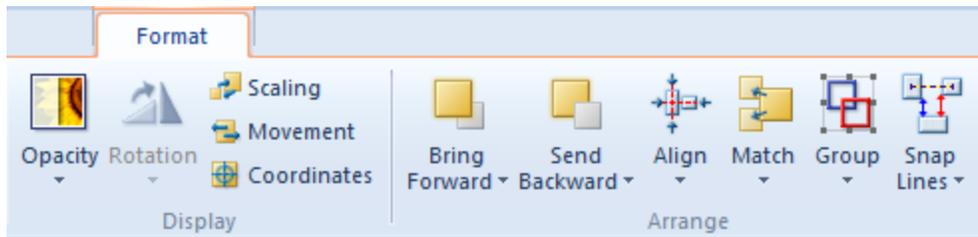
[1]Notes:

- Rotation cannot be used on all objects.
- Align, Match and Group can be used only when two or more objects are selected.
- With the exception of Scaling, Movement and Coordinates, changes apply to all selected objects that a tool may be used with.
- The tools Scaling, Movement and Coordinates will open the selected object's Properties dialog. If more than one object is selected, only the anchor object within the selection set will be affected.

Before looking at the details for using each of these tools, take a moment to review the techniques and shortcuts available for selecting objects. When working with several objects, you might save time by creating a temporary group before using a format command.

When two or more objects are selected, separate Format ribbons will be visible, one for each object type in the selection set. (Not for each object of the same type.) Format changes will be applied to all selected objects that can be affected by the change, regardless of the format ribbon used.

Note: In a selection set, the most recently added object is always the reference object as indicated by a bold border.
Use shift-click or ctrl-click to remove and add objects to the set, thereby changing the reference object.

> The object with the bold border is used as the reference (or "anchor") for alignment and sizing operations.

Related Information:

...Color Selector – Used to change the color of pages, widgets, shapes and images.

...Ribbons – A description of the tools in each formatting ribbon.

Opacity

The opacity of all VTScada objects can be adjusted, making them partially, or fully transparent.
The easiest way to adjust opacity is to select one or more objects, then click the Opacity button in the formatting toolbar. You can use any of the pre-set values for convenience, or click on Custom to open the slider dialog. A value of 100% is fully opaque, and a value of 0% is fully transparent, and therefore invisible.
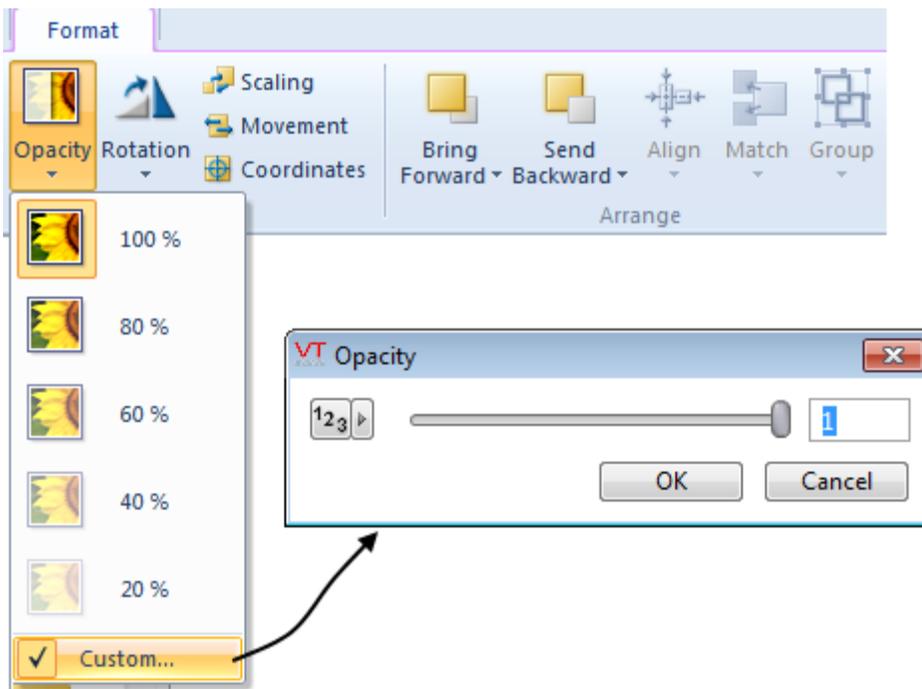
Advanced developers can use the data source button (left of the slider) to tie the opacity value of any object to a tag value or expression, thereby making the object fade or vanish in response to changing conditions.

Rotation

- Does not apply to lines or pipes.

- Shapes, images, and text can be rotated.

- Some, but not all widgets may be rotated.

- If editing a widget that includes images, use the tools in the Images ribbon rather than the Widgets ribbon.

Rotation is always done around the geometric center of an object's bounding box, and always proceeds clockwise in units of degrees.

If you have selected more than one object, each will be turned independently about its own center. Objects that had already been rotated will be reset to zero before the new angle is applied.

All rotation is relative to an object's zero-position, rather than cumulative. The Reset option will always bring an object back to its original orientation.
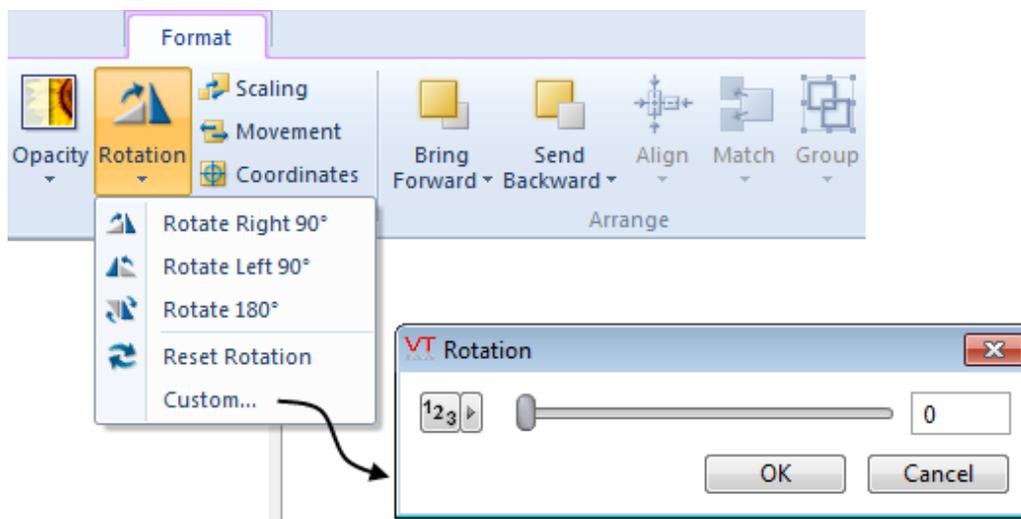
Advanced developers can use the data source button (left of the slider) to tie the rotation value to a tag value or expression, thereby making the object turn in response to changing conditions.

Widgets and rotation

Widgets themselves cannot be rotated, but if the widget uses one or more images (e.g. the equipment widgets and the indicator light widgets), then those images within the widget can be rotated to have the same effect. If this option is available for the selected widget, both the widgets ribbon and the images ribbon will be available. The Multi-Image Widgets panel will open automatically if the selected widget contains more than one image (and if you have not configured the Multi-Image Widgets panel to *not* open automatically).

Related Information:

...Selecting a Data Source

...Multi Image Widgets Panel

Scale

Scaling can be applied to any object, using its properties dialog. Note that, stretching an object by dragging its corners has no effect on the scale values. Rather, scale is applied to the object as placed (or stretched) on the page.

> Note: The result of scaling operations will vary depending on the type of object being scaled. For example, when scaling a pipe (must first be made part of a group or widget) the length of the segments will scale, but the width will remain a set property of the pipe. When scaling an image, the pattern will scale or not depending on the image configuration.
>
> Pages can be configured so that their contents will scale automatically as the page is resized. Refer to the link in the list of related topics.

Scaling can animated by being linked to a tag value or a VTScada programming expression. The result is similar to functionality built into the widgets, Top Bar, Bottom Bar, Right Bar and Left Bar. Scaling can be applied independently to any of an object's sides.

Scaling can be applied to only one object at a time. If more than one object has been selected, only the anchor object (shown with a darker highlight) will be affected. To scale a set of objects, group them first. When the properties dialog is first opened to the scaling options, all will be set to "None". Select the Advanced option for every edge that will have scaling applied.
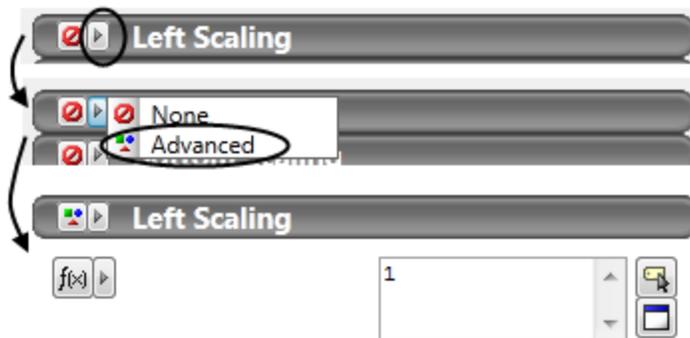
A scale factor of "1" is "100%".
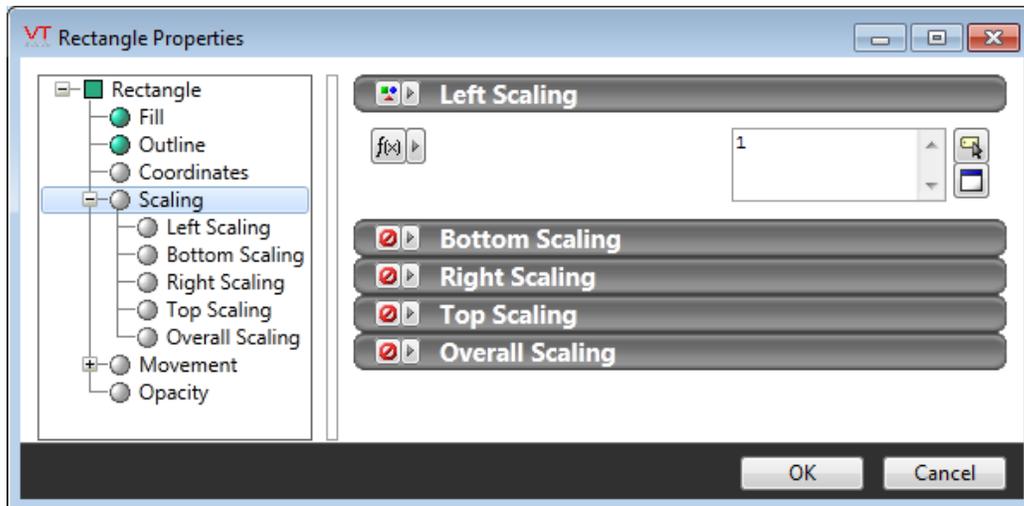
Related Information:

...Movement – Animate objects by tying movement properties to tag values or expressions.

...Coordinates – Pixel-level control of the size or location of an object.

...Select and Group – Group several objects into one before editing.

...Page Size Scaling and Control for Varying Displays – Apply ScaleDisplayContent to a page so that the contents will scale automatically as the page is resized.

Movement

Movement can be applied to any object, using its properties dialog. Note that, repositioning an object by dragging it on the page has no effect on the movement values. Rather, movement is applied to the object as placed on the page.
Movement can be animated by linking it to a tag value or a VTScada programming expression. The result is that the object will travel across the screen in response to changing conditions. For example, a numeric display of a holding tank level that moves up and down the tank, staying even with the fluid level.

Movement can be applied to only one object at a time. If more than one object has been selected, only the anchor object (shown with a darker highlight) will be affected.
When the properties dialog is first opened to the movement options, all will be set to "None". Select the Advanced option in order to view the controls.
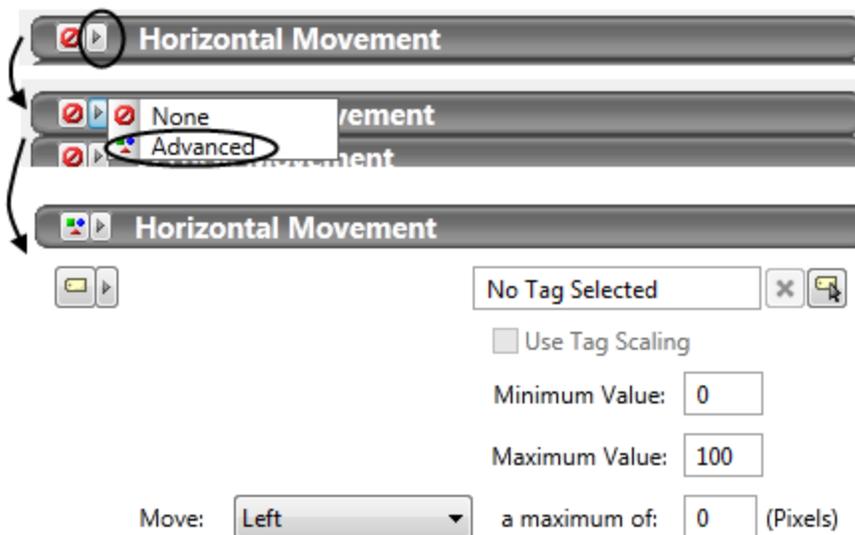
The default option is to select a tag, whose value will be monitored across its scaled range. The "Use Tag Scaling" check box should normally be selected. The object will move from its original position, either left or

right as selected using the drop down, in response to an increasing value of the tag being monitored. When the monitored tag is at its lowest scaled value, this object will be in its original position on the page. When the monitored tag reaches its maximum scaled value, this object will move by the number of pixels you specify.

Related Information:

...Scale – Change the size of objects. Scale can be animated by tying this property to a tag value or an expression.

Outline Properties

For any object that has a line or outline, you can modify the appearance using the properties dialog.
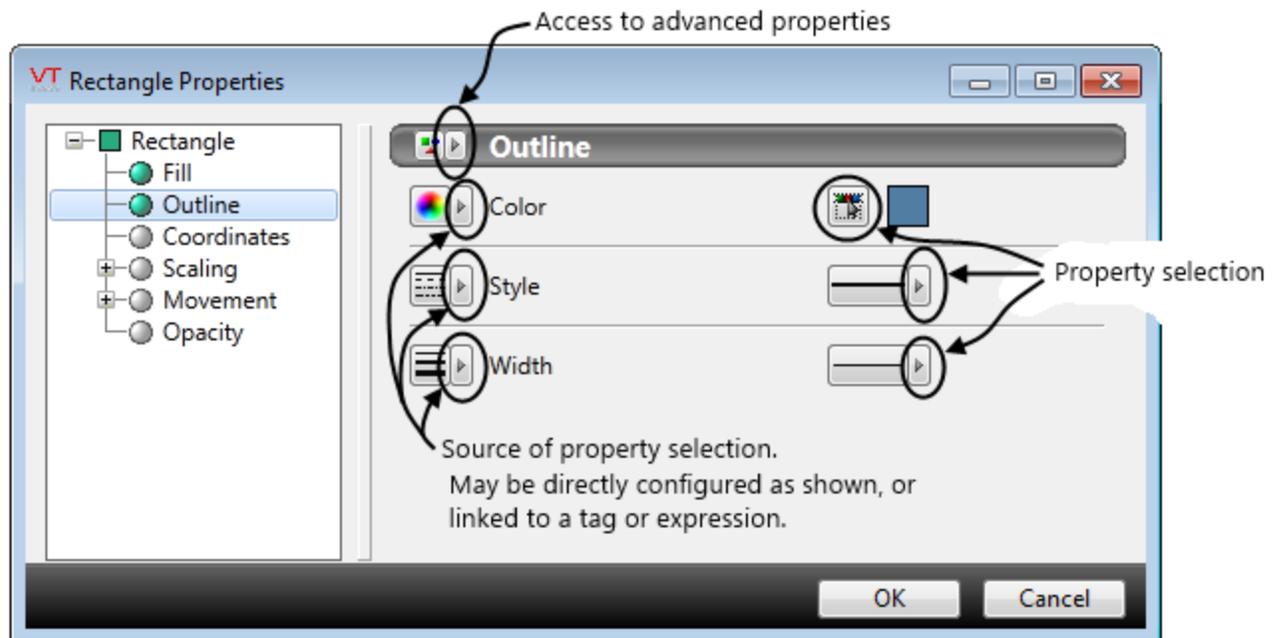
The properties dialog will often open in the "Simple" format. Use the selection at the top of the dialog to access the Advanced format so that you can view and change all the available properties of the line or out-line.

The standard method of setting the property values is to use the select-ors to the right of each property. The color button opens the Select Color dialog, and the width and line style selectors open drop-down menus of options.

You can link property values to tags or expressions, thereby animating any property as it responds to changing conditions in the application.

Related Information:

...Selecting a Data Source

Fill Properties

For any enclosed object, you can modify the appearance of the fill using the properties dialog.
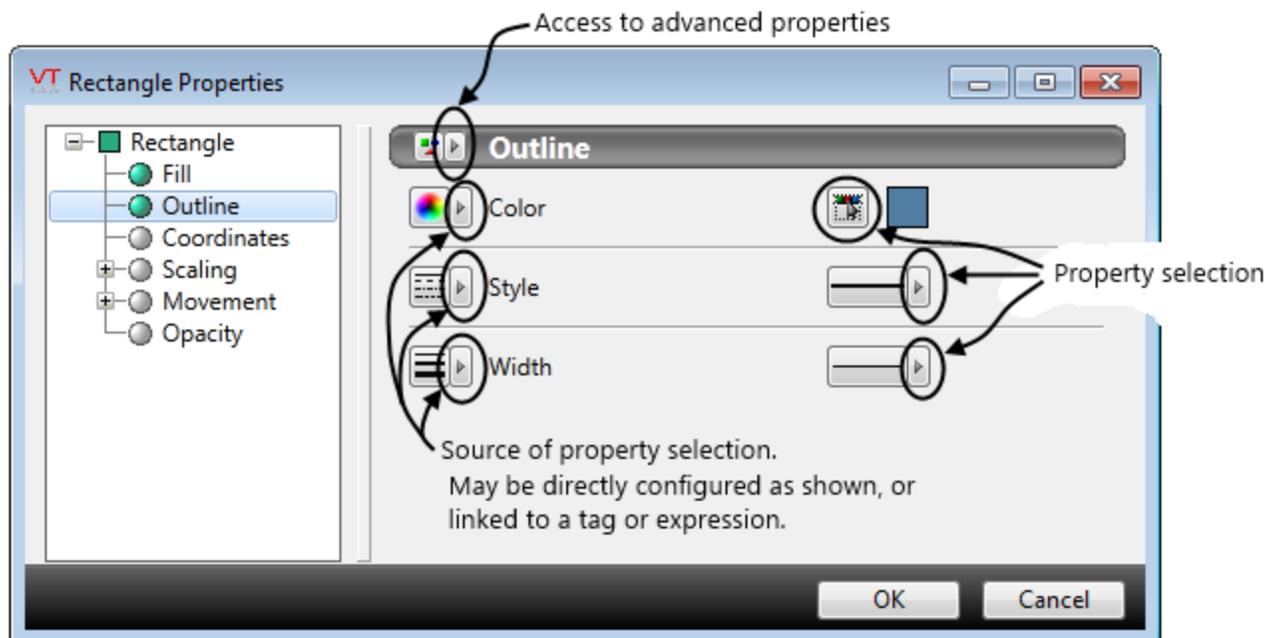
The properties dialog will often open in the "Simple" format. Use the selection at the top of the dialog to access the Advanced format so that you can view and change all the available properties of the line or out-line.
The standard method of setting the property values is to use the select-ors to the right of each property. The color button and background color button both open the Select Color dialog. The pattern selector opens a drop-down menus of options.
You can link property values to tags or expressions, thereby animating any property as it responds to changing conditions in the application.

Related Information:

Match Properties

If you have images, shapes and widgets that are uneven in size, trans-parency, scaling, etc. you can use the Match Properties tools to apply properties from one to others in a selection set.

Note: While Widgets, Images, Text, and Shapes may all be included in a selection set, not all of their properties can be matched.
For example, images cannot match a text font. Widgets cannot rotate.

To match properties:

1. Select two or more objects.
   One object in the selection will have a bold bounding rectangle. It is the anchor / reference object, whose properties will be applied to the other objects.
2. To select a different object as the anchor, hold down the Ctrl key and click it once to deselect, then again to re-select.
   The most recently selected object will be the anchor.
3. Click Match to open the menu of property options.
4. Click on the desired property to match.
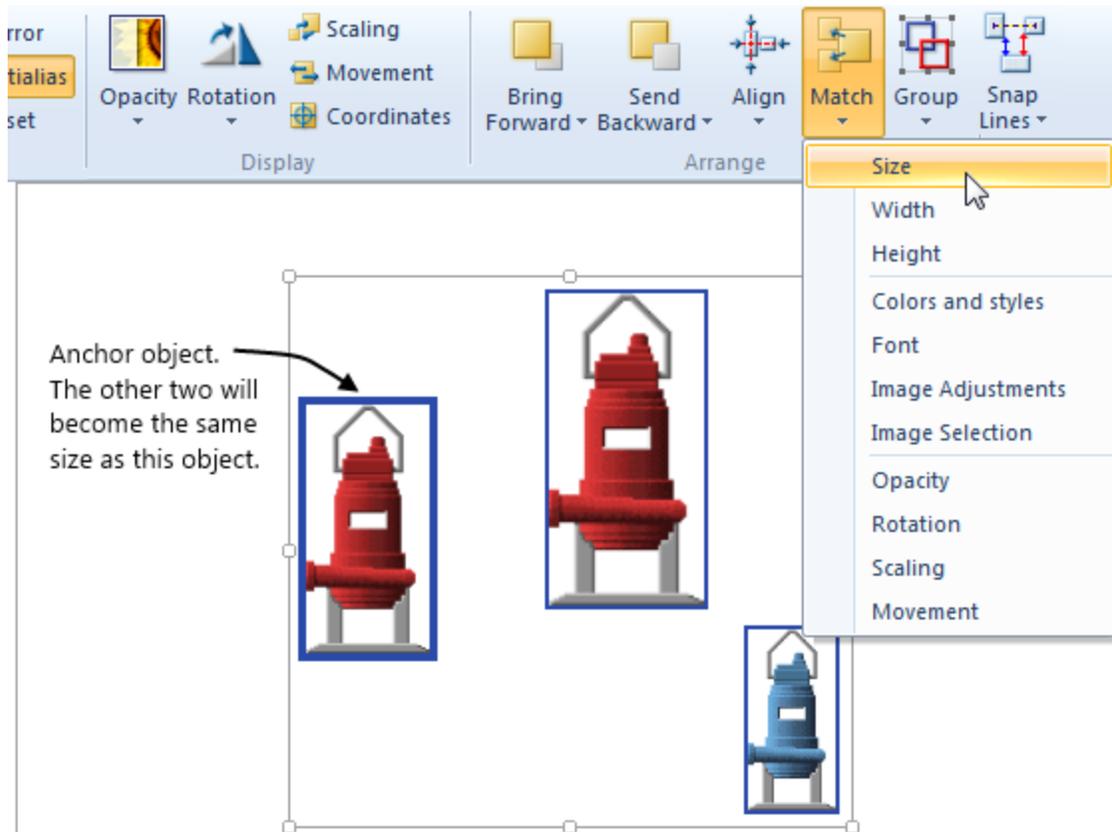
Troubleshooting:

- Unwanted objects are selected.
  A mouse-click will select all objects whose bounding box is under the location of the click.
  Use a window or crossing selection, or use Ctrl to deselect unwanted objects by clicking in an area covered only by their bounding box.

- I chose to match color, but the blue pump remained blue.
  The pump might be blue because it uses a blue image rather than a gray image with a blue color applied. Use the Image Selection option of Match Properties to change the image used by this object.

- Not all of the objects rotated.
  Widgets cannot rotate.

- The wrong object's properties were used.
  Undo. Verify that the bold rectangle is around correct anchor object and match properties again.

Related Information:

...Select and Group – Choose which objects to edit.

...Align – Organize the display.

Color Selector

The color selector is used in all cases where the color of a page, a widget or a shape can be changed (excepting pipes).
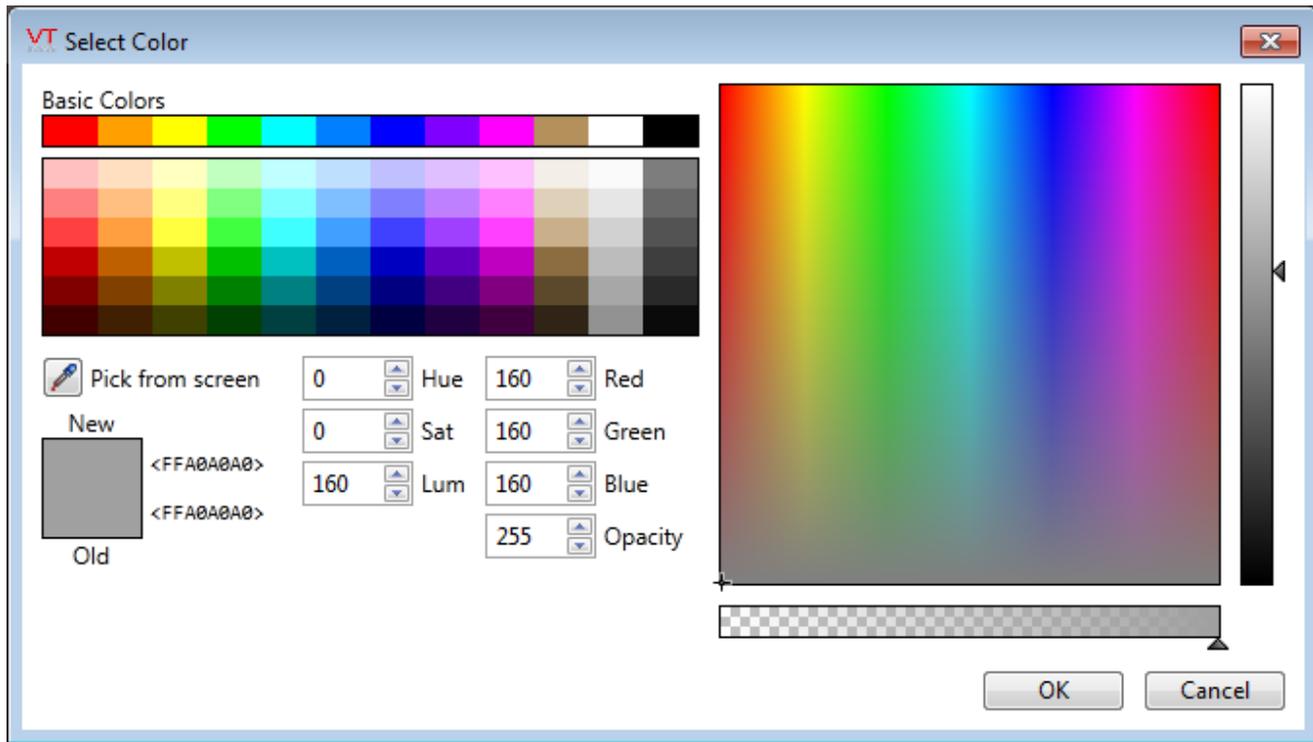
In a few instances, such as when changing the background color of a page, a miniature color selector will be displayed. The miniature version is effective for quickly selecting a color when the full range of options is not required. Use the More Colors button in the miniature selector to open the full color selector.
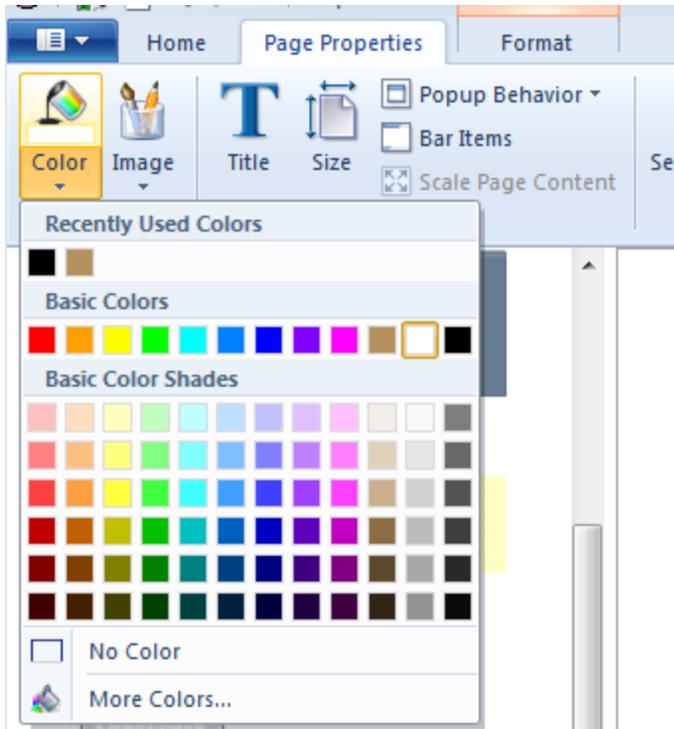
This topic focuses on the options in the full version of the color selector. For your convenience, the last several colors used are provided at the top left of the selector, allowing you to quickly match a new object to the color of an existing one.

A range of basic colors and matching shades is provided to help save you time when developing your applications. The second from the left is Active Orange, used with the Image Change widget.

The Pick Color tool works within any VTScada window, even thought the eyedropper image appears only within the Idea Studio. In other VTScada windows, clicking the standard pointer on any object will pick up that object's color for the selector.

Numeric color values may be provided using either Hue-Luminosity-Saturation or Red-Green-Blue values. You may also set an opacity value if you would like the object being colored to be partially or completely transparent.

The color sample area and sliders are designed to be intuitive to use.

Note: Color values are stored in code using hexadecimal **aRGB**[1] values, having the form #AARRGGBB for red, green, blue.

Related Information:

...Active Orange – Create a region that reacts to the presence of an Image Change widget.

...Transparent Black – Set areas in images that will be transparent when imported into VTScada.

...Image Change Widget – May be used to create a color swatch that is visible only over a patch of Active Orange, and that changes color according to a linked tag's value.

Active Orange

Note: Active Orange is not supported with .PNG format files. When creating an image to use in an Image Change widget, or selecting one to be covered by an Image Change with no image of its own, use the .BMP format.

If tiling an Image Change over a background image that contains areas of Active Orange, the Color Replace property of that background image must be set to Active Orange. That property cannot be set when an image is used as a page background.

The shade known as Active Orange, or Orange 241 has been designated to work with the Image Change widget.
An image change, drawn without a selected image, is a rectangular region that will change color as the linked tag's value changes. The region will be visible only where it is above something drawn using Active Orange (see note, above).

---

[1]A colour value, defined as four, two-digit hexadecimal values. Alpha – Red – Green – Blue. An Alpha value of FF is assumed if only the RGB values are provided.

Active orange is defined as color 241 in the VTScada palette, or the RGB value, R=255, G=160, and B=0, with full opacity. ("<FFFFA000>"). In the Select Color dialog, it is the second color in the row of Basic Colors.
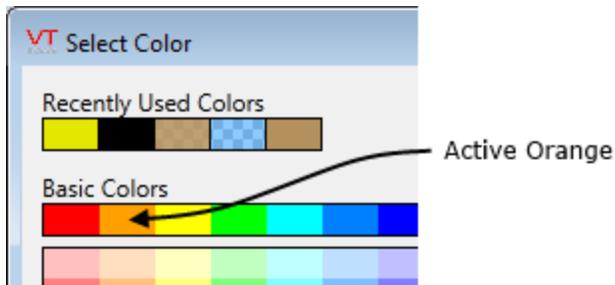
The following example shows a static image that includes an area of Active Orange. An Image Change widget, linked to a digital status tag, is drawn above the region. As the linked tag changes in value between state 0 and state 1, only the region of the Image Change above the Active Orange swatch will be visible and change color.
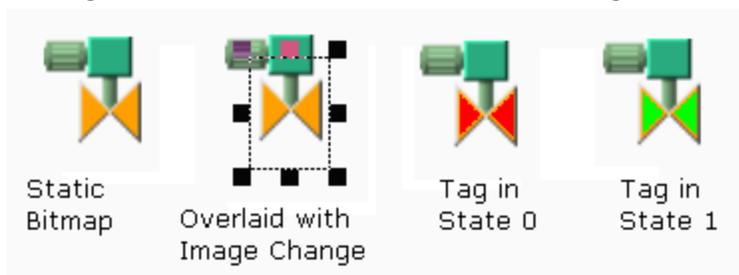
The dashed lines in the second image mark the boundary of the Image Change. It does not matter that covers a larger area than the orange triangles as only the portion on top of the Active Orange color will be visible.

Related Information:

...Transparent Black – Set areas in images that will be transparent when imported into VTScada.

...Image Change Widget – May be used to create a color swatch that is visible only over a patch of Active Orange, and that changes color according to a linked tag's value.

Transparent Black

In imported images that do not support transparency, areas that are pure black in color (RGB value, #000000) will be rendered as fully transparent in VTScada.

> Note: Transparent black is not supported with .PNG files or any other format that supports transparency. Save your image file using the .BMP format if you wish to use this feature.

Any parts of an image that are drawn using this color will be transparent when placed on a page or used as an icon.

To see an example of this at work, try opening any of the equipment graphics provided by VTScada using a graphic software program such as Paint. You will notice that all equipment objects actually have black backgrounds, however, when these images are placed on a VTScada application page, the backgrounds appear to be transparent. An example follows:
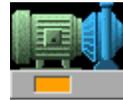
An image, with areas drawn using pure black:

The same image drawn on a white VTScada page:

To create your own custom graphics that include transparent areas, use the transparent black color wherever you want background objects on the page to show through.

Related Information:

...Active Orange – Create a region that reacts to the presence of an Image Change widget.

...Image Change Widget – May be used to create a color swatch that is visible only over a patch of Active Orange, and that changes color according to a linked tag's value.

Adjust Image Dialog

Can be opened using the Adjust button within the properties dialog of any image or widget containing images. Many of the tools within this dialog are also available in the formatting ribbons.

The Adjust Image dialog has two modes. Note that the advanced mode will open by default in some cases, such as when editing an image within an Image Change widget.

Note: The advanced mode includes the button, Reset to Defaults, which may be used to repair poor choices in configuration.
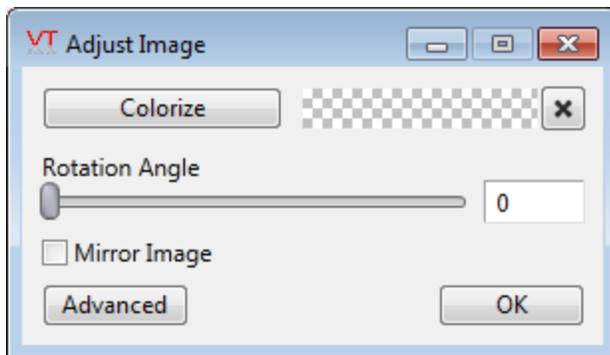
Basic mode
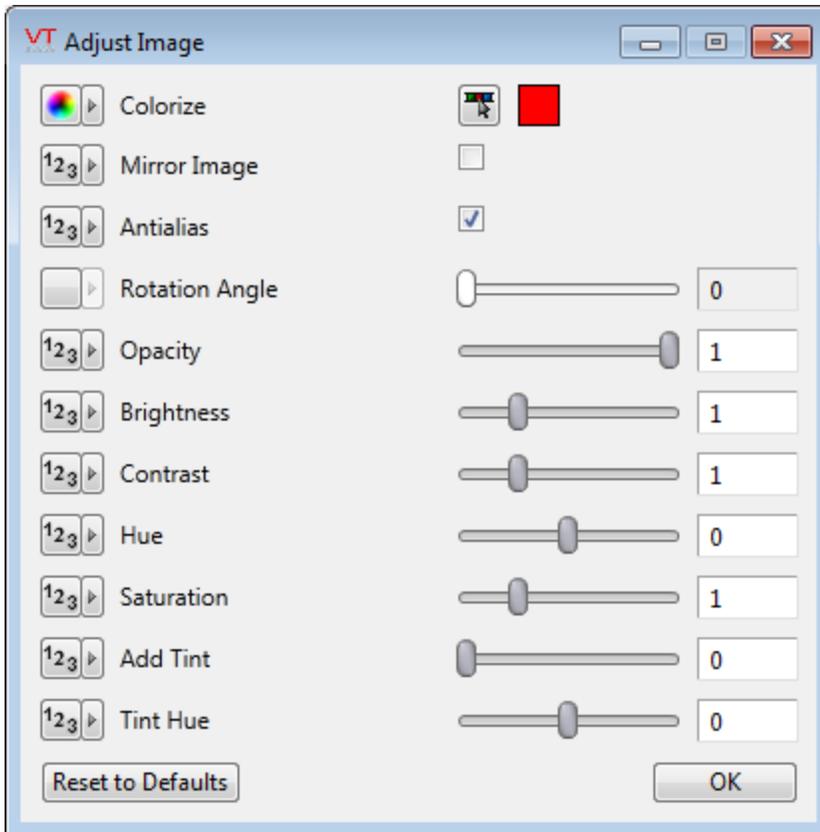
Advanced mode

Note: If linking a widget property to a tag value as part of an expression, you should use the [<TagName>] notation, specifying an absolute path reference to the tag. See the Programmer's Guide for Relative Tag and Property References

Colorize

Many images in the library are provided in a gray-scale format. You can use this to add vibrant (or other) color. Indicator lights, found within various widgets, make use of this technique to allow a plain gray disk to appear as a glowing red or green lamp.

With the Colorize sub-menu, you can select one of the pre-set colors to quickly add a tint, or you can open the Select Color

dialog to gain access to the full range of options for applying tint, hue, saturation, opacity, and other features.

Mirror Image

Flips the image from left to right. There is no vertical mirror option.

Antialias

Causes images that have jagged edges to appear to have smoother lines by feathering the edge. Note that the feathering may increase the size of the image by extending the boundaries.

Rotation Angle

Rotation is always done around the geometric center of an object's bounding box, and always proceeds clockwise in units of degrees.

Opacity

A sliding scale from fully transparent (zero) to fully opaque (1).

Brightness

A value of 1 (one) uses the image's original brightness. Possible values range from zero (black) to four (white).

Contrast

A value of 1 (one) uses the image's original contrast. Possible values range from zero (a uniform gray with no contrast) to four (very nearly washed-out).

Hue

Useful only if the image has a color other than gray. Values range from −180 to 180 and may be thought of as degrees around the circle of a color wheel, where zero is the starting point. By shifting the hue, you change the base color.

Saturation

Describes the intensity of the hue. In this dialog, saturation can range from zero to four, with one representing the originally defined saturation.
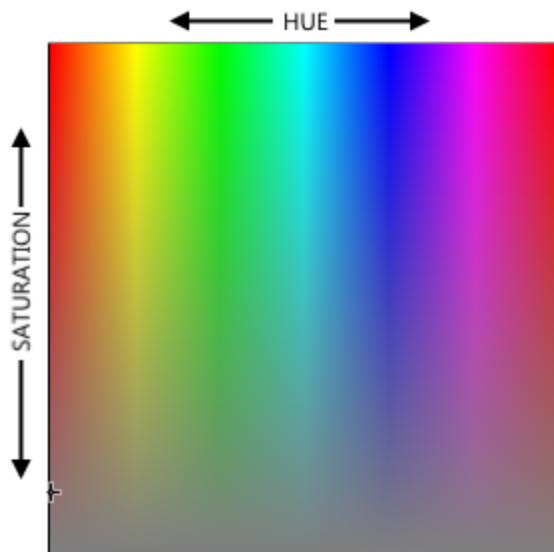
Add Tint

Where hue describes the color of the object, a tint can be thought of as a colored light shining on that object. This slider controls the intensity (saturation) of that light.

Tint Hue

Controls the color (hue) of the light being shone on the object. Like hue, this can range from −180 to 180, representing values around the color wheel.

Image Formatting

Any image that you have added to a page can be modified to suit your purpose. Beyond the usual display and alignment tools, all of the following can be used to adjust images. In all cases, the original image appearance can be restored by clicking the Reset tool. Formatting options change how an instance of the image is displayed. They do not change the image file.

Note: Images have no "Link" option to tie them to a tag. They are essentially static, although any property (movement, scaling, etc.) can be animated by linking that property to a tag value or to an expression. The Style Settings tag has no effect on images and should not be linked to any property.
For information about accessing image properties and linking them to tag values, see: Adjust Image Dialog
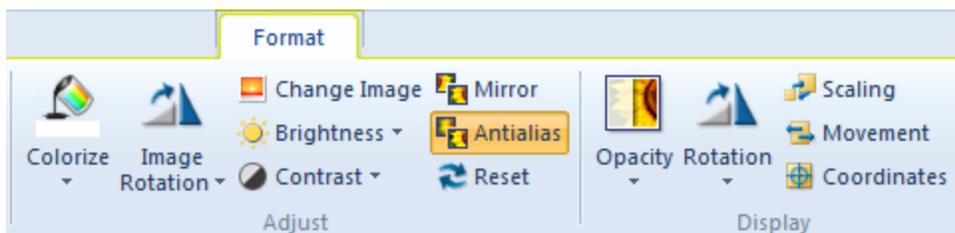
Color Replace
(Properties dialog only) Relevant only if the image includes a color indicator patch, or if within an Image Change widget, you have selected the No Image option. Color indicators within images are areas of [Active Orange](#).

Change Image
Opens the Select Image palette, from which you can select a new image to replace the one being used. Other formatting commands that had been applied to the original image will remain in effect for the new one.

Brightness

Adjust the brightness of the image using one of the pre-set values, or click Custom to open a slider. Brightness can be adjusted up or down. A value of 1 (one) uses the image's original brightness. Possible values range from zero (black) to four (white).
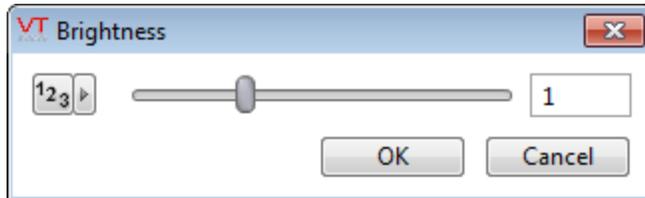
Contrast

Increase or decrease the difference between bright and dark shades within the image. At higher contrast, bright shades become brighter and dark shades become darker, making certain details easier to distinguish up to a point. At very low contrast, all shades approach a uniform intensity within each hue. Values range between zero and two, with one being the image's original contrast.
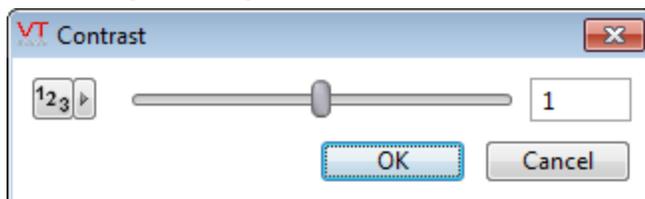
Both the brightness and the contrast dialogs include a data source button (left of the slider), which advanced developers can use to tie the property to a tag value or expression, thereby making the object change in response to operating conditions.

Colorize

Many images in the library are provided as in a plain, gray-scale format. You can use this to add vibrant (or other) color. Indicator lights, found within various widgets, make use of this technique to allow a plain gray disk to appear as a glowing red or green lamp.

With the Colorize sub-menu, you can select one of the preset colors to quickly add a tint, or you can open the Select Color dialog to gain access

to the full range of options for applying tint, hue, saturation, opacity, and other features.

When creating your own gray images, use color #888888 for best results. Gradients based around this shade will also work well.

Mirror

Flips the image from left to right. There is no vertical mirror option.

Anti-alias

Causes images that have jagged edges to appear to have smoother lines. You can also modify an image using its properties dialog. This has links to all of the tools found in the ribbon, plus one: the Tile and Scale Factor option.
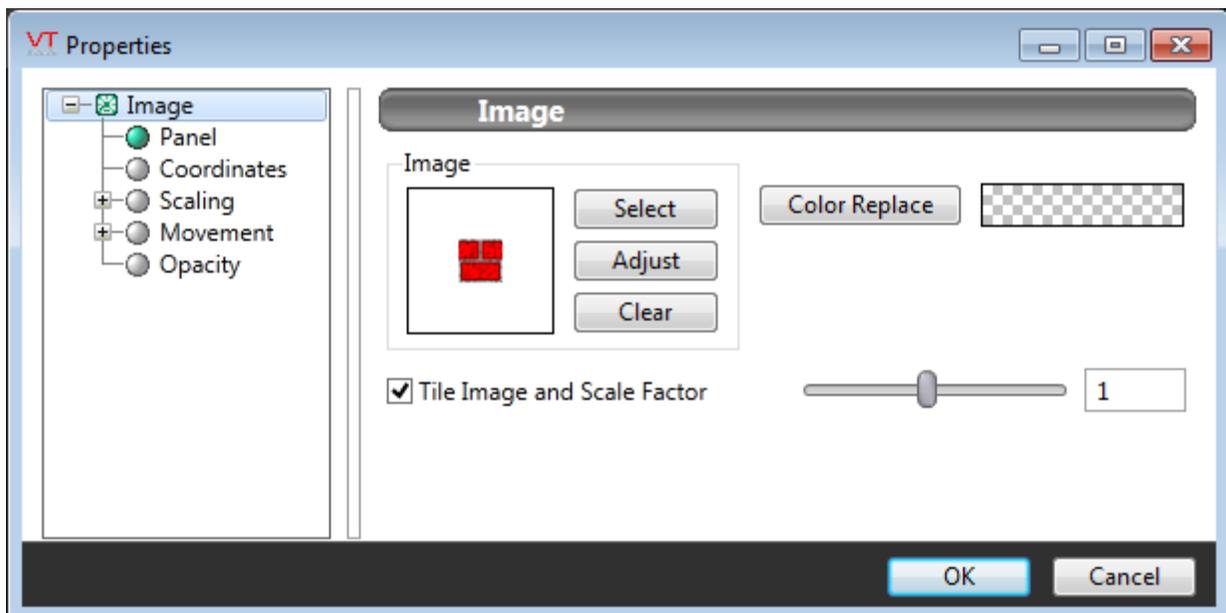
Tiled images

While any image can be tiled, this option is intended for use with backgrounds such as water, wood, stone and metal. When the Tile option is selected, the details show at their original size, and the image is cropped or repeatedly tiled to fill the area you have specified.

If the Tile option is not selected, the image is scaled to fit the specified area.

Two instances of the same image, one stretched to half
the size of the other.
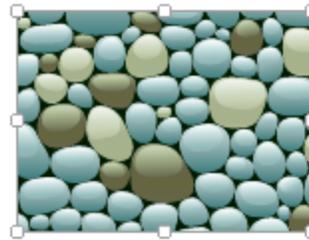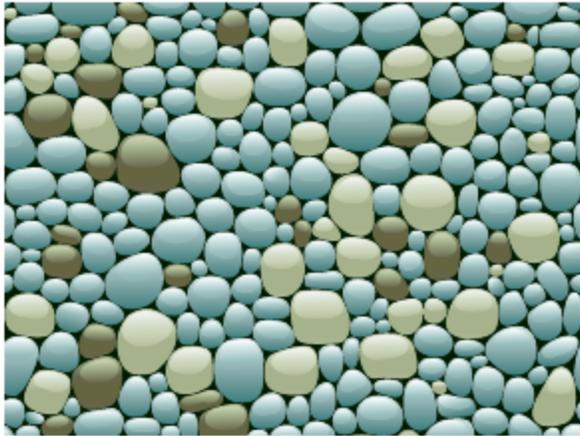
Tile option on
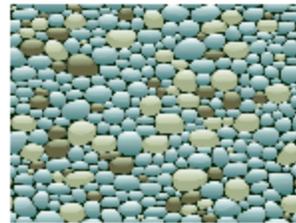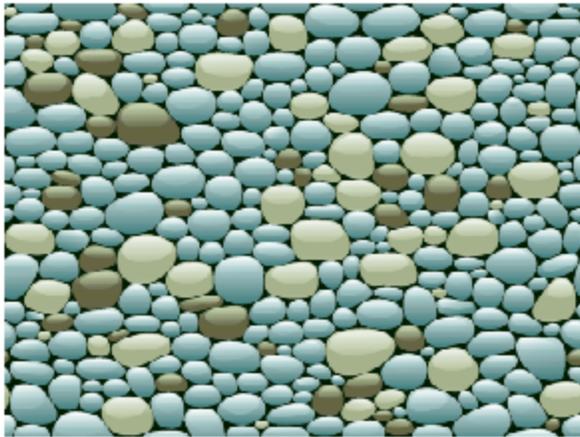
Tile option off

Related Information:

...Adjust Image Dialog

Related Tasks:

...Import Images

...Selecting a Data Source

Text Formatting

You have a varying amount of control over the appearance of text in your
application, depending on how the text was created.
Click once on existing text to select it and open the formatting ribbon.

Double-click on existing text to open the text editing dialog.

- Single-line text, created using the Text command in the ribbon:
  This form of text has the most options for formatting, and is described in this topic.

- Multi-line text, created from the Widgets palette:
  Selection of font, color and border, as described in the Multi-Line Widget topic.

- Labels built into VTScada widgets:
  These are controlled by the built-in Font tags. Use care if modifying those tags as there can be an adverse effect on the VTScada widget labels.

Single Line Text

The Format Ribbon applies to single line (standard) text only. That is, words created using the Text command of the Home ribbon. Use the following ribbon tools to adjust the appearance of selected text.

Color (Highlight and Text)	
The two options control the highlight (background) and the text color. By default, the background is fully transparent. Color options are not stored with named styles.	
Font	
Choose any system font installed on your computer, and set display characteristics including size (measured in points), bold, italic, or underline. Selected text responds to the keyboard shortcuts Ctrl-I (italic), Ctrl-B (bold), and Ctrl-U (underline). Choices made here can be stored in a named style for use with other text.	
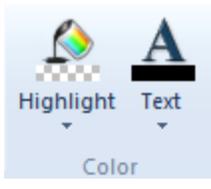
Alignment	Paragraph
Configuration of both horizontal and vertical alignment, relative to the bounding box containing the text. If the text does not fit within the bounding box provided, the end of the line will be clipped.	
Saved Fonts	
Fonts are saved in Font Tags. Alignment and color configuration are not included in the properties that can be saved. If you make changes in the Fonts section of the ribbon, you can choose to apply those changes to one of the existing fonts, or to create a new Font tag.	
Note: Built-in labels on VTScada widgets make use of the standard Font tags. Changes to the format of those may cause various labels to become difficult to read.	
Opening the Manage Fonts dialog results in the Tag Browser opening, filtered to the Font type.	

The ribbon options are also available in the form of a properties dialog box.
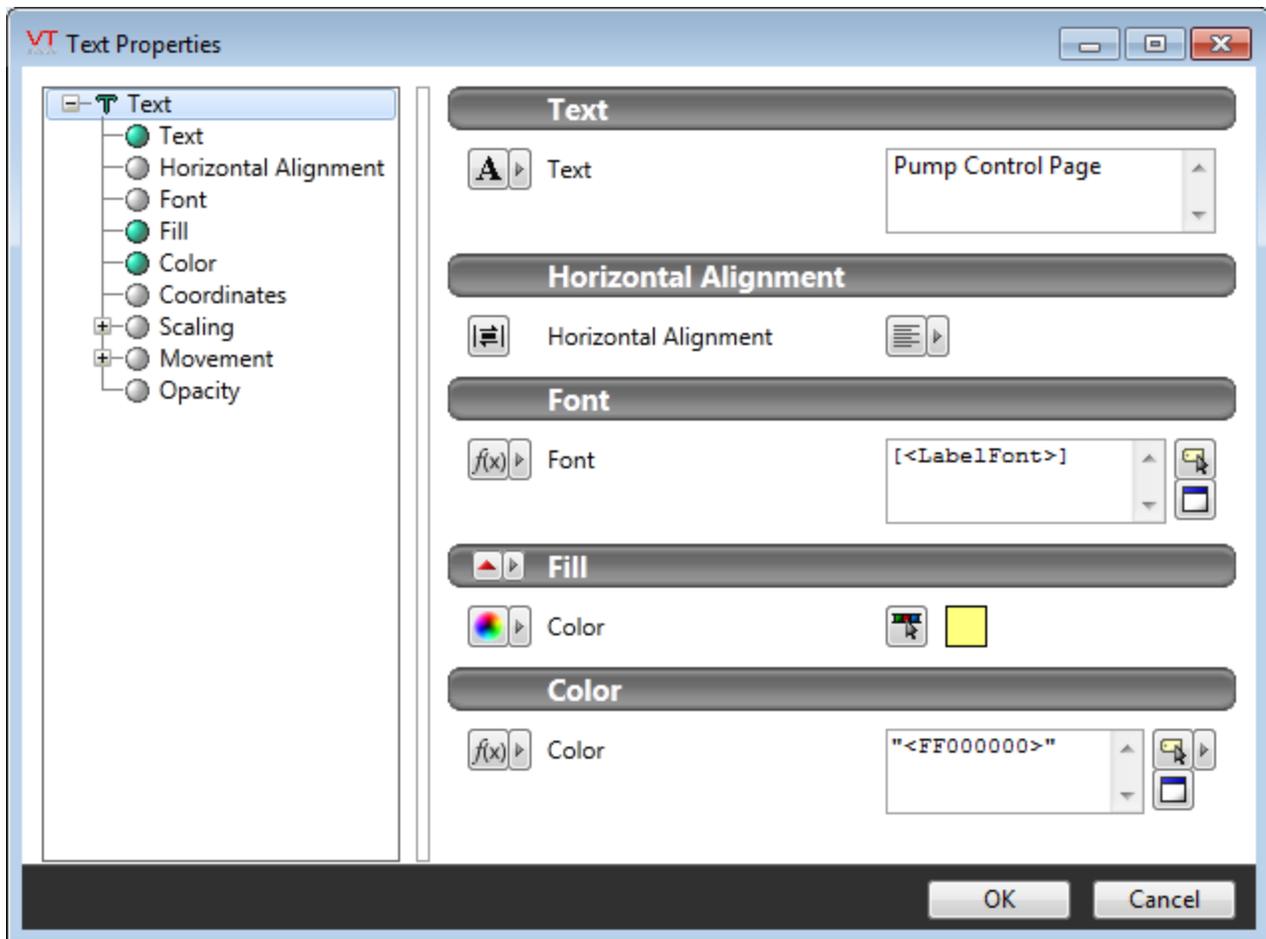
For ease of use, the ribbon is recommended. Advanced users, who are familiar with the VTScada scripting language, can use the tools in the properties dialog to tie display characteristics to expressions or tag values.

Related Information:

...Font Tags – Where VTScada font definitions are stored.

Alarm Configuration

This chapter describes configuration of the VTScada alarm system. Refer to Alarm System Operation in the VTScada Operator's Guide for information about viewing and responding to alarms. Refer to Alarms Tab of the Edit Properties Page in the VTScada Manager's Guide for configuration settings related to alarms.

The VTScada alarm system does much more than notify operators when a pump breaks, or a well is about to run dry. It also provides:

- Event logging to provide a record of operator control actions, security events, system events and more.
- Tools for building customized displays.
- A way to send notifications by phone, pager, email or text message when alarms go unacknowledged.
- The ability to speak alarms on the local server, in addition to the standard warning noise.
- Tools to enable or disable alarms. (For example, switching off alarms before doing system maintenance.)
- The ability to temporarily shelve an alarm, disabling all notifications while leaving the alarm enabled.

Alarm Databases

Alarms belong to Alarm Database tags. These tags specify which historian will be used to information from each alarm. By default, all alarms that you create will belong to the System Alarm DB tag.

For the majority of applications, there is no need to create extra alarm databases, but a few locations might choose to do so. For those that do, it is possible to create display groups so that alarms from several, but not all, databases can be viewed at once. Database groups are defined in your application properties, system section, by adding a property as follows:

```
Property name: AlarmDatabaseGroups
Section: System
```

```
Value: Northeast Group:NorEast Database,NorWest Database;Southwest
Group:SouWest Database, SouEast Database
```

For the preceding, note that each group name is followed by a full colon, then the names of the alarm databases that are to be included in that group, separated by commas. A semi-colon is used between each group name definition. All groups must be defined in the single property, AlarmDatabaseGroups.

General Alarm Configuration

Alarms are configured within tags. Either:

- Within an I/O tag such as the Analog Status, which has built-in alarm features.
- Within a dedicated Alarm tag, which is triggered by some other tag's value or by an expression.

Configure an Alarm Built Into a Tag

This procedure applies to Analog Status, Digital Status, and Pump Status tags, all of which have built-in alarm features.

Built-in alarms monitor only the value of the tag they are a part of. To create an expression that will monitor some function of one or more tags, create an Alarm tag.

VT New Digital Status Properties   ✕

| ID | I/O | **Alarm Setup** | External Alarms | Quality | Display | Historian |

Alarm Priority

[2] High ▾

Alarm State

0 - ▾

On Delay (s)

0

Alarm Rearm Time (s)

3600

Sound

Alarm Disable

☑ Disable

⦿ Constant ◯ Expression ◯ Tag

☐ Trip Alarm

☐ Popup Enable

☐ Alarm Rearm Enable

[OK] [Cancel]
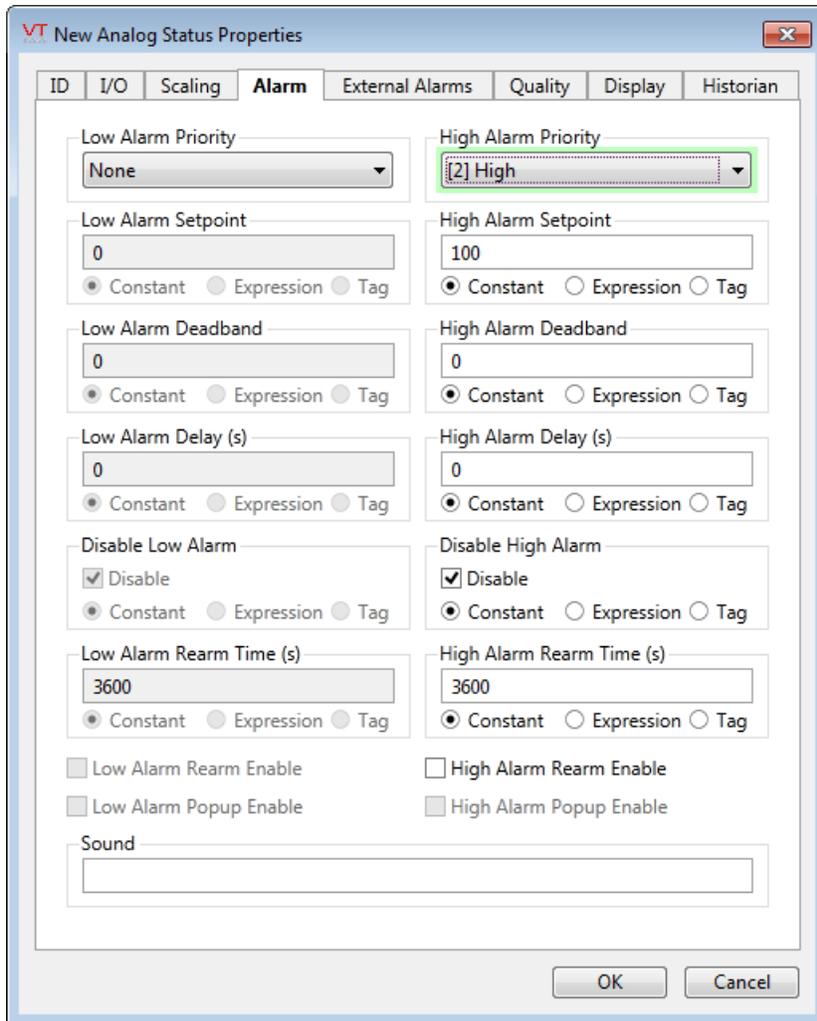
1. Open the "Alarm" or "Alarm Setup" tab of the tag.

2. In the case of an Analog Status tag, choose which alarm to configure first; high or low.

3. Select a priority for the alarm.
    Initially, all built-in alarms are both disabled and set to "None" as a priority. The None option exists so that the tag will not be included in a list of disabled tags. The Disabled option exists so that operators can enable or disable alarms if necessary for maintenance tasks or other reasons.

4. Configure a set-point.
    For a Digital Status or Pump Status this is a selection of state 0 or state 1. For the Analog Status, this is any number or any tag or expression that will resolve to a number.

5. Set the "Disable" option to "0" (zero) to enable the alarm.

Before the system becomes operational, it may be desirable to leave alarms disabled, thus preventing nuisance alarms.

6. Configure other options as necessary for your system.
   Refer to the description of the Alarm Tag for a complete reference of alarm options.

Configure an Alarm Tag

An Alarm tag can monitor the value of any other tag, or by using an expression, some function of one or more other tag values. Use an Alarm tag when you want greater control over what value is to be monitored, or when you want to monitor the value of a tag that does not have a built-in alarm. (This list includes analog and digital inputs, all output tags, calculation and function tags, and more.) Also, use Alarm tags when you want to add extra alarms to a tag that has built-in alarms.
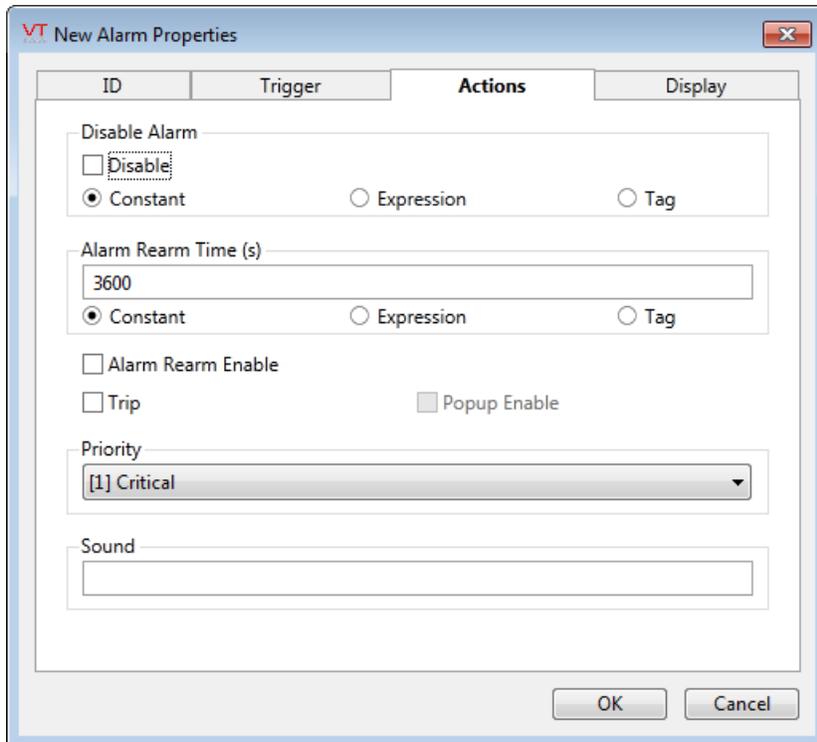
1. Add the new Alarm tag, using either of two ways:

- From within a tag such as the Analog Input, open the Alarm tab and click Add. This will automatically place the new tag as a child of the input.

- Using the tag browser, add a new Alarm tag.

2. If the Alarm tag was created from the tag browser, select the source tag, whose value is to be monitored.

3. Select the function to be used for the alarm condition.

4. Set the trigger value.
    This can be a constant, an expression or some other tag's value.

5. Select the function that compares the trigger to the source tag.
    The order of the comparison is always, source tag's value to set point value.

6. Open the Actions tab and select the priority of the alarm.

7. Ensure that the Disabled option is not set to 1, unless you are still com-missioning the system and wish to avoid nuisance alarms.

8. Configure other options as necessary for your system.
    Refer to the description of the Alarm Tag for a complete reference of alarm options.

An Alarm List is built into the Alarm Page. You may draw and configure Alarm Lists on any page you choose, creating customized alarm lists that relate to a specific set or group of tags. Refer to the notes for building a custom alarm display.

> Note: A maximum of 1000 records will be displayed in an Alarm List unless otherwise configured using the property, AlarmPageHistoryRecordLimit.
> If there are more than this number of records within the current time frame, a Load More button will be available. You can load successive groups of alarm records up to the maximum of 100,000 (or current value of AlarmPageHistoryHardLimit).

Troubleshooting:

- Values met the trigger condition, but no alarm happened.
  First, check that the alarm is not disabled or shelved. (The alarm page provides an easy way to list alarms in either category.)
  Next, check that the alarm priority is not 0 – Event. Event alarms are useful, but display no warnings.
  Next, check that the triggering condition is properly set, with the correct comparison.

Related Tasks:

...Alarm System Operation – See the VTScada Operator's Guide.

...Customized Alarm Display – Build a custom alarm display.

Related Information:

...Alarm System Tags – Reference for Alarm, Alarm Database, Alarm Priority, and Alarm Status tags.

...Spoken Alarms on the Server or Workstation – Hear alarm details instead of a warning noise.

...Alarm Message Templates – Control the information sent by email, text, phone, or in spoken alarms.

...Alarm Notification System - Alerts by Phone, Email or SMS-Text - Send notifications off-site.

...Alarm Notification System Tags - Reference for Modem, Roster, and SMS Appliance tags.

...Alarm Notification System Properties in the VTScada Admin Guide

...Alarm Data Logging - Reference: What is included in the alarm history log.
Refer to the VTScada Admin Guide for...

...Alarm Operation Privileges: - Security privileges relating to alarms.

...Alarms Tab of the Edit Properties Page - Commonly used configuration properties.

...Alarm Properties - Complete list of configuration properties.

...AlarmPageHistoryRecordHardLimit - Absolute maximum number of alarms that may be viewed in an Alarm List, using the Load More button.

...AlarmPageHistoryRecordLimit - Maximum number of alarms that may be viewed at one time in an Alarm List.

Alarm Indicators

VTScada notifies operators of the occurrence and priority of alarms using the following symbols and sounds.

Automatic Page Navigation
Disabled by default. You can choose to have VTScada automatically open a page where the tag related to an alarm is displayed if there is no operator-activity for a configured number of seconds after the alarm occurs. Four properties in the alarms tab of the Edit Properties page of the

Application Configuration dialog control whether and how automatic navigation will occur.

Display Manager Title Bar

When an alarm with any priority level above Event has been triggered, the alarm symbol in the Display Manager's title bar will flash red to notify operators. A slash through the alarm symbol indicates that sounds are muted.

Alarm Page

When an alarm has been triggered, the Current list and the Unacknowledged list on the alarm page will display details about the alarm. The Active list will also show the same alarms, except for those configured as "Trip" alarms.
An alarm that has not yet been acknowledged is always highlighted in the color associated with its priority in the alarm list. An alarm that has been acknowledged is not highlighted, but its text appears in the color associated with its priority in the alarm list as long as it is also active.

Alarm Sounds

When an alarm has been triggered, an alarm siren will sound to alert operators. When the alarm has been acknowledged, the alarm sound will cease.

Spoken Alarms

You can configure VTScada to speak an alarm using the Microsoft SAPI speech engine. This can be done either in addition to, or instead of, the siren.

Station Symbol

When an alarm associated with a specific station has been triggered, the center of the station symbol on the Overview page will flash in red for all alarms with a priority higher than Event. Two examples of station symbols indicating alarm conditions follow:
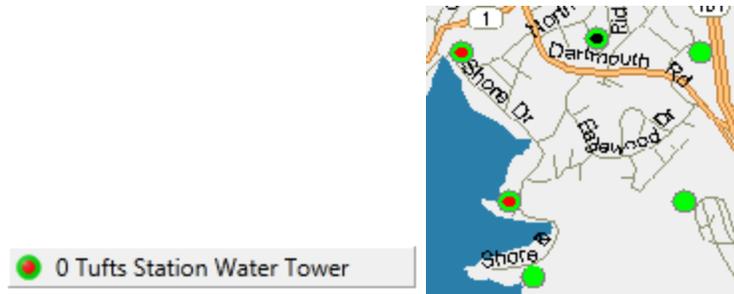
When the alarm is acknowledged, the center of the station symbol will stop flashing but will continue to display red to indicate that the alarm is still active.

Station Page

When an alarm associated with the I/O at a specific station has been triggered, the widget for the I/O tag will flash in red for all alarm priorities above Event. Additionally, the alarm will be highlighted in the station page's alarm list in the color associated with its priority (see Alarm Priorities for a list of the default colors associated with alarm priorities).

Widget Indicators

Many widgets will indicate when the linked tag is in an alarm state. These can be linked to a Style Settings tag and will blink red (or other color, as configured in the Style Settings tag) when the linked tag has triggered an alarm.

Two widgets exist expressly for indicating alarms: the Alarm Priority Icon and the Alarm Priority Box.

The "Draw" widget cannot be linked to a Style Settings tag, but will turn red to signal an active alarm. This widget is also able to show that one or more of the built-in alarms of the linked tag is disabled.

Box indicates a disabled alarm.

Pump running status
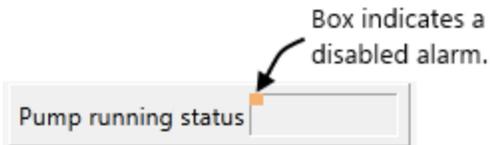
Related Information:

...Alarm Priority Tags – Controls the notification color and sound associated with each priority level.

...Alarm Status Tags – Can be drawn to indicate the presence or absence of any alarm matching a configured filter.

...Alarm Priority Icon – A widget that displays the status of the linked alarm (or alarms in children of the linked tag)

...Alarm Priority Box – An Alarm Priority Icon plus a box that can be placed around the equipment image.

...Style Settings Tags – Controls the color and behavior of alarm indications built into certain widgets.

Orphaned Alarms

It is rare but possible for an alarm to become "orphaned" during application development. For example, the following (completely contrived*) set of steps might lead to an orphaned alarm:

   A. Create an alarm and allow it to be activated.
   B. Export the tag database to Excel.
   C. Remove the activated alarm tag.
   D. Set the property, AlarmDatabasePurgeDelay to a high value to delay VTScada's automatic clean-up process.
   E. Restart the application.
   F. Import the spreadsheet containing the modified tag list.
   G. From the Alarm Page, open the details dialog for the alarm.
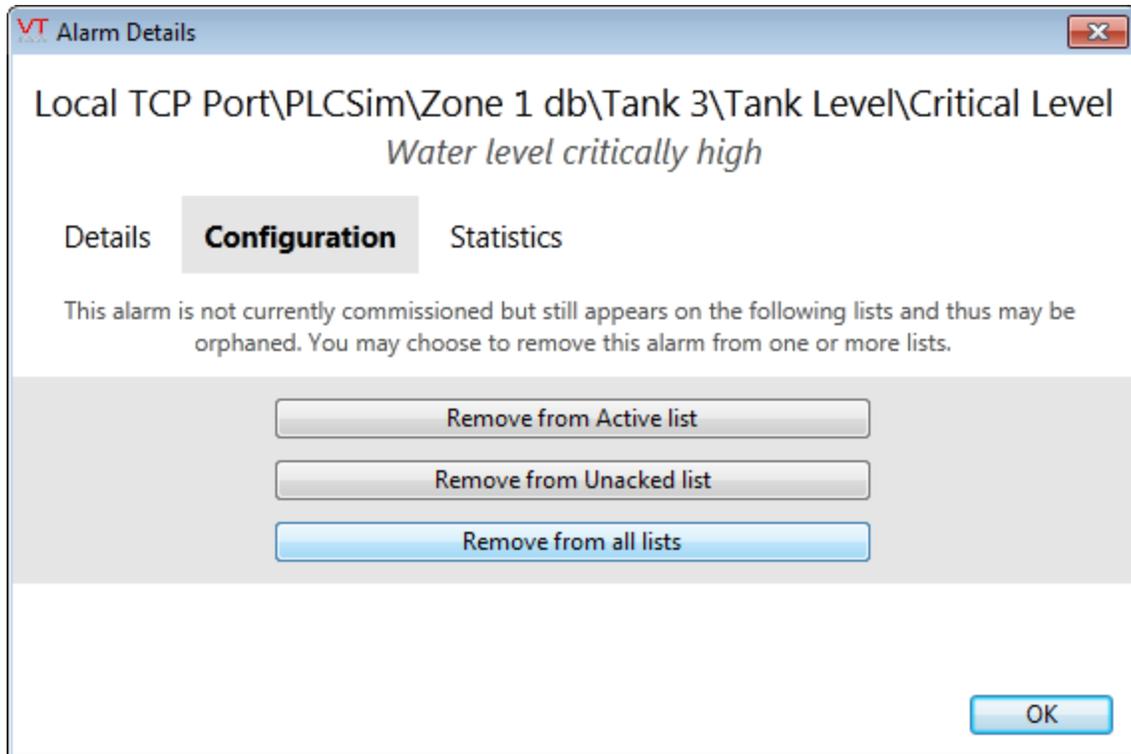
You can use this dialog to remove the alarm from the lists. In most cases, it is advisable to remove orphaned alarms from all lists. You will be prompted to confirm the action. (No record can ever be removed from the history list, but if a tag has been deleted, the record will show only its unique ID. This is done to avoid confusion with any new tag that might be created using the same name as the deleted tag.)

(*)Note that these steps are not recommended practice, and there is no guarantee that they will succeed in creating this situation, which is undesirable in any event. The steps are provided only to give a sense of how an orphaned alarm might come to be.

Related Information:

AlarmDatabasePurgeDelay in the VTScada Manager's Guide

Spoken Alarms on the Server or Workstation

You can configure VTScada to speak an alarm at the workstation or server. Alarms can be spoken either in addition to the standard tone, or

instead of the tone. This feature is available on VTScada Internet Clients as well as workstations running VTScada.

You can control what details are spoken by changing the template property. You also have control over how words are pronounced.

If an alarm is acknowledged while it is being spoken, the voice will immediately stop.

Requirements:

In order to hear a spoken alarm, all of the following must be in place:

- Microsoft SAPI engine installed on the computer. (Installed by default with Windows®)

- The workstation has speakers and audio ability.
  Refer to your system configuration. Ensure that the speakers are turned on and that the volume is set to an audible level, both on the speakers and in the computer.

- Application Properties related to the alarm text-to-speech system are set.
  At a minimum, you will require the application property, AlarmSpeechEnable to be set to 1.

- The Alarm Priority tags is configured to enable alarm speech.
  As circled in the following image, the alarm speech enable option is checked and the Alarm Annunciation Cycles set to a value greater than 0 in the Alarm Priority tag used by your alarm.

- The alarm is not configured to use a sound file.
  The Alarm Priority tag may use a .wav file. If the number of tone cycles is set greater than 0, the .wav file will place before the spoken alarm. If the Alarm tag has a .wav file configured, only that sound will be played.
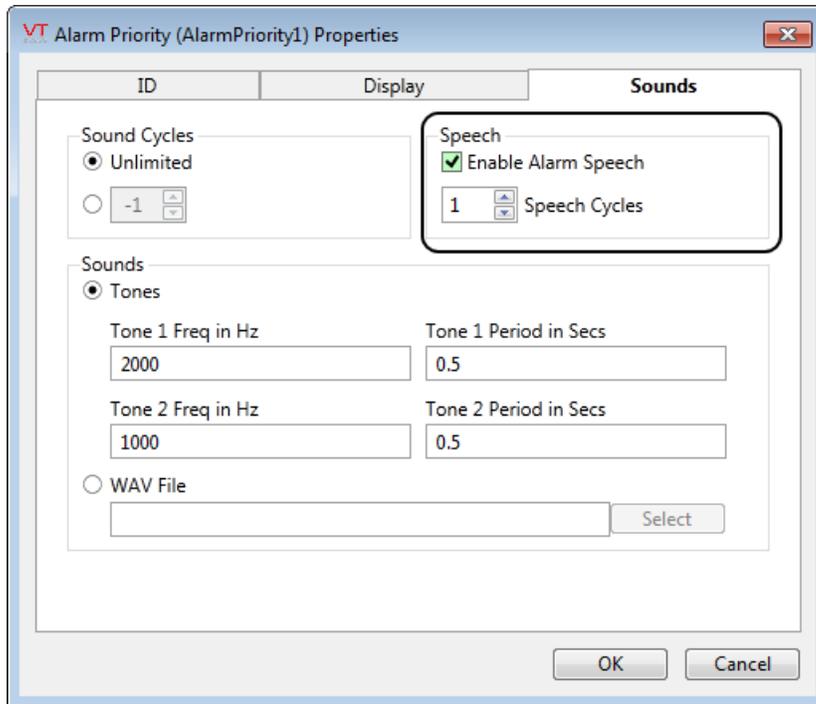
If the Tone Cycles is set to either 0, or the default of -1, then only the spoken alarm will sound. If set to 1, the tone will play once before the alarm is spoken.

Configuration Options

The following options are available to you to control how, when and where alarms are spoken.

- An alarm is spoken or not based on the configuration of its Alarm Priority tag.
  The Enable Alarm Speech is checked and Alarm Annunciation Cycles set greater than zero.
- Alarm tones or .WAV files may be played zero or more times before an alarm is spoken.
  .WAV files may be configured in the Alarm Priority tag, but not in the Alarm. Alarms with a configured .WAV file will not be spoken.
- Spoken alarms may be configured to play one or more times.
- Spoken alarms may be deactivated globally with the configuration setting, AlarmSpeechEnable.
- If using the SAPI speech engine, you can select which voice (supported by the speech engine) to use.

If your operating system is Microsoft Vista, it is recommended that you do not use the Microsoft Mary voice.

- You can select whether alarms should be spoken on all workstations, or on only the alarm server.

Configuration Settings

The following is an overview of the configuration settings that control the spoken alarms.

- AlarmSpeechEnable
  Set to 1 for alarms to be spoken. Alarm speech is disabled by default. Use workstation-specific copies of this property to enable speech on some stations and disable it on others.

- AlarmSpeechTemplate
  Provides a template for spoken alarms to use. May be a combination of words and replaceable parameters.

- AlarmSpeechVoice
  If using the Microsoft speech engine, there are several voices that you could use. You can select which voice to use with this variable. In general, it is best to set this variable to a dash (-) to indicate "configured Windows voice".

- AlarmSpeechQuality
  Determines the format quality used for speech using the Microsoft speech engine. The number used for this variable will match an entry in a numbered list of formats. When set to -1, the quality native to the voice engine will be used. The default value of 6 balances audio quality against network transfer overhead.

- AlarmSpeechInit
  May be used to provide an initialization string to the speech engine to control the voice. Please refer to the SAPI documentation for the details of this initialization string.

- SpeechEngine
  May be used to select an alternate speech engine.

- UseOldSpeechEngine
  Obsolete. Maintained only for backward compatibility. If enabled, spoken alarms will not have access to the pronunciation configuration options.

Related Information:

...Alarm Priority Tags – Enables speech for a given priority and controls annunciation cycles.

... Speech Lexicon Configuration – Control how words sound.

...Alarm Notification System – Alerts by Phone, Email or SMS-Text – Remote notification of alarms.

...Refer to the VTScada Admin Guide for:

...Alarm Properties – Alarm-related application properties.

...Alarm Notification System Properties – Related application properties.

Alarm Message Templates

A template can be defined using application properties to control the content of alarm messages. Separate templates can be defined for each of:

Type of alarm	Setting name
Spoken alarms	AlarmSpeechTemplate
Telephone alarms	AlarmDialerTemplate
Emailed alarms	AlarmEmailTemplate
Paged alarm messages	AlarmPagerTemplate

Note: In addition to these, you may set the subject line for alarm emails using the property, AlarmEmailSubjectTemplate.

Format control strings take the form: TemplateName = control string, The control string can be a combination of words and the following replaceable parameters. Note that the pause can be valuable in adding clarity to the message by separating words that might otherwise run together when spoken.

Parameter	Meaning
%A	Area of the Alarm tag.
%D	Date of the alarm
%F	Full tag name

%H	Short tag name
%M	Alarm description
%N	New sentence for email and pager messages.
%O	Name of the operator logged on at the time the alarm was triggered.
%P	Priority of the alarm.
%S	Status of the alarm
%T	Time of the alarm
%U	Units of the Triggering tag.
%V	Alarm value (this is the value of the alarm trigger at the time that it triggered the alarm)
%W	Pause for ¼ second. Has no effect on email or pager messages.

Examples:

```
AlarmSpeechTemplate = %P alarm. %T, %M
```

Speaks (for example): "Critical alarm. 10:35 pm, Well level high"

```
AlarmSpeechTemplate = VTScada %P alarm in %A %W %M has reached %V at
%T.
```

"VTScada critical alarm in zone one <pause> Holding tank level has reached eighty at eleven forty nine a. m.".

Related Information:

Refer to the VTScada Admin Guide for:

...AlarmTemplateDateFmt – Controls the date format spoken.

...AlarmTemplateTimeFmt – Controls the time format spoken.

...AlarmEmailSubjectTemplate – Sets the subject line for alarm emails.

Customized Alarm Display

You can add an alarm list, configured to show some or all of its tools, to any page in your application. The list can be customized in advance so that operators will see only the list relevant to that page.

All the tools required to work with an alarm list are built into the list widget, but you might decide to suppress everything other than the alarm display and provide additional buttons that perform only specific tasks. For example, rather than providing the full toolbar, you can suppress it and provide an Acknowledge Shown button and a Mute button.

The Alarm Tools folder contains a Legacy Tools sub-folder. These provide backward-compatibility with legacy applications (those built using VTScada 11.1 or earlier), which used an older form of the Alarm List.

Note: Most of these tools are designed to interact with a named alarm list that you have drawn. Therefore, you are advised to start your custom alarm display with an Alarm List, configured to have a unique name.

Related Information:

...Alarm Priority Box

...Alarm Priority Icon

...Legacy Alarm List - Displays alarms in a way that can be filtered and sorted. Required before creating most of the following buttons.

...Ack All Button - Acknowledge all alarms.

...Ack Selected Button - Acknowledge a single, selected alarm from a named Alarm List.

...Ack Shown Button - Acknowledge all alarms shown in a named alarm list.

...Legacy Alarm Banner - Creates a display of alarms that cannot be filtered or sorted.

...(Legacy) Alarm Date Selector - Choose which date to view. If alarm history is stored in multiple files, the Next and Previous Log File buttons may be used.

...(Legacy) Alarm Go To Most Recent Button - Provides a quick way to reset the list.

...Alarm List Button – Change the Alarm List to display a different category of alarms.

...(Legacy) Filter Dialog Button – Opens a dialog to apply one or more filters to the selected Alarm List.

...(Legacy) Filter Field Button – Filter the current alarm list by a single, pre-selected option.

...(Legacy) Log File Date – Displays the date of the log file being viewed when the list shows alarm history.

...Mute Button – Silence all alarms, both old and new.

...(Legacy) Next Log File Button – Step to the next file of logged alarms when viewing alarm history.

...(Legacy) Prev Log File Button – Step to the previous file of logged alarms when viewing alarm history.

...Silence Button – Silence the most recent unacknowledged alarms.

...(Legacy) Sort Button – Sort the alarm list by any numeric property.

Alarm List

Used to draw a customizable alarm list on any application page. This is the same list that appears in the Alarm Page, including all tools and filters. The difference is that, when drawn on another page, you have control over which parts should be included and which display list should be shown.

Note: A new Alarm List widget was introduced with version 11.2. Among other changes, it no longer has a name. Some of the alarm tool buttons will still work with the new list (although, none are necessary) but, these will affect the first list they find on a page. If you intend to draw two alarm lists on one page, and you wish to use the independent alarm tool widgets to work with those lists, group each list and the tools related to it.

Better to use only the new Alarm List widgets.

Your configuration options. All of the display options that are available in the list's toolbar (and more) can be configured here. If you choose to hide the toolbar, then operators will not be able to override the values that you set.
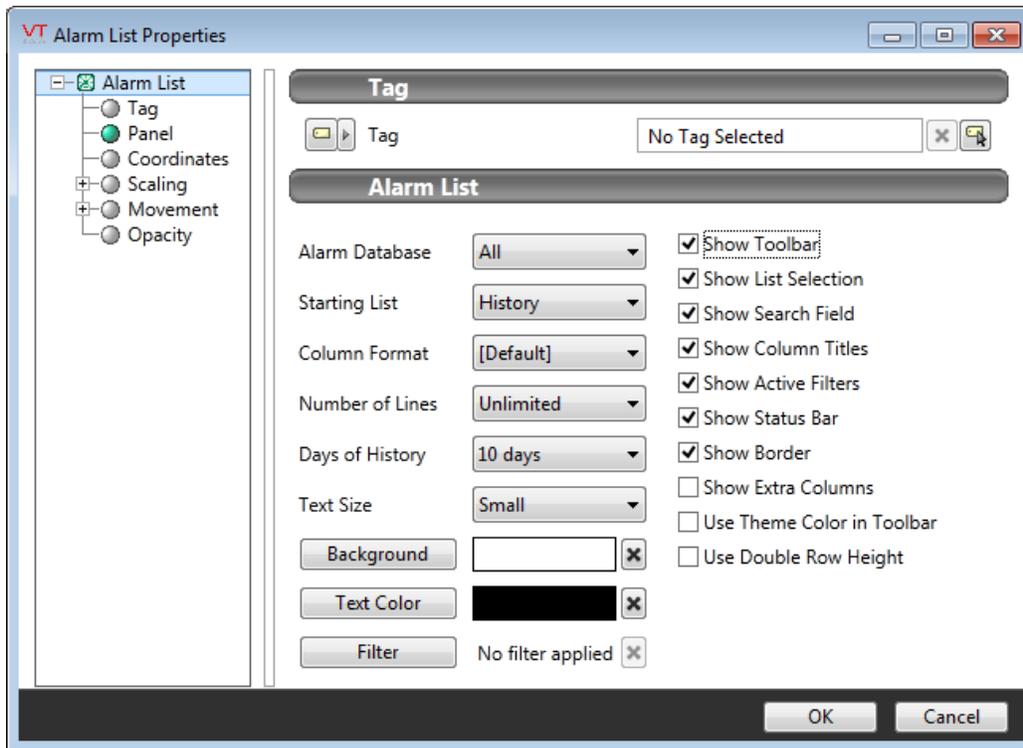
Tag

Select a tag if (and only if) you want this list to show only alarms due to that tag and its children. This is often used to create an alarm display that is specific for one site, where the tags at that site are organized into a hierarchy under one parent tag.

Alarm Database

Use this option to pre-filter the display to show alarms in all databases (the default) or from only one alarm database.

Starting List

The lists are History, Active, Unacked, Current, Shelved, Disabled and Configured. The History list is the first and therefore the default. Current, Active or Unacked (unacknowledged) are the most common lists to use.

A related check box in this dialog can be used to either allow or deny operator selection of other lists.

Column Format

Select a column format from those defined in the Alarm Database.

Available Line Formats	
Default	Varies with selected database and list. All columns relevant to the selected list are displayed.
Alarms	All alarm-related columns relevant to the selected list are displayed.
Events	All event-related columns relevant to the selected list are displayed.
Legacy 1	Priority (color bar) | Event | Priority (icon) | Time | Area | Name | Description | User
Legacy 2	Priority (color bar) | Time | Name | Description
Legacy 3	Priority (color bar) | Time | Ack | State | Status | Priority (icon) | Area | Name | Description | User
Legacy 4	Priority (color bar) | Time | State | Event | Priority (icon) | Area | Name | Description | User
Legacy 5	Priority (color bar) | Ack | State | Event | Priority (icon) | Area | Name | Description | User
Legacy 6	Priority (color bar) | Time | Event | Priority (icon) | Area | Name | Description | User
Legacy 7	Priority (icon) | Time | Name | Description | Ack
Popups	Priority (icon) | Ack | | Name | Description

Number of Lines

Limit the number of alarms that will be shown in the list. If selected number of lines exceeds the display area, a vertical scroll bar will be added.

Days of History

Select one of the three presets: 10 days, 30 days or 90 days, to set the right balance between the time required to populate the list and the amount of information required by the operator.

Text Size

Choose between small, medium and large for the text font size.

Filter

This button opens the Advanced Filtering dialog. Filters will be applied to and saved with the list. The filter bar will always be shown with the list so that operators will know that the list they see is filtered. Operators can clear the filter temporarily, but it will be reapplied whenever the page is refreshed. They can create (temporary) new filters only if you have selected the option, Show Toolbar.
In the properties dialog, a message will be added beside the button to remind you that a filter was created:

Show Toolbar

The toolbar is the set of buttons across the top of the alarm list. Hide this if you do not want to allow operators to use any

of these functions when viewing the list.

Show List Selection

The List Selection bar is used to change what is shown in the alarm list. Hide this if you do not want to allow operators to switch to any list other than the one you configured for this display.
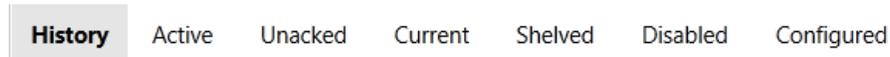
Show Search Fields

The search field is located to the right of the list selection bar. It enables operators to search for alarms that they are allowed to see using any scrap of information known about that alarm. Hide this if you do not want to allow operators to search for alarms other than the ones that your list has been configured to show.

Show Column Titles

Column titles are useful for new operators, but those who are familiar with the alarm list will not need the titles to know what each column of information in the list means. You might choose to hide the titles on pages where the extra space is needed.

Show Active Filters

Active filters are those that operators set using the tools in the

toolbar. They are shown on a bar that is added below the list selection whenever operator-defined filters are active. If the toolbar is hidden, there will be no active filters and therefore no need to either show or hide this option.

When the bar is visible, less space is available for the list of alarms. Choosing to hide active filters, but still allowing operators to set those filters can cause confusion and errors.

Show Status Bar

The status bar is shown below the alarm list. It tells the viewer which list is being viewed (History, Current, ...) and how many records are shown in the list. When viewing the History list, the status bar will include the current History Range Limitation.

Show Border

Adds a one-pixel border around the alarm list.

Use Theme Color in Toolbar

By default, the title and alarm list selection bars have a white background. Check this option to re-color those bars using the application's current theme color.

Background & Text Color

The background color applies to the list, and also to the area of the toolbar and list selection bar if you have not chosen to use the theme color for those areas.

If you change the background color, you should also choose a text color that contrasts clearly against the background so that

operators do not have trouble reading the information in the list.

Related Information:

Refer to the VTScada Operator's Guide for:

...Sort and Filter the Alarm List – Describes both the Advanced Filtering dialog and the Filter by Selection tool.

...Alarm List View and Fields – Includes a description of each column in the list.

Ack All Button

Acknowledge all alarms.

This button is enabled only if there are alarms to acknowledge and you are authorized to acknowledge them

> Note: This button is not limited to any alarm list. "All alarms" means exactly that. If your intent is to acknowledge only those alarms in the filtered list, use the Ack Shown button instead.

The properties dialog for the Ack All button:
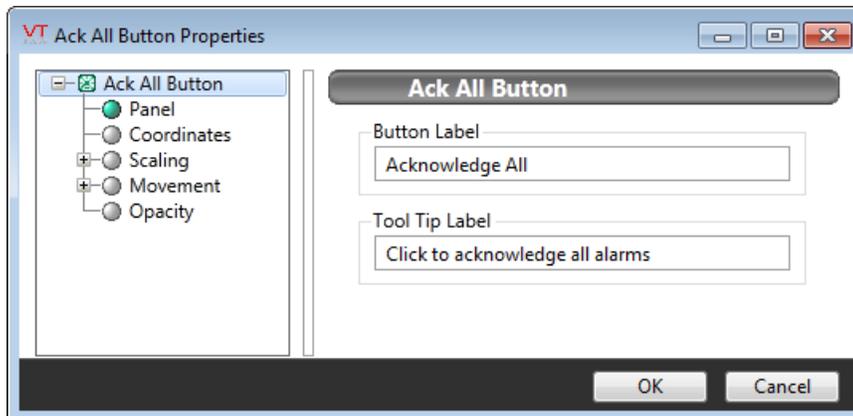
Button Label

Enter the text you wish to be displayed on this button. Labels

should be kept short and precise, as the length of the label ultimately determines the size of the button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tips Label

Enter the text that you wish to be displayed on this button's tool tip (which opens when the mouse pointer is rested on this button).

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a named Alarm List

...Ack Selected Button – Acknowledge a single, selected alarm from a named Alarm List.

...Ack Shown Button – Acknowledged all alarms shown in a named Alarm List.

Ack Selected Button

Click to acknowledge a single, selected alarm from a named Alarm List. This button will be enabled only when an alarm that needs to be acknowledge has been selected, and the logged-in operator is authorized to acknowledge them.

The properties dialog for a completely configured Ack Selected button is as shown:

Name

Used only when this button is in the same page or widget as a legacy alarm list, in which case it is required to associate this button with the legacy list.
If this widget is used with a modern Alarm List, it will be associated with that list automatically. If a page will contain two Alarm List widgets, each list and its related control buttons must be grouped into a widget.

Button Label

Enter the text you wish to be displayed on this button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.
The Tool Tips Label field (see next) is provided to enable you to enter a more lengthy description of the button's purpose.

Tool Tips Label

Enter the text that you wish to be displayed on this button's tool tip (which opens when the mouse pointer is rested on this

button).

...Legacy Alarm List – This button is relevant only in relation to a named Alarm List

...Ack All Button – Acknowledge all alarms.

...Ack Shown Button – Acknowledge all alarms shown in a named Alarm List.

Ack Shown Button

Creates a button that can acknowledge all alarms shown in the named alarm list. Alarms that are not shown because they do not match the list's current filter, will not be acknowledged.
Enabled only if the list has alarms that can be acknowledged, and the logged-in operator is authorized to acknowledge them.
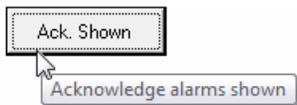
An example of a completely configured Ack Shown Editor dialog:
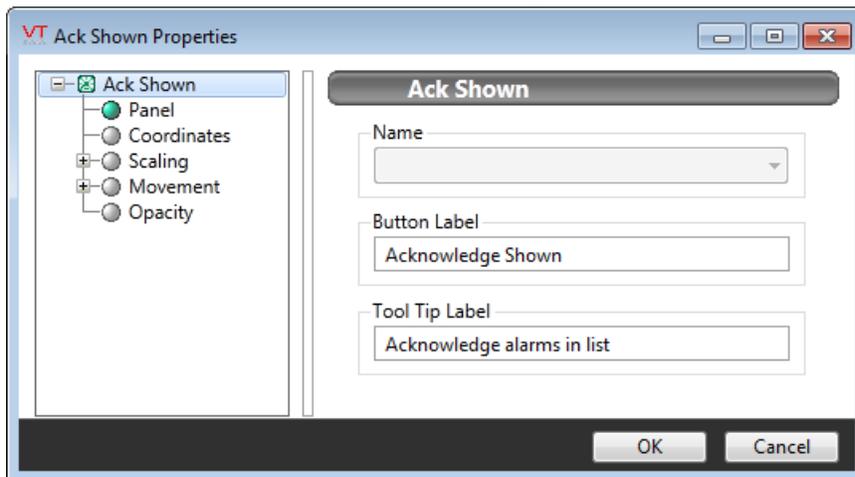
Name

Used only when this button is in the same page or widget as a legacy alarm list, in which case it is required to associate this

button with the legacy list.

If this widget is used with a modern Alarm List, it will be associated with that list automatically. If a page will contain two Alarm List widgets, each list and its related control buttons must be grouped into a widget.

Button Label

The text entered in this field will be displayed on button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.

If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tips Label

The Tool Tips Label field is used to enter the text that you wish to be displayed on this button's tool tip (which opens when the mouse pointer is rested on this button).

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a named Alarm List

...Ack All Button – Acknowledge all alarms.

...Ack Selected Button – Acknowledge the selected alarm in a named Alarm List.

Alarm List Button

Change the Alarm List to display a different category of alarms. There are 6 possible alarm list categories (Active, Current, Configured, Disabled, History, Unacknowledged). Draw one instance of the alarm list button for each category that you might want to see displayed in the list.
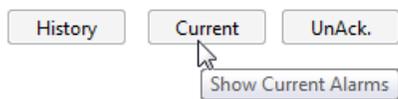
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

The properties dialog for the Alarm List Button:

Name

Used only when this button is in the same page or widget as a legacy alarm list, in which case it is required to associate this button with the legacy list.

If this widget is used with a modern Alarm List, it will be associated with that list automatically. If a page will contain two Alarm List widgets, each list and its related control buttons must be grouped into a widget.

Button Label

Enter the text that you wish to be displayed on this button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.

Tool Tip Label

Enter the text that you wish to be displayed on this button's

tool tip (which opens when the mouse pointer is rested on this button). Use this field to compensate for the brevity of the button label by providing a more complete description.

Starting List

Select the category of alarms that you wish to be displayed in the alarm list when this button is clicked. This can be one of:

- History
- Active
- Unacknowledged
- Current
- Disabled
- Configured

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a named Alarm List

Mute Button

Use to create a button that can be clicked to silence all alarm sounds (current and future). No alarm sounds will occur while this button is toggled on.
Enabled only when there are alarms to mute and the current operator is permitted to use this feature.

Note: All alarm lists are affected by this button.

The properties dialog for the Mute Button:
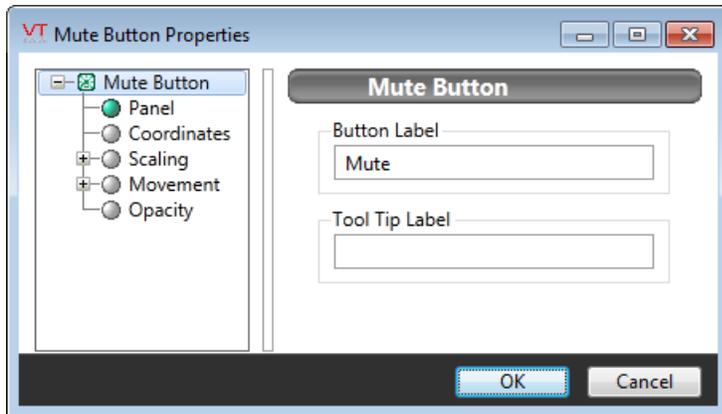
Button Label

The text to be displayed on the button.
If the label is to include an ampersand (&), then the & character
will need to be doubled in order to appear.

Tool Tip Label

A description of this tool, visible only when the operator hov-
ers the mouse pointer over the button.

Related Information:

...Silence Button – Silence the most recent unacknowledged alarms.

Silence Button

Silence the alarm sound associated with the most recent unac-
knowledged alarms in your system. New alarms will sound.
This button is enabled only when there are alarms to silence and the cur-
rent operator is permitted to use this feature.

Note: All alarm lists are affected by this button.

The properties dialog for the Silence Button:
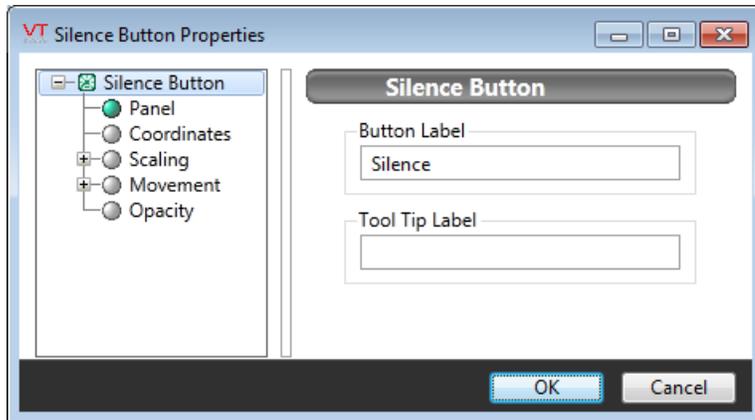
Button Label

The text to be displayed on the button.

If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tip Label

A description of this tool, visible only when the operator hovers the mouse pointer over the button.

Related Information:

...Mute Button – Silence all alarms, both old and new.

Legacy Alarm Banner

Note: Provided for backward compatibility, this will create the legacy Alarm Banner, which existed in VTScada prior to version 11.2 The current Alarm List widget should be used in all new applications.

Creates a display of alarms, similar to a legacy Alarm List, except that the banner cannot be filtered. If more than one is shown the most recent alarm will be at the bottom of the list.

An example of a completely configured Alarm Banner properties dialog:
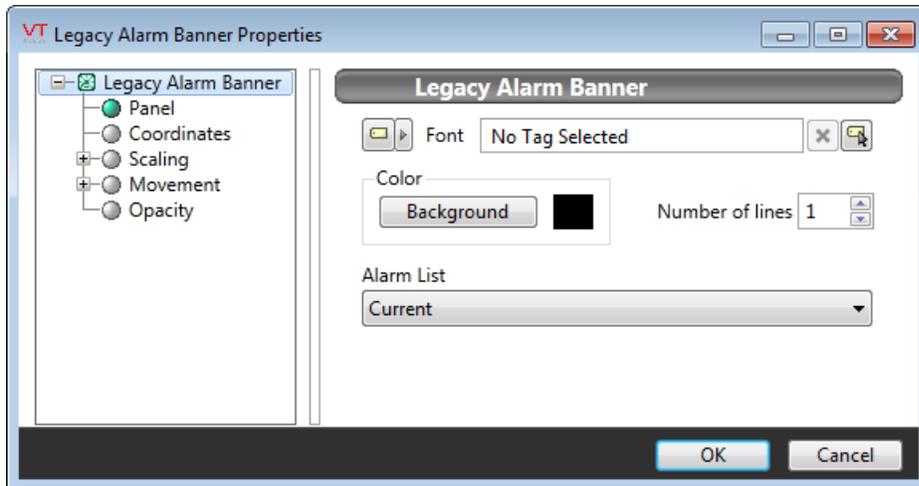
The result of the above configuration is a region on the screen showing the most recent alarm in the History list, on a black background.

Font

Select the display font for the banner, as configured in one of the available font tags.

Color

Sets the background color for the alarm banner. Since the text will be drawn in white, a dark color is recommended to maximize visibility.

Number of lines

Used to select how many alarms are to be shown in the banner. The most recent alarm will always be shown at the bottom of the list..

Alarm List

Select which alarm category you wish to display in the banner. There are 7 possible alarm list categories (Active, Current, Configured, Disabled, History, Unacknowledged, Shelved).

Related Information:

...Legacy Alarm List – Displays alarms in a way that can be filtered and sorted.

Legacy Alarm List

Note: Provided for backward compatibility, this will create the legacy Alarm List, which existed in VTScada prior to version 11.2 The current Alarm List widget should be used in all new applications.

Used to draw a custom alarm list on any application page.

The properties dialog for configuring a Legacy Alarm List .
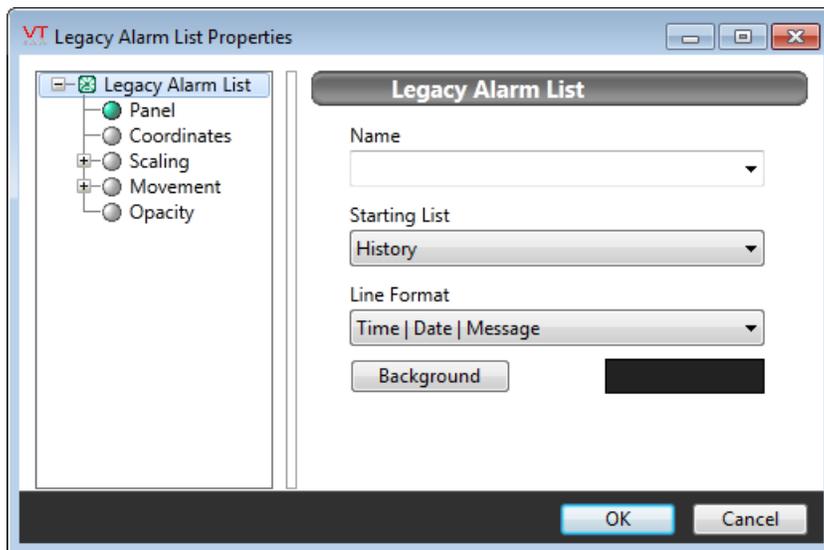
Name

> Enter a name for this custom alarm list.

Font

> Select a Font tag to use in this alarm list. The style and font within the Font tag will be used to display the alarms and events in this alarm list.

You may choose one of the four default Font tags that are included with every VTScada application, or you can create your own custom Font tag and assign it attributes as you require.

Line Format

Select how much information should be displayed in the list. Options include the following:

List Options
Status | Priority | Time | Date | Area | Message | Operator
Time | Date | Message
Time | Date | ACK | Active Icon | Status | Priority | Area | Message | Operator
Time | Date | Active Icon | Status | Priority | Area | Message | Operator
ACK | Active Icon | Status | Priority | Area | Message | Operator
Time | Date | Status | Priority | Area | Message | Operator

Note: The format of the text displayed in the Event/Status and Priority columns can be modified using application properties. (See: Application Properties for Alarms).

Background

Select the background color you wish to be displayed for this alarm list. When the Background button is clicked, the Select Color palette will open to enable you to select the color you desire. The selected color will be previewed to the right of the Background button. You can also choose to display ruled lines between each alarm.

Starting List

Select the name of the alarm list that you wish to be displayed

each time the page upon which this alarm list is draw is opened. You may choose from the following alarm lists:

- **Active:** Displays all alarms that are in an active state; that is, all alarms for which the condition that caused them to trigger still exists. Active alarms are also displayed in the "Current" alarm list (see above).

- **Current:** Includes alarms that are either active, unacknowledged, or both.

- **Configured:** Includes a reference to all Alarm tags that have been configured for this application.

- **Disabled:** Displays all alarms that have been flagged as disabled within your application.

- **History:** Displays all alarms and events that have occurred for this application, including operator control actions (e.g. output of a value to equipment), remote configuration actions (such as updating files or tags), Alarm Notification System actions (such as the notification of operators of unacknowledged alarms by voice, pager, or email), and security-related events (such as operators logging on or off).

- **Shelved:** Displays all alarms that have been flagged as shelved within your application.

- **Unacknowledged:** Displays all alarms that have not yet been acknowledged, regardless as to whether or not the alarms are still in an active state. Unacknowledged alarms are also displayed in the Current alarm list (see above).

Related Information:

...Legacy Alarm Banner – Creates an alarm list that cannot be filtered or sorted.

(Legacy) Alarm Date Selector

When viewing alarm history in a legacy Alarm List, you may want to view alarms from some day in the past. Use the Alarm Date Selector to create a button that will open a calendar selector. By selecting a date from the

calendar, operators can change the display to show the alarm history from that date. Double arrows at the upper corners of the calendar window allow operators to change the month displayed.

Selecting Sept. 4th, 2009.

The current date will be shown circled in red. The tool tip for the Alarm Date Selector button will read "Alarm History Selector".
The selection of alarm list is now obsolete.
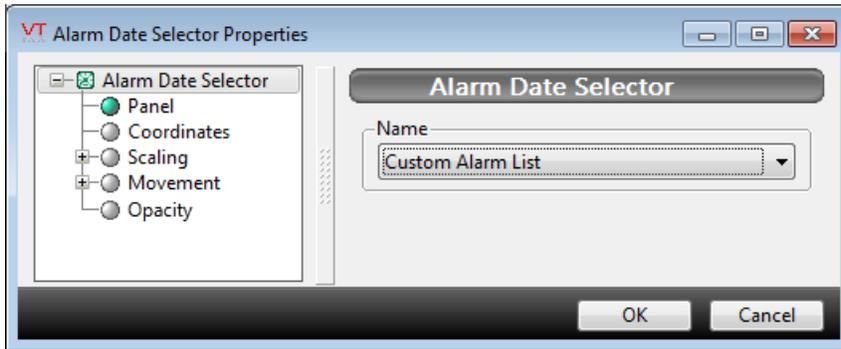
Related Information:

...Legacy Alarm List – This button is relevant only in relation to a named Alarm List

...(Legacy) Alarm Go To Most Recent Button – Provides a quick way to reset the list.

...(Legacy) Next Log File Button – Step to the next file of logged alarms.

...(Legacy) Prev Log File Button – Step to the previous file of logged alarms.

(Legacy) Alarm Go To Most Recent Button

This tool is a companion to the Alarm Date Selector. If an operator has adjusted the alarm list display to view the history for a date other than

today, the Alarm Go To Most Recent button provides a quick way to reset the list.
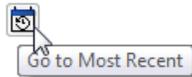
The properties dialog for this button provides only two configuration tools.
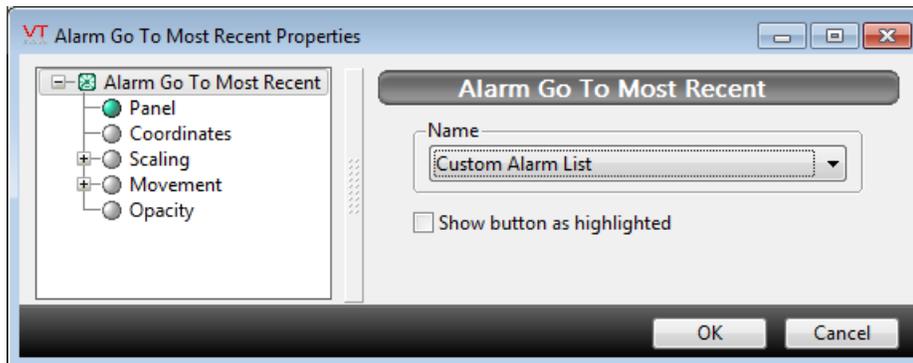
Name

Choose which legacy Alarm List will be affected by the button.

Show Button as Highlighted

You can also modify the behavior of the button, such that it will be highlighted whenever a date other than that of the most recent alarm has been displayed.

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a legacy Alarm List

...(Legacy) Alarm Date Selector – Choose which date to view. If alarm history is stored in multiple files, the Next and Previous Log File buttons may be used.

(Legacy) Filter Dialog Button

Opens an The Advanced Alarm Filtering dialog dialog. Operators can use that dialog to apply one or more filters to the selected Legacy Alarm List.
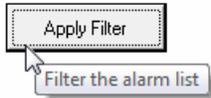
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

The properties dialog for this button will look as shown:
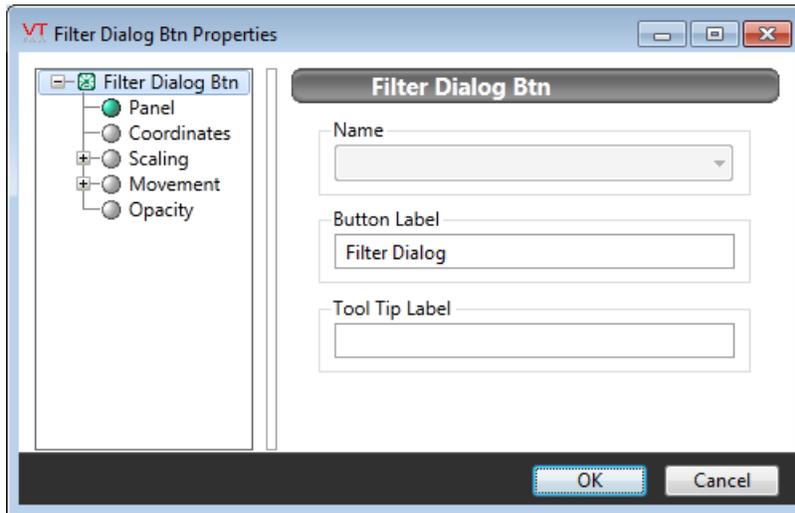
Name

Select which legacy Alarm List will be affected by this button.

Button Label

Enter the text you wish to be displayed on this button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.

Tool Tip Label

Enter the text that you wish to be displayed on this button's tool tip (which opens when the mouse pointer is rested on this button).

The Advanced Alarm Filtering dialog
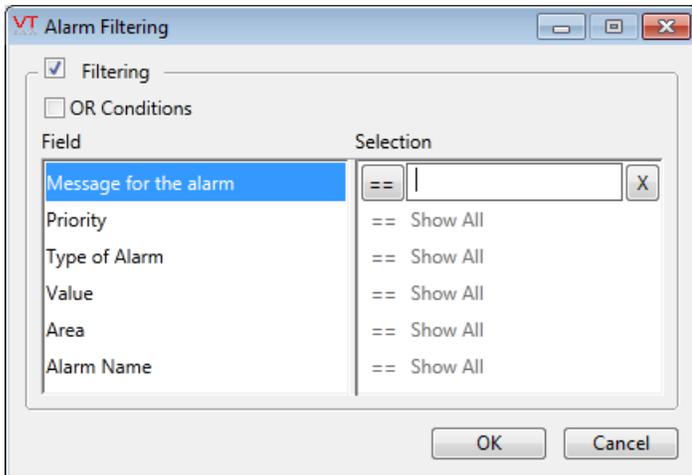
This dialog provides a variety of ways to filter a custom alarm list. Descriptions follow:

Filtering

Select the Filtering check box to enable the Alarm Filtering dialog.

OR Conditions

The OR Conditions check box can be clicked to indicate that you wish the configured alarm filtering parameters to be "OR'd" together. For example, if you set Priority to "1" and Area to "East" then selecting the "OR Conditions" check box would result in filtering based on "Priority is equal to '1', OR Area is equal to 'East'". If the OR Conditions check box is not selected, configured alarm filtering parameters will be "AND'ed" together. For example, deselecting the OR Conditions check box would result in filtering based on "Priority is equal to '1', AND Area is equal to 'East'".

Field

The Field column displays the available alarm list fields by which you may filter. You may choose one of:

Message for the Alarm: Used to filter the alarm list by a specific Alarm tag's Description property. Note: Wildcard characters are not permitted. Enter the full description as it appears in the Alarm tag's Description field. In the event that a Description property has not been configured for an Alarm tag, the message for the alarm will contain the name of the Alarm tag, as it appears in its Name field.

Priority: Used to filter the alarm list by a specific alarm priority. The available default alarm priority index numbers are as follows:

Alarm Priority	Index	Associated Alarm Priority Tag Name
No Alarm	0	AlarmPriority0
Critical Alarm	1	AlarmPriority1
High Alarm	2	AlarmPriority2
Warning Alarm	3	AlarmPriority3
Notice	4	AlarmPriority4

Type: Select if you wish to filter the alarm list based on a specific alarm type. Note: By default, all system alarms in VTScada (security, modem, operator logging, and report) have a type of "1000". VTScada programmers may specify their own type values for custom Alarm tags, and can therefore use the Type field to filter them. Custom Alarm tags and applying type filtering is beyond the scope of this guide.

Operator Name: Select if you wish to filter the alarm list based on a specific operator.

Area: Select if you wish this button to filter the alarm list based on a specific Area property. Note: The VTScada alarm system logs security-related events, operator control action-related events, remote configuration-related events, and Alarm Notification System-related events for reporting in the

History list on the Alarm page. These entries have the following Areas:

- Security-related events have a default area of "Security".
- Operator control action-related events and remote configuration-related events have a default area of "OperatorLog".
- Alarm Notification System-related events have a default area of "Alarm Dialer".

Note: You can change the field descriptions displayed in the Field column using application properties. See: Application Properties for Alarms.

Alarm Name:

The Alarm Name option can be selected to filter based on names.
This can be useful for viewing the history of a particular alarm, or for limiting the display to alarms matching a name pattern such as PUMP*.
Wildcards are available to expand the selection. The asterisk wildcard (*) matches any number of characters, beginning at the location of the wildcard. The question mark wildcard (?) matches any single character at exactly the location of the wildcard.

Selection:

The Selection column is used to supply a comparison operator and a value to compare the matching field entry to. The following operators are supplied:

Operator	Description
==	Equal To
>	Greater Than

<	Less Than
*?	Wildcard (i.e. you may enter characters combined with either the ? or the * wildcard characters)
!=	Not Equal To
<=	Less Than Or Equal To
>=	Greater Than Or Equal To
!*?	Not Wildcard (i.e. exclude results that match the entered characters combined with either the ? or the * wildcard characters)

Example:

Filter the Alarm List Using the Alarm Filtering Dialog

1. Click the Filtering check box to enable the Alarm Filtering dialog.
   If there is to be more than one condition in the filter, decide whether the alarms much match all the conditions (e.g. "Priority <=2 AND Area == West") or whether the alarms need match only one of the conditions (e.g. "Priority <= 2 OR Area == West").

   a. If only one of the conditions need be met, check the OR box.
   b. If all the conditions must be met simultaneously, do not check the OR box.

2. Click the field on which you wish to filter in the Field column.
   The corresponding Selection field will become enabled.
3. Click the operator button until the operator you wish to use appears.
4. Enter the value on which you wish to filter in the Selection field.
5. Repeat the previous two steps to configure the next filtering criteria (if necessary).
   An example of a completed Alarm Filtering dialog:
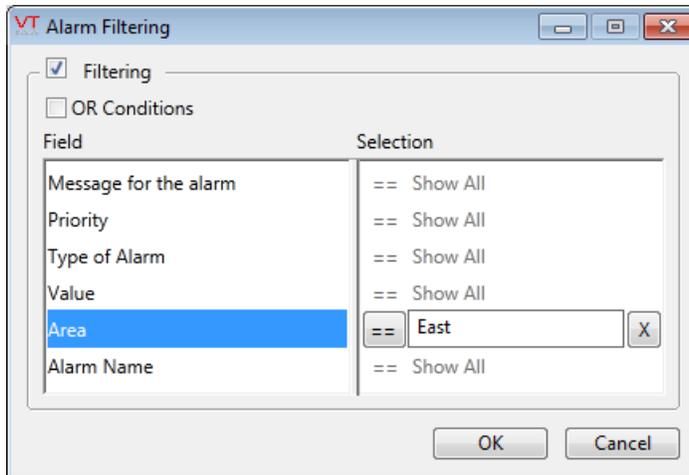
6. Click OK.

   The Alarm Filtering dialog will close, and you will be returned to your custom alarm page where the Filter Dialog Button object will display a checkmark, indicating that the alarm list has filtering applied to it.

The result of the above configuration is that the associated alarm list will display all alarms whose priority is less than or equal to "2" (i.e. critical and high alarms), and whose area is "East". The following information will assist you in removing filtering criteria from the alarm list.

A filter can be removed by clearing it from the Alarm Filtering dialog.

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a legacy Alarm List

...(Legacy) Filter Field Button – Filter the current alarm list by a single, pre-selected option.

(Legacy) Filter Field Button

Filter a legacy Alarm List by a single, pre-selected option. Any alarm property may be used as the basis of the filter.

It is standard practice to create several instances of the Filter Field Button in order to provide a range of filtering options (including a wildcard to remove all filters).

The Filter Field Button's properties dialog:
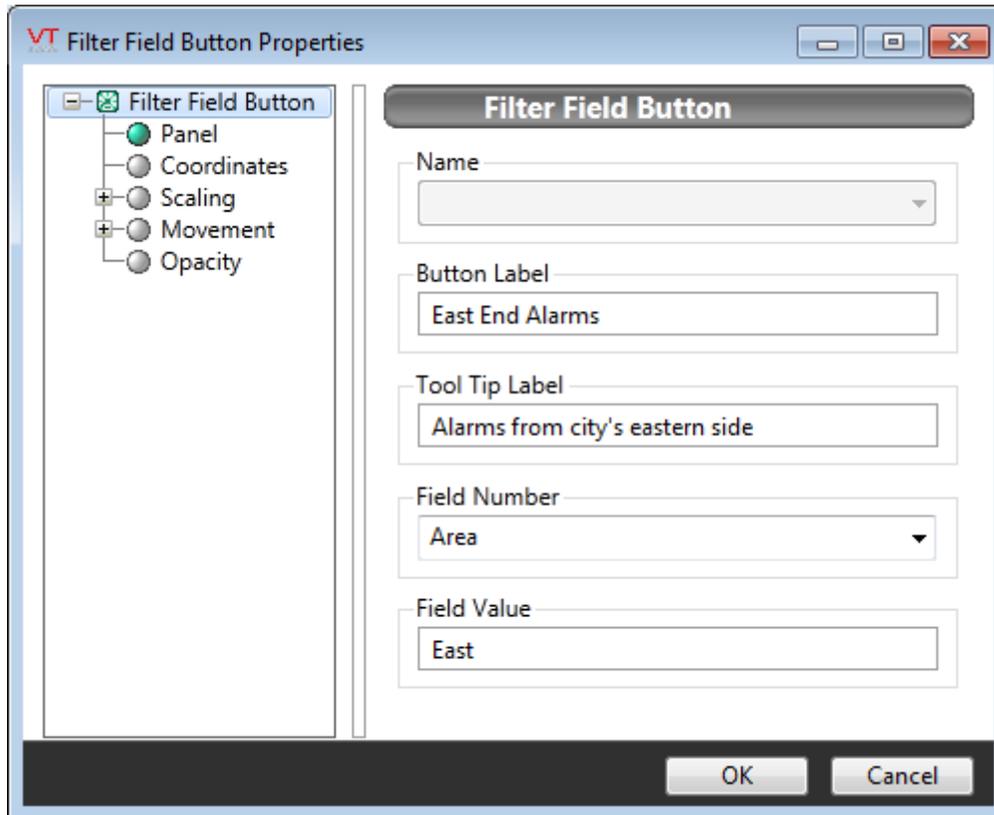
Filtering Rules

You can configure multiple Filter Field Button objects to change the filter on the alarm list according to different criteria; however, you can only apply one filter to the alarm list at a time using this method. While the alarm list is being filtered using a Filter Field Button object, the active button stays latched in to indicate the filter that has been applied to the alarm list. Clicking a second Filter Field Button object removes the first filter, releases the first button, latches in the second button, and applies the second filter to the alarm list.

Removing Filters

To remove all filters from the alarm list, you should create a single "No Filter" or "Remove Filter" button. A Filter Field Button can be configured to remove all filters from the alarm list simply by leaving its Field Value field blank. Only one button is required to remove all filters, regardless of the number of filter buttons you configure.

Name

Required to associate this button with an existing legacy Alarm List.

Button Label

Enter the text that you wish to be displayed on this button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tip Label

Enter the text that you wish to be displayed on this button's tool tip (which opens when the mouse pointer is rested on this button). Use this field to compensate for the brevity of the button label by providing a more complete description.

Field Number

Select the alarm database field by which you'd like to filter. This can be one of:

Area: Filter the alarm list based on a specific Area property.
   Note: The VTScada alarm system logs security-related events, operator control action-related events, remote configuration-

related events, and Alarm Notification System-related events for reporting in the History list on the Alarm page. These entries have the following Areas:

Security-related events have a default area of "Security".

Operator control action-related events and remote configuration-related events have a default area of "OperatorLog".

Alarm Notification System-related events have a default area of "Alarm Dialer".

Operator: Filter the alarm list based on a specific operator.

Type: Filter the alarm list based on a specific alarm type.

Note: By default, all system alarms in VTScada (security, modem, operator logging, and report) have a type of "1000". VTScada programmers may specify their own type values for custom Alarm tags, and can therefore use the Type field to filter them. Custom Alarm tags and applying type filtering is beyond the scope of this guide.

Priority: Filter the alarm list based on a specific alarm priority.

Message: Filter the alarm list based on a specific Description property.

Note that wildcard characters are not permitted. Enter the full description as it appears in the Alarm tag's Description field. In the event that a Description property has not been configured for an Alarm tag, the message for the alarm will contain the name of the Alarm tag, as it appears in its Name field.

Status: Filter the alarm list based on a specific alarm or event status.

Alarm Status	Index
Active	b1
Unacknowledged	b2
Normal Acknowledged	0
Active Acknowledged	1

Normal Unacknowledged	2
Active Unacknowledged	3
Disabled	4

Field Value

Enter a single, valid criterion by which you wish to filter the alarm list, according to the option you selected in the Field Number drop-down list.
For example: If you select "Priority" in the Field Number drop-down list, you may enter "1" (critical) in the Field Value field. The result is that when this button is clicked, the alarm list will display only those alarms that have a critical priority.

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a legacy Alarm List

...(Legacy) Filter Dialog Button – Opens a dialog to apply one or more filters to the selected legacy Alarm List.

(Legacy) Log File Date

Obsolete. Alarm history is now stored using the VTScada Historian, rather than in separate alarm history files.

(Legacy) Next Log File Button

Obsolete. Alarm history is now stored using the VTScada Historian, rather than in separate alarm history files.

(Legacy) Prev Log File Button

Obsolete. Alarm history is now stored using the VTScada Historian, rather than in separate alarm history files.

(Legacy) Sort Button

Sort a legacy Alarm List by any numeric property.

Note: Sorting is different from filtering. Sorting does not stop alarms from being displayed in the alarm list; rather, it changes the order of the items being displayed. Filtering excludes certain entries from being displayed in the alarms list. Both filtering and sorting may be applied to the alarm list simultaneously; for example, you may filter to display all alarms that have their Area parameter configured as "East" and sort the results by alarm priority.

Sorting can be removed from the list by creating a second Sort Button. In the second button, set the Field Number drop-down list to the same field as its corresponding sort button, and leave the Minimum and Maximum fields blank.

The properties dialog for the Sort Button dialog.
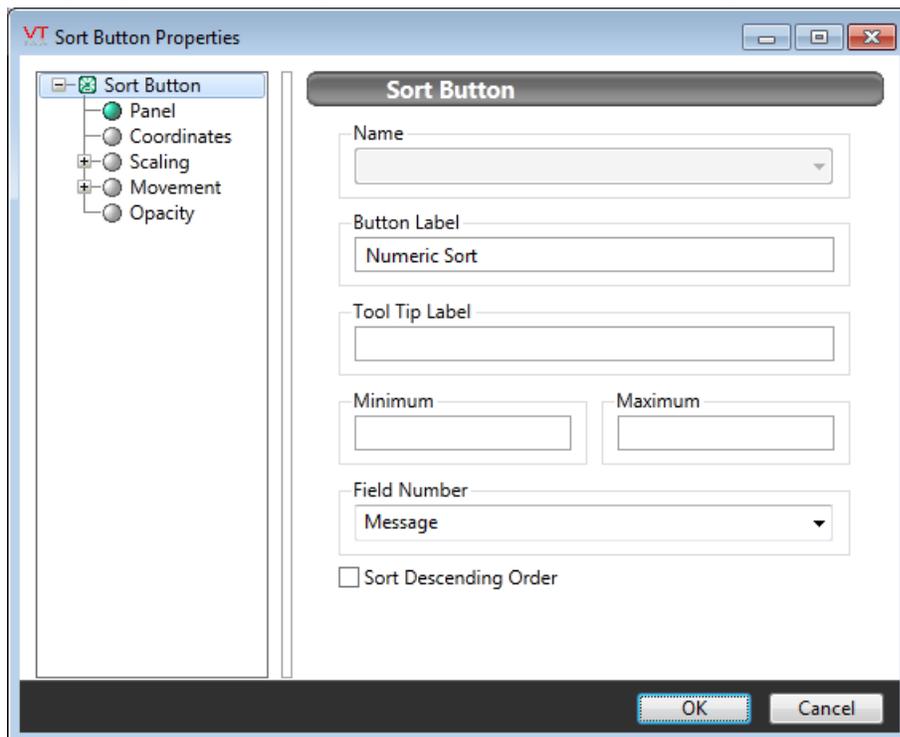
Note: The History alarm list cannot be sorted using this button.

Name

Required to associate this button with a legacy Alarm List.

Button Label

The text entered here will be displayed on this button. Labels should be kept short and precise, as the length of the label ultimately determines the size of the button.
If the label is to include an ampersand (&), then the & character will need to be doubled in order to appear.

Tool Tips Label

Use this field to compensate for the brevity of the button label by providing a longer description of the button's function. This text will be displayed only when the operator hovers the pointer over the button.

Minimum

Enter the lowest numeric value in the sort range.
The lowest alarm priority is 0 by default.

Maximum

Enter the highest numeric value in the sort range.
The highest alarm priority is "4" by default.

Field Number

Select the name of the alarm database field by which you wish to sort.
Note: this button does a numeric sort – fields with values that are non-numeric will not be sorted. Thus, while it is possible to select "Area" it is effective to do so only if areas are numbered rather than named.

Sort Descending

The Sort Descending check box can be selected if you wish to sort the alarm list in descending order (i.e. with the highest priority alarms appearing at the bottom of the .

Removing a Sort

The following image displays a completed example of a Sort Button Editor dialog that will remove priority sorting when clicked.

Related Information:

...Legacy Alarm List – This button is relevant only in relation to a legacy Alarm List

---

Email Setup for Alarms and Reports

If your VTScada license includes the Alarm Notification System option, you can send alarm notifications to operators by email, and (optionally) you can allow them to acknowledge those alarms by replying to the email.

For all VTScada installations, you are able to configure reports so that they will be delivered by email.

VTScada does not have a built-in email server. For both alarms and reports by email, you will need access to an email server and you will need to provide connection parameters for it, including the account and password to use. All account credentials are stored in an encrypted format.

> Note: Do not use a personal or business account for testing inbound email configuration. VTScada will always delete all messages from in the inbox after reading them. This is necessary to avoid having the inbox fill with old messages.

Email configuration can be accessed using the Alarms tab of the Edit Properties dialog:
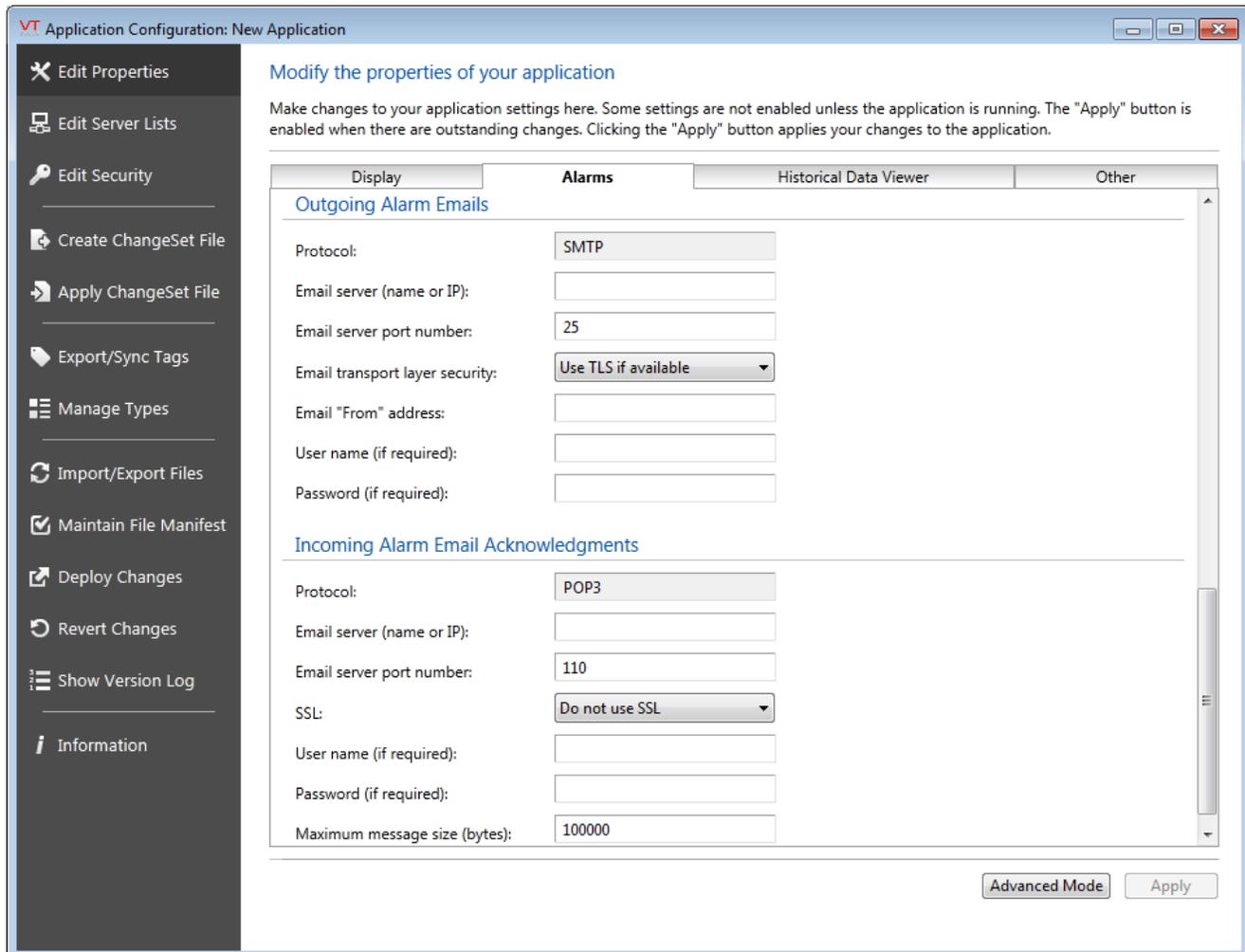
Both inbound and outbound connections can be configured here. You will need the inbound option only if your license includes the Alarm Notification System and you plan to allow alarm acknowledgment by email.

Outbound Email Configuration

VTScada is not an email server. Ensure that you have access to an email server before configuring outbound email options.

To configure VTScada to send alarm notifications or reports by email:

1. Open the Email Options dialog.
   Protocol: Only the SMTP protocol is allowed, therefore there is nothing to configure for this field.

2. Enter the name or IP address of the email server.
   Typically, this will take the form: "mail.yourcompany.com" for an internal email server or, using Google's Gmail as an example, "smtp.gmail.com".

3. Enter the port to use on your email server.
   For many local email server programs, this will be 25. If using Transport Layer Security, the port will usually be 587, but check with your email provider.

4. Select whether to use Transport Layer Security
   TLS is required by some email servers, such as Google's Gmail.

5. Provide a from address, that will be attached to your messages.
   Note that some email servers will check that this is a valid email address and will reject the message if it is not.
   If you are configuring for alarm acknowledgment by email, this field must include the email mailbox address configured in the Inbound Email tab of the dialog.
   Multiple addresses may be used, separated by semi-colons. In the case of alarm acknowledgment by email, the other addresses will be each receive a copy of the acknowledgment.

6. Provide the username and password required by the email server for authentication.
   These are not your VTScada security credentials. The values you entered will be stored in an encrypted form for your security.

Troubleshooting:

- Emails are not being sent.
  Refer to the Alarm History. If VTScada encounters an error while attempting to send an email, a report will be generated and saved to the alarm history.
- No errors reported, but emails are still not arriving.
  Check the spam rules on the receiving end. An aggressive filter may be sending your alarm or report notifications to the junk-mail folder.

Inbound Email Configuration

Warning: Do not use a personal or business account for testing inbound email configuration. VTScada will always delete all messages from in the inbox after reading them. This is necessary to avoid having the inbox fill with old messages.

To accept alarm acknowledgments by email, configure the inbound email section after correctly configuring the outbound email section.
Alarms can be acknowledged by email, only in response to an alarm notification email sent from the application. The field, Email From Address must match the account on your mail server that this page is configured to access. If any part of the server configuration is left blank, there will be no attempt to receive emails. If the server information is incorrectly configured, an event notification (visible in the alarm history) will be generated for every failed attempt to access the server.

To configure VTScada to receive alarm acknowledgments by email:

1. Open the Email Options dialog.
   Protocol: Only the POP3 protocol is allowed, therefore there is nothing to configure for this field.

2. Enter the name or IP address of the email server.
   Typically, this will take the form: "pop.yourcompany.com" for an internal email server. If using Google's GMail™ then it will be "pop.gmail.com".

3. Configure the port number to access on the email server.
   Always check with your provider to determine the port to use, but for most local email server programs, this will be 110. Google's GMail uses port 995.

4. Select whether or not the connection should be made using a secure socket link.
   If using GMail, always select the Use SSL option.

5. Enter the username and password, required to access the account on the POP3 server.

6. Adjust the maximum message size if required.
   The maximum message size is measured in bytes. This is a safety feature, preventing large incoming messages from slowing the system. Messages larger than the defined size will be deleted from the server rather than downloaded.

Troubleshooting:

- Operators send the alarm acknowledgment email back, but the alarm is not

acknowledged.

Ensure that their roster entry includes their current security account name for VTScada. An acknowledgment code is attached to emails only when the roster identifies the receiver's account name.

Related information that you may need:

...AlarmEmailTemplate – See the Application Property section of the Developer's Guide. Control the message content in outbound alarm emails.

...SMTPPort – See the Application Properties section of the Developer's Guide. System setting to hold the port number.

...AlarmCheckMail – Set the frequency for regular checking of the inbound mail account.

...AlarmCheckMailFastPoll – Set the frequency for checking the inbound email account when there are alarms to acknowledge.

Alarm Notification System – Alerts by Phone, Email or SMS-Text

Note: The Alarm Notification System is a separate feature that may or may not be enabled by your VTScada installation key. Contact a representative of Trihedral Engineering Limited for further details.

The Alarm Notification System will send alarm notifications to off-site operators by any of phone, email, pager message or SMS-text. Operators will see or hear the details of active alarms and (optionally) may acknowledge those alarms using the same format with which the alarm was sent. Only alarms with priority Critical (1) or High (2) will be sent, and those only after they have gone unacknowledged for a configured length of time. The Alarm Notification System will continue to cycle through the roster in sequence until the alarm is acknowledged, or until an email, text message or pager message that is not flagged "continue sequence" has been sent.

If a new alarm should occur while operators are being notified of existing alarms, or the first person in the active roster has already been informed of an existing alarm when the new alarm occurs, the Alarm Notification System will initiate a new contact to the first person on the active roster.

Having an operator log on remotely through the Alarm Notification System does not affect the status of any operator(s) who may be logged on at the site.

Hardware Requirements:

Hardware requirements vary according to the form in which notifications are sent.

- For spoken alarms and pager notifications, the Alarm Notification System uses a modem to make outgoing calls.
- For email notifications, configure an email server or provide access to a commercial server such as G-Mail™ or HotMail™.
- For SMS-text notifications, you will need a SMS appliance. SMS hardware is not covered in this guide.

VTScada Tag Requirements:

You will need to configure some or all of the following tags in your VTS system:

...Modem Tags – A Modem tag, if using a voice modem.

...SMS Appliance Tags – A SMS Appliance tag, if sending and receiving text messages.

...Roster Tags – One or more Roster tags, to control who is contacted, and how the notification is sent.

...Trigger Tags – Trigger tags to automatically activate Rosters on schedule. Manual switching is done with a Make Active widget.

Other Configuration:

In addition to hardware and tag configuration, the following parts may also be required:

...Alarm Area Filtering – Alarm Area Filtering or ...

...Realm Area Filtering – Realm Area Filtering, to control which alarms an operator may view.

... Speech Lexicon Configuration – Voice Lexicon configuration to control how words are pronounced.

...Make Active Widget – Draw the Rosters as Make Active buttons, to enable manual switching from one roster to another. Automatic switching is done with a Trigger tag.

...Alarm Notifications (Notification Options) – Configure which alarm priorities are to be sent and the delay before sending. Also, which menu or sub-menu should be read first in Alarm Notification System calls.

...Alarm Message Templates – Template configuration to control what is sent in the alarm message.

...Allow Acknowledgment by Email or SMS Text – Configuration requirements to allow acknowledgment.

...Accounts and Roles – VTScada Admin Guide – Configure security accounts to control who may acknowledge alarms, and to associate a user name with each alarm acknowledgment.

...Alarm Notification System Properties – VTScada Admin Guide – Related settings and properties.

Related Information:

...Alarm Data Logging – Alarm System Event Logging

...Roster Selection Hierarchy – Rules governing which roster to use for an alarm.

...Activate a Roster – Covers manual and automatic activation.

...Voice Modem Installation and Removal – Verify that a modem is compatible and install it correctly for use by the Alarm Notification System.

...Alarm System Operation – See the VTScada Operator's Guide

Roster Selection Hierarchy

The system will select which active Roster to call for an alarm by matching its Area property to the source of the alarm. The following rules are applied in order to make that match. Note that rules for Alarm Area Filtering and Tag Area Filtering take precedence – operators will not be called for an alarm that occurs in an area they are not authorized for. Whenever a device driver is referred to in these rules, it means the driver tag configured in the I/O tag. If that is a Polling Driver tag or a Driver-MUX tag, then that will be examined. No other driver tags in the communication chain will be examined.

Warning: This rule applies only to status tags, control tags and the Pulse Input. For Selector Switches, input tags and output tags, the area of the device driver will not be checked when working through the roster hierarchy. Refer to the following table:

I/O Type	Driver's Area Checked	Driver's Area Not Checked
Analog Control	●	
Analog Status	●	
Digital Control	●	
Digital Status	●	
Pulse Input	●	
Pump Status	●	
Selector Switch		●
String I/O		●
Analog Output		●
Analog Input		●
Digital Input		●
Digital Output		●

The rules are slightly different for alarms built into Status tags (First image) and those for Alarm tags (Second image).

Alarms built into Status tags:

The alarm is built into a Status tag (Analog Status, Digital Status or Pump Status)

↓

Does the area of the status tag match the area of an active Roster? — **Yes** → Use that Roster.

No

↓

Does the area of the status tag's driver match the area of an active Roster? — **Yes** → Use that Roster.

No

↓

Is there an active Roster configured with the area, "Default"? — **Yes** → Use that Roster.

No

↓

Do not call

Alarm tags:

There is an Alarm tag, triggered by some other tag.

Does the area of the Alarm tag match the area of an active Roster? — Yes → Use that Roster.

No

Does the area of the triggering tag match the area of an active Roster? — Yes → Use that Roster.

No

*-Status, *-Control and Pulse Input types only

Does the area of the triggering tag's driver match the area of an active Roster? — Yes → Use that Roster.

No

Is there an active Roster configured with the area, "Default"? — Yes → Use that Roster.

No

Do not call

Note: Support for the area property "GeneralAlarm" was removed in version 11.1.

Create a Roster Tag

Roster tags are specialized tags that can contain up to 30 contact records for operators who are to be contacted by the VTScada Alarm Notification

System when alarms have gone unacknowledged. A Roster tag may consist of multiple records for a single operator so that individual will be contacted by voice, email, and SMS Text in sequence, or you may configure one record for each operator. You may have as many Roster tags in your system as you require.

The following instructions will help to guide you through the configuration of a Roster tag.

1. Open the Tag Browser.

2. Select Roster from the Types drop-down list.
   The Tag Browser will display any Roster tags that have been configured for your application.
   By default, one Roster tag is included with every VTScada application. This tag is named Default Call-out Off, has no operator records, and is used to disable the Alarm Notification System when needed. This tag should not be modified or deleted.

3. Click New.

4. Enter a name for this Roster tag in the Name field
   (E.g. Weekend, Evenings, or Holidays).

5. In the Area field, enter an area matching that of the alarms that should be called out by this roster.
   Note: refer to Hierarchy Rules for Triggered Alarms and the Alarm Notification System and Creating Operator Groups for the Alarm Notification System for information on the Area property and the Alarm Notification System. The area field is very important to rosters.

6. Enter a description for this Roster tag in the Description field.

7. Click the Contacts tab.
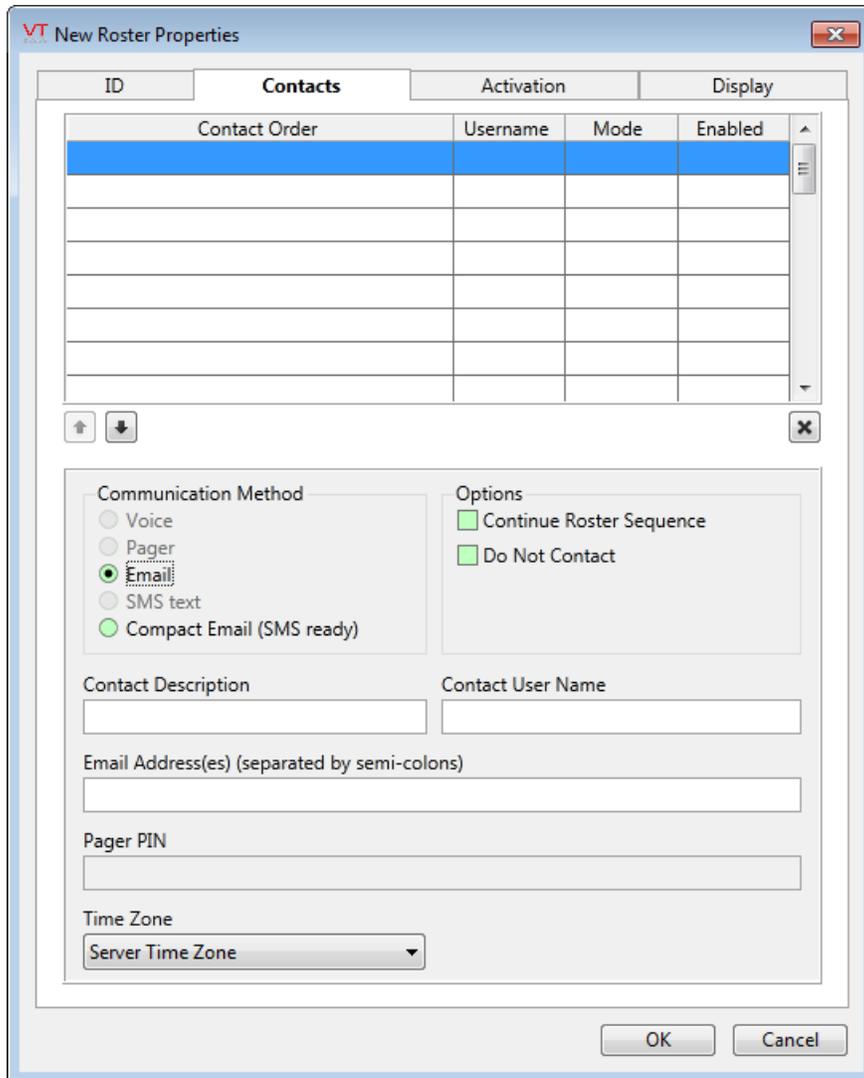
8. Enter an operator's name in the Name field.
   This will be the first operator on this roster to be contacted in the event that an alarm has gone unacknowledged, and this is the active roster.

9. Select the Communication Method radio button that reflects the contact mode you wish to use for this operator.
   The data entry fields will change according to your choice of communication method.

Note: For the SMS text option: While VTScada will always attempt to send the text message, there is no guarantee that all messages will be delivered by the cellular provider. Use the SMS option with caution.

10. Enter a phone number (Voice, SMS message), pager number (Pager), or email address (Email) in the Phone Number/Email Address field.
11. If the selected contact method for this operator is Pager, enter the pager's PIN number in the PIN Number field.
12. If the selected contact method is Email, and if you have configured your system so that alarms can be acknowledged by email, then provide the contact's User Name(*).
    Provide the user's security account name in the Contact User Name field.
13. Select the corresponding Keep Polling check box if the selected Contact Mode is Pager or Email and you wish the system to continue to contact the next operator record even if this pager or email transmission was successful.
14. Repeat for Person 2 etc. until you have added all the designated operators you require for this roster.
15. Click OK.
    The new Roster tag will now appear in your Tag Browser.

(*) Operators can be notified of alarms by email whether the Username field is completed or not. Operators can only acknowledge alarms by email if their security account username has been provided.

Troubleshooting:

- The communication method you want to select is not enabled.
  Communication methods are enabled only when VTScada is able to use those methods. For example, if you have not attached an SMS device and configured an SMS Appliance tag, then you cannot choose to send an alarm message by SMS text.

Related Information:

...Roster Tags

...Roster Selection Hierarchy

Activate a Roster

For all the Roster tags that share an area, only one may be active at a time. As one roster becomes active, all other rosters for that area become inactive. This is typically done to create separate rosters to use

at different times of day. For example, one roster for a day-shift and another for a night-shift.

Note: Note: Whether or not existing alarms are sent to a roster as it becomes active depends on the configuration of the rosters.

Alarms are not transferred: Given two rosters, the first having no contacts (since operators are assumed to be on-site) and the second having a list of contacts for when operators are not on-site. As the second roster becomes active, existing alarms are not sent to it.

Alarms are transferred: Given two rosters, each of which has a list of contacts. As the second roster becomes active, existing alarms that were being called out to the now-inactive roster will switch to the currently active roster.

Manual Activation

Rosters can be activated manually using a Make Active button. One should be drawn (on any page of your choice) for every roster in your application. All operators with access privileges to the page can then switch active rosters at the click of a button.

Automatic Activation

Rosters can be activated automatically by linking them to other tags. The Trigger tag in particular, was created for exactly this purpose. A Trigger tag can be designed to activate and deactivate at specified times or in response to some external condition, making it a perfect choice for automatic switching of active rosters.

These instructions show how to configure a Trigger tag and use it to activate a roster every morning at 9:00 a.m. Note that the Trigger tag is a powerful tool with many options. These instructions will cover only the most basic procedure.

The first set of steps will configure the trigger:

1. Open the Tag Browser.

2. Select a Trigger tag as the tag type.

3. Click New.

4. Provide a Name, Area and Description for the Trigger tag.

5. Select the On Conditions tab.
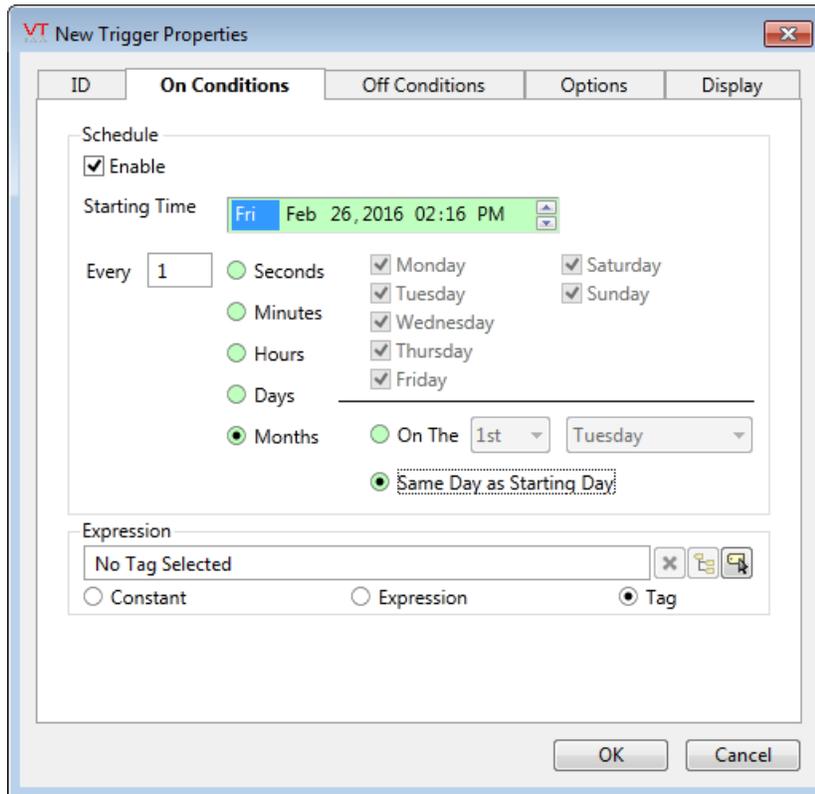
6. Set the Starting Time to today's date at 9:00 AM

7. Set the Every box to "1" and the time frame to "Days".

8. Ensure that all the days of the week are selected
   If you wish to activate a different roster on weekends, leave Saturday and
   Sunday unchecked. You can repeat this procedure for as many rosters as
   you need.

9. Leave the Expression box set to No Tag Selected.
   This option is used only if you wish to activate the trigger based on an
   external condition.

10. Select the Off Conditions tab.
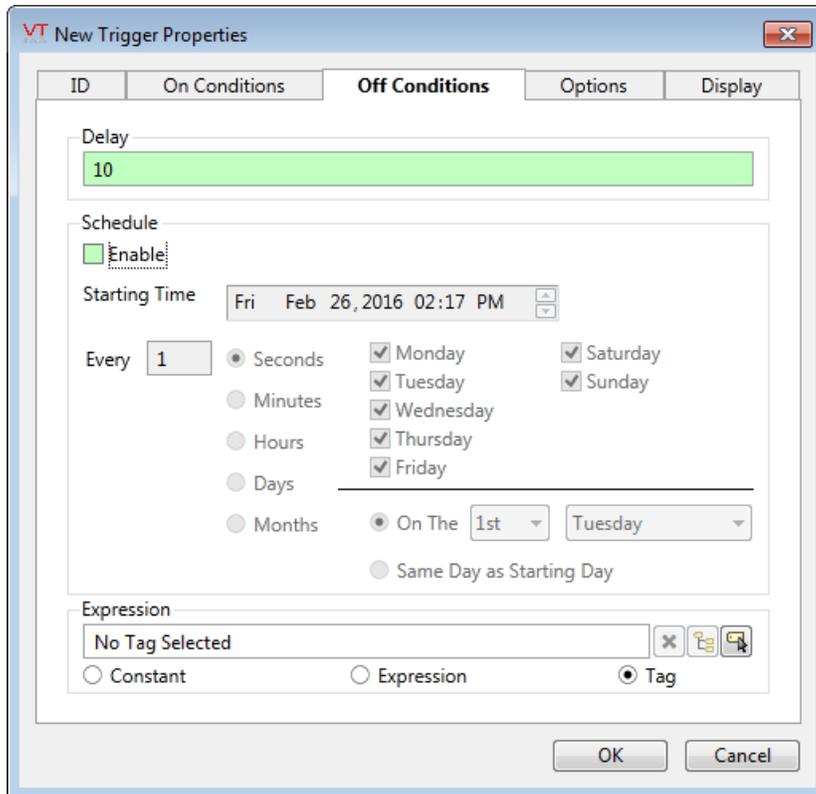
11. Enter the number "1" for the Every field

12. Press Tab or Enter to record the 1 and activate the other controls

13. Set the time frame to Days

14. Set the Starting Time to 5:00 PM on today's date.

15. Ensure that same days are selected as in the On Conditions tab.

16. Click on OK to save the new Trigger tag.

Now that you have a trigger, you can use it to automatically activate a roster, as follows:

1. Create a Roster tag, or open the properties dialog for a Roster tag you created earlier.

2. Select the Activation Tab
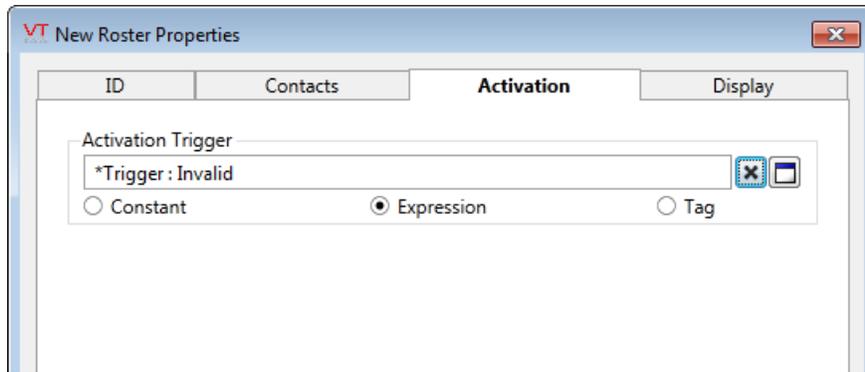
3. Select the Trigger tag you created in the last set of steps.

4. Click on OK to save the Roster.

Repeat the steps for as many Rosters as required.

Related Information:

...Roster Tags

...Make Active Widget

...Roster Selection Hierarchy

Alarm Notifications (Notification Options)

The Alarm Notifications section of the Alarms tab in the Edit Properties dialog is used to set the conditions under which registered operators will be contacted by your application's Alarm Notification System. In versions of VTScada prior to release 11, these controls were found in the Alarm Notification System Options dialog, which is now obsolete.

Configurable options include: the length of time before the notification system notifies remote operators that an alarm has gone unac-knowledged, the priority of alarms for which notifications should be sent, the greeting to use when an operator has been contacted by phone and the initial menu section that will be presented.

Dial-out Priority:

Set the category of alarms of which you wish the Alarm Noti-fication System to notify operators. This can be one of:

Action	Description
Do not dial on alarms:	The Alarm Notification System will be disabled.
Dial on priority 1 alarms only:	Only radio button, the Alarm Notification System will notify operators of alarms with the most urgent priority (i.e. alarms whose Priority property has been set to 1-Critical).

Dial on priority 1 or 2 alarms:	The Alarm Notification System will notify operators of alarms with the most urgent priority (i.e. alarms whose Priority property has been set to 1-Critical) and alarms with the next highest priority (i.e. alarms whose Priority property has been set to 2-High).

Note: This setting is stored in the system property, MaxCallAlarmPriorityReported.

Delay (secs) to Dial Priority 1 Alarms:

Set to the number of seconds you wish the Alarm Notification System to wait before notification of an unacknowledged priority 1 alarm is sent to operators. If a priority 1 alarm is acknowledged before the time set in this field, then the call-out will be aborted. The default value for this field is 30 seconds.

Delay (secs) to Dial Priority 2 Alarms:

Set to the number of seconds you wish the Alarm Notification System to wait before notification of an unacknowledged priority 2 alarm is sent to operators. (If the dial-out priority is set to this level).
If a priority 2 alarm is acknowledged before the time set in this field, then the call-out will be aborted. The default value for this field is 90 seconds.

Delay (secs) Between Calls:

Set to the number of seconds you wish the Alarm Notification System to wait between outgoing calls to designated operators. The default value for this field is 0 (i.e. no delay between calls to designated operators).

Spoken Name in Phone Greeting:

Enter a greeting to be spoken by the Alarm Notification System when initially greeting operators. The default value for this field is "The VTScada System".

Initial Section:

Use this to set the type of alarms that will be annunciated first. Options include the

Main Menu – provides access to all other sections. Operators may enter a 1 for unacknowledged alarms, 2 for active alarms, 3 for the control menu, 4 for station status or * to hang up.

Unacknowledged Alarms – if the Alarm Notification System called the operator, this will be all alarms in the call list array. If the operator phoned in to the Alarm Notification System, these will be only those unacknowledged alarms in the operator's realm.

Operators may enter a 1 to repeat the list, 2 to acknowledge alarms or * to return to the main menu.

Active Alarms – all active alarms, viewable from the operator's realm. Operators may enter a 1 to repeat the list or a * to return to the main menu.

Station Status – applies to VTScada systems that have polling stations. Operators will be prompted for a station ID, following which the system will provide the status of Pumps, digital inputs and analog status values in that order.

Control – operators may use this option to loop through the available control tags in a selected station, choosing to write a 1 or 0 for each digital output control.

Echo Phone Through Speaker:

Can be selected to command the Alarm Notification System to broadcast through the PC speakers in addition to the phone. This enables local users to listen to the Alarm Notification

System annunciate phone messages.

Allow Acknowledgment by Email or SMS Text

If you have configured the Roster to send alarm notifications by email or SMS text, you can also configure it to accept alarm acknowledgments by the same method. Operators will receive an acknowledgment code by email for each alarm in the system. By replying to the message, they acknowledge the alarm.
The acknowledgment code is tied to the specific alarm and to the operator's security account name. The acknowledgment code will not apply to any other alarm, including a later alarm triggered by the same tag, nor will it work for any other operator.

Configure for alarm acknowledgment by email:
1. Configure the Email Options dialog as follows.
   a. The "From" address is the account where VTScada will receive acknowledgments. This is the address that operator's reply will be sent to.
   b. Configure a POP3 server with the account that VTScada will use to receive alarm acknowledgments.
2. Configure the Roster so that entries that send notifications by email include the operator's security account user name.
3. Configure the settings, "AlarmCheckMail" and "AlarmCheckMailFastPoll" to control the frequency at which VTScada checks for replies.
4. Configure "AlarmEditEmailAck" to choose whether the operator will be required to edit the message before replying.
   Note that some email servers will shut down an account that polls too often, as a suspected spam source. The default value of 60 seconds for AlarmCheckMailFastPoll will trigger this on some third-party email servers.

Configure for alarm acknowledgment by SMS text:
1. Configure the Roster so that entries that send notifications by SMS text include the operator's security account user name.

2. [Optional] Ensure that AlarmNotifySMSAcknowledge is set to TRUE, to cause a follow-up text message to be sent from VTScada, confirming that the alarm was acknowledged.

The message received by the operator will appear similar to the following. The alarm code is unique to the operator, the alarm and the time of the alarm, making it extremely difficult to guess. Codes cannot be reused or given to another operator to use.

```
From: The VTScada System [mailto:The VTScada System]
Sent: December-14-11 10:43 AM
To: Joe Operator
Subject: The VTScada System: 1 alarm
12/14/2011, 10:42 AM. Zone3 Water Level LOW (23%).
Code: LPXX
To acknowledge the alarm, reply with ACK*XXXX, where XXXX is the
alarm code in this email.
```

A confirmation message will be sent upon successful alarm acknowledgment. No reply to the acknowledgment will be sent otherwise. Confirmation messages can be suppressed by setting AlarmNotifyEmailAcknowledge to false (zero).

In the Alarm History, the words "by email" will be appended to the operator's name to show that the alarm was acknowledged via email, and "by text" to show that the alarm was acknowledged by SMS.

Related Information:

...Email Setup for Alarms and Reports – Refer to the VTScada Admin Guide for details about configuring an email server to send or receive messages.

...SMS Appliance Tags – Configure an SMS device.

...AlarmNotifyEmailAcknowledge – VTScada Admin Guide – Suppress email confirmation messages.

Speech Lexicon Configuration

The Speech Lexicon dialog can be used to adjust how words are pronounced when spoken by voice engine. If using a third party speech engine that provides its own lexicon configuration, disable the

VTScada Speech Lexicon by changing the value of the property, EnableLexiconDialog to 0.

The Speech Lexicon is opened from the Alarms tab of the Edit Properties window:
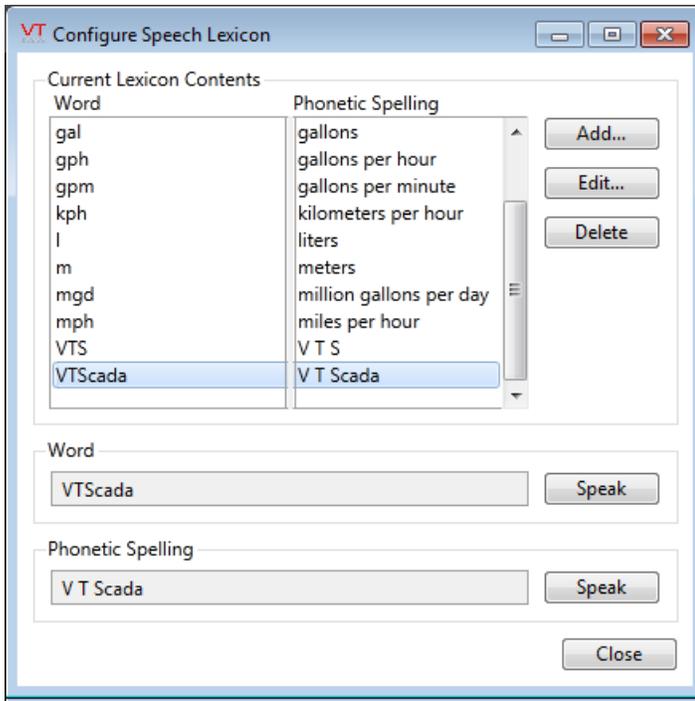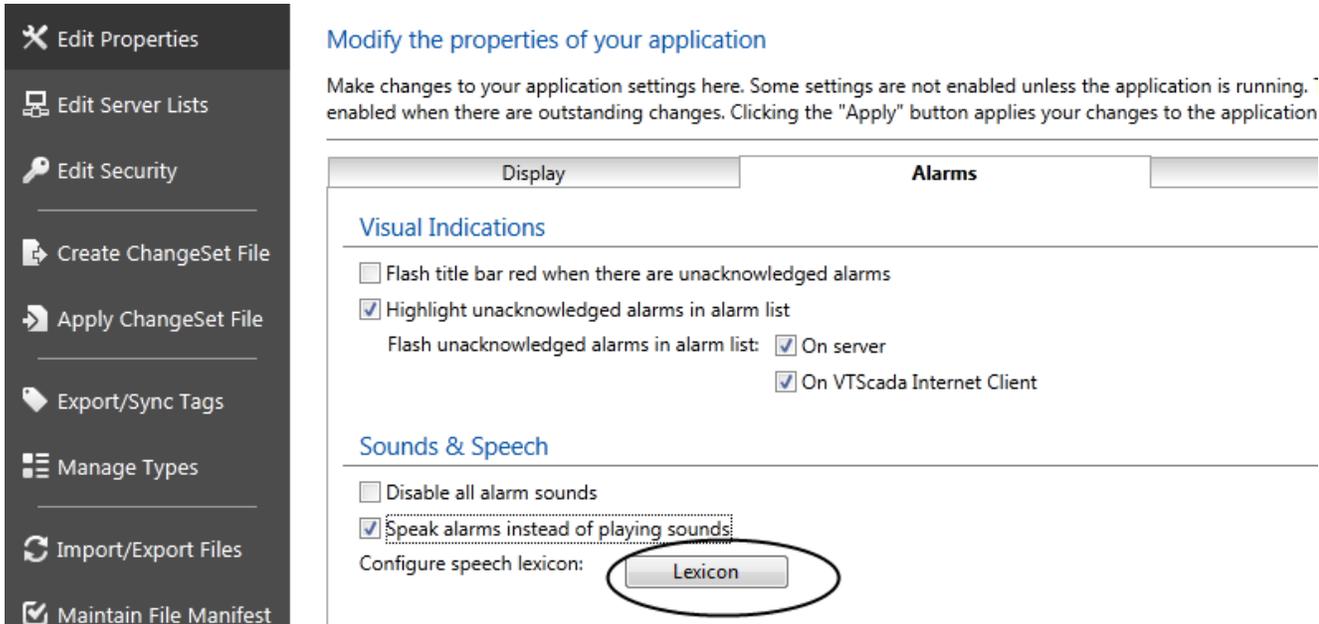
The lexicon settings are used with alarms spoken at the workstation, and with the Alarm Notification System. Although the speech engine will

pronounce many words plainly and without a flaw, you should review abbreviations, place names, operator names, and alarm messages, adjusting as required so that they are pronounced clearly by the speech engine. Also, some users may prefer a localized pronunciation for certain words.

Set Speech Pronunciation

1. Open the Configure Speech Lexicon dialog.
2. Click Add. The Add Lexicon Word dialog opens as shown.
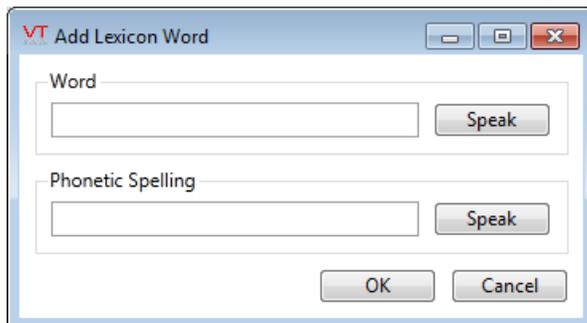
3. Enter the word, phrase, or place name in the Key field.
4. Click Speak button to the right of the Word field.
   The speech engine speaks the word, name, or phrase in the Key field.
5. Identify which syllables need to be changed.
6. In the Phonetic Spelling field, enter the word or phrase using a Phonetic Spelling Reference, rather than a dictionary-correct spelling.
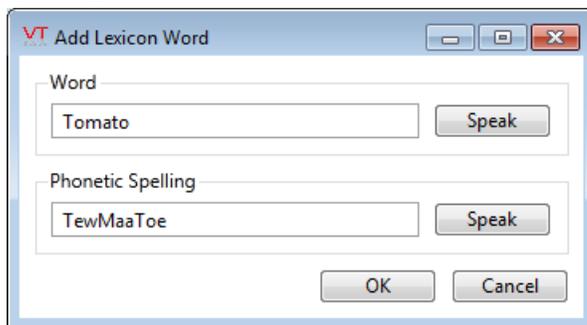
7. Click the Speak button to the right of the phonetic spelling field.
   The speech engine speaks the word, name, or phrase as you have spelled it.
8. Repeat as required to adjust the pronunciation until it sounds as you would like.
9. Click OK. The word, name, or phrase is added to the Special Words list of the Dialer Pronunciation dialog.

10. Click Add to add another word, name, or phrase, to the "Current Lexicon Contents" list. Click OK to close the dialog.

Note: Certain two-letter combinations may be pronounced as individual letters rather than their combined sound (example: HI). Others may have default words assigned (example: "AK" may be pronounced "Alaska" on some systems). Use of an underscore with a two-letter combination may help to avoid these behaviors.

You can add, test, and configure as many unusual words, phrases, names, roots, abbreviations, places, and operator names as you require.

Related Information:

...Phonetic Spelling Reference

Phonetic Spelling Reference

The following phonemes may be used in the VTScada Lexicon Dialog to control the phonemes of spoken words.

Note that, mixed case words (CamelCase[1]) will be broken into shorter syllables (lower case letters) and individually pronounced letters (upper case letters). Thus, "VTScada" is pronounced "Vee Tee Scada".

Phoneme	Example	Phoneme	Example
-	syllable boundary (hyphen)	h	h as in help
!	sentence terminator (exclamation mark)	ih	i as in fill
&	word boundary	iy	ee as in feel
,	sentence terminator (comma)	jh	j as in joy
.	sentence terminator (period)	k	c as in cut
?	sentence terminator (question	l	l as in lid

---

[1]A typographic convention where each word in a multi-word name is capitalized. So-named because the capitals visually suggest a camel's humps.

	mark)		
_	silence (underscore)	m	m as in mat
1	primary stress	n	n as in no
2	secondary stress	ng	ng as in sing
aa	a as in father	ow	o as in go
ae	a as in cat	oy	o as in toy
ah	u as in cut	p	p as in put
ao	o as in dog	r	r as in red
aw	ou as in foul	s	s as in sit
ax	a as in ago	sh	sh as in she
ay	i as in bite	t	t as in talk
b	b as in big	th	th as in thin
ch	ch as in chin	uh	oo as in book
d	d as in dig	uw	oo as in too
dh	th as in then	v	v as in vat
eh	e as in pet	w	w as in with
er	ur as in fur	y	y as in yard
ey	a as in ate	z	z as in zap
f	f as in fork	zh	s as in pleas–ure
g	g as in gut		

Alarm Data Logging

VTScada automatically stores all transactions & events that occur in the system. Every alarm configuration event is logged as is every alarm

occurrence and operator response.

VTScada applications start with two Alarm Database tags; the System Alarm DB and the System Event DB. Both store information using the System Alarm Historian tag. While rarely necessary, you may add more Alarm Databases and may create more Historians to be used by those databases.

> Warning: Do not use the same Historian to record both process values (tag I/O data) and alarm information.

The System Event DB will store all security-related events (operators logging on or off, managers changing permissions), all operational tasks that write to a device, a record of automatic report generation events, errors such as failed attempts to send email notifications, and more. The System Alarm DB stores more than just a history of alarm activations and acknowledgments. All transactional information such as changes of alarm state and changes to alarm configuration are also recorded.

Related Information:

...Logging of Operator Control Actions – Reference for what is logged.

...Logging of Security Events

...Logging of Alarm Notification System & Related Events

...Alarm Event Reference – Describes the possible event types.

Logging of Operator Control Actions

Event-priority alarms are built into all VTScada actions that write values to equipment. The following actions will be recorded in the History alarm list:

- Changing the value of an output tag
- Transmitting data via a Modem tag
- Receiving data via a Modem tag
- Creating, editing or deleting a page note. (Does not write to equipment, but is considered an operator action.)

For each operator control action-related event in the History list, the following information is logged:

- The name of the PC on which the security event occurred (e.g. [WorkstationX])
- The name of the tag that the operator used to perform the control action
- The username of the user who was logged on when the control action occurred.
- The area property of the tag.
- A message field is used to add information about specific events.

The message format is controlled by property, OperatorLogTemplate, as follows

OperatorLogTemplate = ^C ^F ^O ^I ^N

The following control codes may be used:

^M	[Deprecated] Machine name
^U	[Deprecated] User name
^T	Tag name
^D	Quoted tag description
^d	Unquoted tag description
^A	Tag area
^O	Old value
^N	New value
^C	OperatorLogChangedLabel (Property value)
^F	OperatorLogFromLabel (Property value)
^I	OperatorLogToLabel (Property value)

For example:

```
[Workstation19] [JSmith] Changed DO_2 "Motor Power" from 0 to 1
```

Logging of Security Events

VTScada logs the following events related to security to the alarm history:

- User logs on
- User logs off
- New account or role added
- Account copied
- Account modified
- Account deleted
- Administrative settings modified

For each security-related event in the History list, the following information is logged:

- The name of the PC on which the security event occurred (e.g. [WorkstationX])
- The name of the tag that the operator used to perform the security action
- The username of the user who was logged on when the security action occurred.
- The area field will be, "Security".
- A message field is used to add information about specific events.

An example of a security-related event entry in the History list:

```
Event Priority Time Date Area Message
Event Event 09:57:10 Nov 29 Security [WRKSTNX] JSmith has logged on
```

Logging of Alarm Notification System & Related Events

VTScada logs the following events related to the Alarm Notification System to the alarm history:

- An operator is notified by voice, email, SMS text or pager of one or more alarms
- An email has not reached its destination

The following events are also included in this section:

- Email configuration error (server time-out, failed attempt…)
- Scheduled report was created or not.
- Modem error.

For alarm-notification events, the following information is logged:

- The name of the operator contacted (taken from the Roster tag's contact records)

- The email address for the operator contacted
- The pager terminal phone number and the pager's phone number/PIN for the operator contacted
- The phone number for the operator contacted
- The number of alarms reported
- The number of alarms not reported (for voice calls only)
- Other messages describing the event.

Other events in this section will log the name of the logged-on user, the time of the alarm, priority and event, and a message describing the event.

- The Area for Alarm Notification System – related events is always "Alarm Dialer"
- Schedule reports will have the area, Report.
- Modem events will have the area, Modem.
- Email configuration errors will have the area, System.

Examples of Alarm Notification System-related event entries in the History list:

```
Event Priority Time Date Area Message
Event Event 09:57:10 Nov 29 Alarm Notification Email sent to ryan@tri-
hedral.com with 2 alarms
Event Event 09:56:06 Nov 29 Alarm notification Paged 453-6487/4585041
with 2 alarms
Event Event 09:55:23 Nov 29 Alarm notification Called 835-1575 - 1
alarm reported, 1 not reported
```

Note: For pager entries, the first number is the pager terminal phone number, and the second number is the pager's phone number/PIN. Also, the event descriptions for voice calls are slightly different than pager or email calls because voice calls are interactive. The Alarm Notification System might report all the alarms, it might only report some of the alarms before the operator hangs up, or it might report zero alarms if the operator does not enter their PIN.

Alarm Event Reference

Acknowledge

Recorded when an operator acknowledges an alarm.

Active

The triggering condition for the alarm is valid, and the alarm is not configured as a trip alarm.

Active Ack

An alarm is activated and acknowledged in the same transaction. This event type is included for compatibility with alarm standards, but will not occur within VTScada.

Commission

Recorded when an alarm is created. Also recorded when the tag that has the alarm is re-enabled after having been disabled.

Decommission

The tag containing the alarm has been disabled or deleted.

Disable

The disable option has been set in the alarm's configuration.

Enable

An alarm that had been disabled has been re-enabled.

Event

Used only for VTScada events such as operator control actions.

Modify

A transaction in which the alarm was reconfigured.

Normal

The triggering condition for the alarm is no longer valid (and the alarm is not a trip alarm), but the alarm has not yet been acknowledged.

Normal Ack

An alarm is cleared and acknowledged in the same transaction. This event type is included for compatibility with alarm standards, but will not occur within VTScada.

Notify

A phone call, email or text message was sent by the VTScada Alarm Notification System.

Off Normal

An alarm that is activated without becoming unacknowledged. This event type is included for compatibility with alarm standards, but will not occur within VTScada.

Purge

Orphaned alarm events are purged automatically. These records are not deleted, but are marked as purged and hidden unless you choose to view them using the control in the alarm list toolbar.

Rearm

The alarm was acknowledged, but left in an active state long enough to be rearmed. (Automatic rearming is an optional configuration property of alarms.

Recommission

Moving or renaming an alarm tag will always cause a Decommission / Recommission pair as the old name is decommissioned and the alarm is recommissioned under the new name.

Rename

Records the event of an alarm being renamed.

Shelve

Recorded when an operator shelves the alarm.

Trip

The triggering condition has occurred for an alarm configured with the Trip flag.

Unshelve

Recorded when an operator deselects the shelve option for the alarm, or when the preset time for which the alarm will be shelved has expired.

Logging and Reporting – Configuration

Values from any tag that has a numeric or text value can be saved for later use in reports or trend graphs. These tags belong to the Trenders tag group.

The work of recording data is done by a Historian tag. For many applications, the built-in System Historian tag is sufficient for all logging and needs no extra configuration. Larger applications may require multiple Historian tags in order to distribute disk I/O activity across several servers or to configure the tag to send information to a third-part database.

For logging to happen, all that is required is that there be a link between the tag with values to be logged and a Historian. In many tags, this link is configured for you. For example, Analog Status, Digital Status and Pump Status tags all have a built-in link to the Historian. (You may choose to disable that link for tags that you do not want to log.) Other tag types such as Calculation and Analog Input must be linked to the Historian through a Logger tag. The function of the Logger tag is to control the timing for when values are written and to select the Historian tag that will do the writing.

Logged values can be viewed using the Historical Data Viewer (trend graph), or they can be used in one of the reports supplied with VTScada. In both cases, developers have several options to provide access to these tools, and to control how they appear and operate.

This chapter describes the various storage options, when to use multiple Historian tags, when and how to configure Historians on multiple servers and how to view legacy data that was recorded using the now-obsolete Log Manager.

Integration With Third-Party Reporting Tools

You have several options for using third-party reporting tools with VTScada:

- If your license includes the ODBC Server option, you can use any ODBC-compatible program to query VTScada using SQL selection statements. See: SQL

Queries of VTScada Data: The ODBC Server

- VTScada is compatible with XLReporter® from SyTech Incporporated®. A datasheet is available from the Trihedral website: https://www.tri-hedral.com/wp-content/uploads/2014/10/Tech_VTScada.pdf

- VTScada is compatible with DreamReport® from Ocean Data Systems®. A datasheet is available from the Trihedral website: https://www.tri-hedral.com/wp-content/uploads/2014/10/Dream-Report-Product-Over-view-for-VTScada.pdf

Related Information:

...Report Tags

...Historian Tags

...Historian and Logger Configuration – Alternate storage, multiple Historians, multiple servers, multiple logging.

...Historical Data Viewer Configuration – Create and save pen selections. Control what information is displayed and how it is formatted.

...Report Configuration – Ensure that your report has the information you need.

...Run-Time Configuration of Report Tags – Customization tools for use with Report Tags.

...SQL Queries of VTScada Data: The ODBC Server – Treat VTScada as a relational database program.

Historian and Logger Configuration

The job of the Historian tag is to ensure that your data is recorded correctly and can be retrieved efficiently. By default, that information will be written to a proprietary storage format, located in the Data folder of your application. When the connection to a Historian is built into a tag, that tag's values will be written with every change. The use of a deadband on the source tag is recommended, in order to avoid logging system noise. When using a logger tag between a data source and a Historian, you can

choose to log on change, on a schedule, or both. You also have the option of turning logging off – for example, not recording turbidity measurements while no flow is going past the measuring device.

Note that, there can be only one simultaneous connection between a data source tag and a Historian. If a Logger tag is added to a tag that already has a link to a Historian, the result will be the loss of information. There is a way around this restriction, as described in one of the following subtopics.

While Historian tags have been designed to provide rugged, efficient service without configuration, the following options are available to you:

- Data Limiting – you can configure the Historian tag to save data from only the last X days or N records.
- Alternate storage locations. You can direct the data to be saved to a location of your choosing, either on the computer that runs the application or on another server in your network.
- Redundant storage locations. In a remote application, each server running the Historian tag can be configured to save its copy of the data to its own copy of the database.
- Alternate storage formats. Instead of using the VTScada database system, you may use any of the following databases to store data: Oracle, MS SQL Server, MySQL or SQLite.
- Load distribution between multiple Historian tags. By adding more Historian tags, each running on a separate server and saving to separate databases, you can ensure that the data logging requirements of extremely large systems does not exceed any one server's I/O capacity.

Related Information:

...Historian Tags – Reference.

...Historian Data Storage Options – Storing data in an alternate location or a third-party database.

...Configuring Multiple Historians – Load sharing for large applications.

...Historians and Multiple Servers – Backup and redundancy.

...Monitor the Historian's Connection- Create an expression to warn when the connection to the Historian is lost.

...Historian Status – Monitor all Historian tags.

...Viewing Legacy (Log Manager) Data with the Historian – Continuity for applications built before version 10.0.

...Logger Tags – Reference.

Related Tasks:

...Plot Trends and Graphs – General instructions for using the HDV are provided in the VTScada Operator's Guide.

Historian Data Storage Options

Unless otherwise configured, Historian tags will save data to a VTScada proprietary database system within the application's data folder. These files are designed for use solely by VTScada. Values can be read only by VTScada and cannot be modified by any means without damaging system integrity.

> Note: While the properties described in this topic can be set in the Edit Properties dialog, you are advised to use the configuration fields of the Historian tag instead.

You can configure a Historian tag to send its data to an alternate location, or to a third-party database. The available databases are as follows:

- SQLite, using ODBC driver version 0.86 or later
- Oracle version 10g or later. Requires the following Oracle privileges: Create any index, Create any sequence, Create any table, Create any trigger, Create session, Create user, Drop any table, Drop user, Insert any table, Select any table, Update any table
- SQLServer version 2000 or later
- MySQL using the MySQL ODBC Connector version 5.1.6 or later. Requires the following MySQL privileges: Select, Insert, Update, Delete, Create, Drop, Index

If using a third-party database, you do not need to create tables, as this will be done by VTScada. It is strongly recommended that a specific database be reserved for use only by the VTScada Historian.

Note that VTScada stores data in a form that has been optimized for performance, reliability and synchronization. Attempting to extract history data directly from a database using SQL statements may be challenging. You are advised to use the VTScada ODBC Server for direct SQL access since it has been designed to handle the complexities of the data structures for you.

If you are using the VTScada database system for data storage, then there is no need to set any value for the Historian tag's Storage Type property. You might set the Storage Location property if you want to save the data somewhere other than the default Data\History folder of your application.

If writing to a third-party DBMS, the storage location should be either a connection string in the form:

```
Driver=ServerBrand;Server=ServerName;Database=DBName;Uid=user;Pwd=pwd
```

Alternatively, it may be a reference to a DSN, as follows:

```
DSN=MyDSNName
```

Prerequisites:

If modifying the location to a new folder, then the only prerequisite is that the folder exist. Before configuring VTScada to use a third-party database, all the following must be in place:

- The database program must be installed and running.
- An ODBC driver for your program must be installed. The driver version must be 32-bit or 64-bit to match your VTScada version.
- You may use either a connection string as shown in the following examples, or you may use a DSN.
- A database must be created for use by VTScada.

- A user account must be created for use by VTScada. This account must have CREATE and WRITE privileges in the database, so that VTScada can create tables and write data to those tables.

Examples:

For example, to direct the System Historian to use a SQL Server database, identified by a connection string, the Historian tag's data store properties would be set as follows:

> StorageName: SystemHistorian
> Type: ODBC
> Storage Location: Driver=SQL Server-
> ;Server=ServerName;Database=DBName;Uid=user;Pwd=pwd

When using ODBC Historians, it is important to specify the StorageName parameter. This will be used as the schema name in the database. If left blank, the unique id of the Historian tag will be used as the schema rather than its short name.

Configure Two Data Stores

One Historian in an application that runs on a primary and backup server. Each server has its own distinct database.
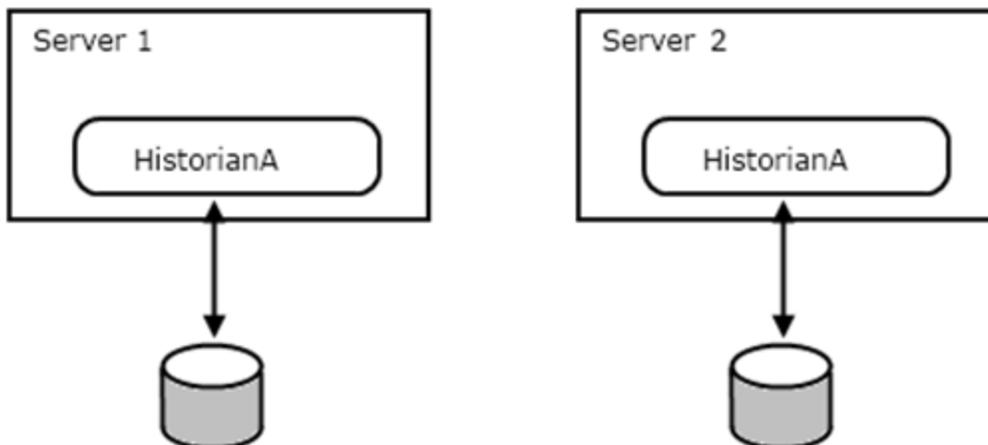
To record to both the VTScada data store and a third party ODBC database there must be at least two VTScada workstations in your system. You can then configure the Historian tags using parameter expressions

that use a different configuration on each workstation. In the following examples, "SQLBackup" is the name of the computer where the ODBC database is installed. Parameter Expressions are added to the configuration fields of the Datastore tab of the System Historian as follows. These expressions check the workstation name using "WkStaInfo(0)", and configure for the ODBC database if the workstation is SQLBackup, or use default settings otherwise, by returning "Invalid". Note that the test for workstation name is case-sensitive.

StorageName:

```
WkStaInfo(0) == "SQLBackup" ? "SchemaNameForDB" : Invalid
```

Type:

```
WkStaInfo(0) == "SQLBackup" ? "ODBC" : Invalid
```

Storage Location:

```
WkStaInfo(0) == "SQLBackup" ? "Driver=SQL Server-
;Server=ServerName;Database=DBName;Uid=user;Pwd=pwd" : Invalid
```

Note: Warning: Data store configuration should be done before the application begins to collect information. Upon changing any Historian's storage type or location, that Historian will lose access to previously collected values.

Related Information:

...Historian Properties: Datastore Tab

...StorageLocation & StorageType – In the VTScada Admin Guide, properties reference

...SQL Queries of VTScada Data: The ODBC Server – Treat VTScada as a relational database program.

Query a 3rd-Party DBMS

If you have chosen to store your VTScada data in a 3rd-party database management system, there are two factors to be aware of:

- Historian redundancy may be used to place copies of data in multiple databases. This is done to ensure reliability.

- Tables are named using the unique ID of each tag. Note that VTScada allows characters in a unique ID that may not be allowed in a table name by your DBMS, therefore the match may not be exact.
- A look-up table is provided to help you find table name for each tag. This look-up table includes the columns, "friendlyName", "uniqueID" and "table-name".

For example, to retrieve one day's data from a tag named, "Flow1", you will need to perform two queries:

```
SELECT tablename FROM friendlyTagNameLookup WHERE friendlyName =
'Flow1';
```

Use the result of this query to extract the values that you wanted:

```
SELECT Timestamp, Value FROM 'tablename' WHERE Timestamp > '2014-09-
13 00:00:00' AND Timestamp < '2014-09-14 00:00:00'
```

Related Information:

...SQL Queries of VTScada Data: The ODBC Server

Configuring Multiple Historians

In most smaller systems (ranging up to a few thousand tags) you may not need to have more than a single Historian tag. Reasons why you might consider adding one or more additional Historians include:

- Load distribution in larger systems – each Historian can be configured to save its data to a separate server, thus reducing load on any individual server.
- Alternate storage locations for selected tags – you may configure each historian to save to a different directory, or to a different storage format.
- Alternate configurations for data limiting – you may choose to limit the amount of data stored for some tags, but not limit data collected from others. Each Historian can have its own configuration.

Standard Configuration
One historian logging to the
FileDB on one server.
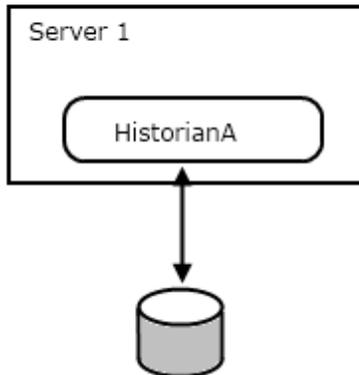
Two Historians, each using a distinct database.
The databases may on separate disk drives
and may be of different types.

Server 1

HistorianA

Server 1

HistorianA   HistorianB
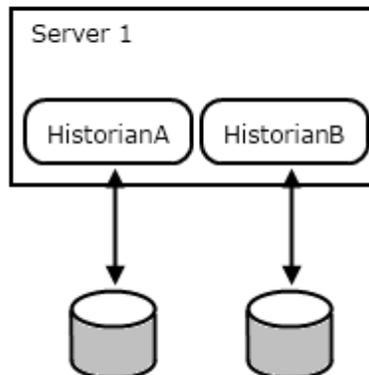
To add a second Historian, you need only create another Historian tag. Logging tags (and tags with built-in loggers such as the Analog Status) should be configured such that approximately half use one Historian, and the remainder use the other. Your ability to review data using viewing tools such as the Historical Data Viewer and the Reports page will not be affected by fact that the information is spread between different databases.

Related Information:

...Historian Status – Monitor all Historian tags.

Historians and Multiple Servers

For purposes of redundant backups or load distribution, you can add one or more backup servers to your Historian configuration.

One Historian in an application that runs on a primary and backup server.
Each server has its own distinct database.
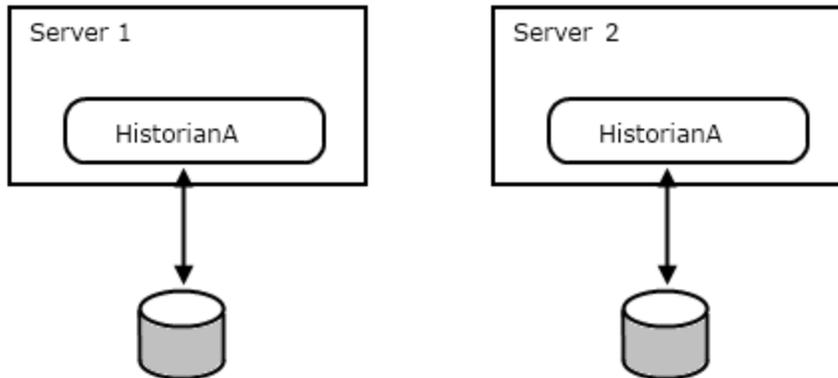
If you do no configuration of your Historian tags, then this is the situation you would have with a remote application running on a primary and backup server. HistorianA would be the System Historian, saving data to the VTScada database system on each server. The advantage of this configuration is that it creates a fully redundant copy of all data. If Server 1 is the primary, it will direct Server 2 to also write all data as it arrives. If one or the other server goes offline for a period of time, data will be synchronized when access to that server is restored.

A variation of this is to configure the Historian to save data to specific database on one server. This would most likely be one of the ODBC database options described in an earlier chapter.

Note: To prevent data loss, ODBC Historians should always have a backup server using a distinct, alternate data store. The VTScada database format is always to be preferred for reliability.

One Historian in an application that runs on a primary and backup server.
Both Historians log to the same datastore.
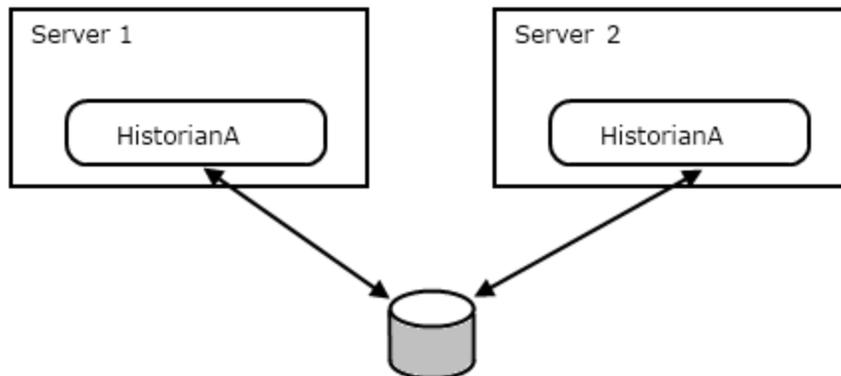
For this configuration, while both Historians will write to a single database, they will be writing to separate schemas on that database. This configuration would occur if the following configuration is one on the primary VTScada server and left unchanged when the application is installed on the remote server.

```
SystemHistorianStorageType      = ODBC
SystemHistorianStorageLocation  = Driver-
r=SQLite;Server=DBServer;Database=DBName;Uid=user;Pwd=pwd
```

There is no particular advantage to having the System Historian write to separate schemas on the same server. Therefore, if using an ODBC storage type you should consider using a workstation configuration file to set a unique StorageLocation value on each server.

Related Information:

...Configuring Multiple Historians – Instructions.

... Historian Load Distribution – A more complicated example that incorporates load distribution and redundant databases by using multiple Historians on multiple servers.

Monitor the Historian's Connection

You can use a VTScada expression to verify that any given Historian is connected to the data store on a given server. This expression may be created within a Calculation tag or as the condition for an Alarm tag.

Given a Historian named, "SystemHistorian" storing data on a server named "LogServer1" the expression to use is as follows:

```
Scope(\Code, "SystemHistorian")\GetServerStatus("LogServer1")
```

This expression will return true (1) if connected and Invalid otherwise. By monitoring the connection, you can quickly raise an alarm if there is a loss of communication.

Related Information:

...Historian Status – Library object for monitoring all Historians in an application.

...Creating Expressions – The VTScada expression language is described in the VTScada Programmer's Guide

Viewing Legacy (Log Manager) Data with the Historian

Applications that were in use and logging data prior to the introduction of the Historian tag will have legacy data. The Historian tag can read this legacy data, but it must be configured to do so.
Three application properties are used by the Historian tag to find legacy data. The property, UseLegacyHistoryPriorTo will be configured for you when you convert an older application to the current format but, you may need to adjust the values for your particular application.

UseLegacyHistoryPriorTo

> Sets a time stamp marking the latest date and time that legacy data was recorded. This will usually be the moment that the application was con-verted from one system to the other but, if your application is configured to use a time-delay when logging data, then you may need to adjust the UseLegacyHistoryPriorTo value to account for the offset.

LegacyHistoryPath

> Required only on a workstation that does not have its own copy of the data. Typically set to a network path, identifying the machine and the folder (usu-ally "History") where the legacy data is stored.

LogPath

> This legacy property predates the Historian tag. If present, it should be left unchanged.

> Note: Legacy data is not synchronized between servers in a remote application. It exists only on the computers that you explicitly copy it to. You should copy it to the primary server and all backup servers to ensure that the data is continuously available. In a remote application, copy the data from whichever server is deemed to have the most complete copy of the data.

Should a pure client lose its connection to the remote application, it will not have access to the legacy data.

Related Information:

Refer to the VTScada Admin Guide for the following properties:

...LegacyHistoryPath – Properties reference.

...LogPath

...UseLegacyHistoryPriorTo

Configure Two Logs for One Tag

You may need to create two methods of logging the data from a single tag. This might be required if you have something that rarely changes in value, perhaps a holding pond, and need to show that the value was measured and recorded at a regular interval.

> Note: Rule: You must not configure two loggers on the same tag, or add a logger to a tag that has built-in logging.

You can follow the rule, but still log the tag's values twice by adding a Function tag that mirrors the value of the original tag, and log that.

To add a second logger:

1. Create a Function tag.
2. In the Inputs tab of the Function tag, select the tag you want to mirror.
3. Ensure that the Function tag has an attached Logger, configured to the time interval required.

For reporting and trending purposes, you will need to select both the original tag and the Function tag. (Or, whichever you are interested in for a particular report.)

Historical Data Viewer Configuration

The Historical Data Viewer (HDV) is a standard part of every VTScada application. Operators use this page to view live and historical data on a graph or to list the data values in a grid. Instructions for use can be found in the chapter, Using the Historical Data Viewer Page.
You can provide several methods of access to the HDV in order to make it easy for operators to see the information they need. Options include:

- The HDV link in the main menu – opens the main HDV page.

- A Page Button or Page Hotbox – also opens the main HDV page.

- The HDV button or HDV hotbox in the Standard Library – opens a customized HDV page. See: Open HDV Button and Open HDV Hotbox.

- Drawing methods for input tags will open an HDV window when the operator clicks on them, but this feature may be switched off by checking the Disable Trends option in the properties dialog for the widget.

- Many tag types (analog and digital, input and output) can be drawn using the Plot Data widget. Use this to add an HDV display for a single tag to any page. See Plot Data Drawing Method.

The image shows a Historical Data Viewer interface with labeled callouts. Labels around the toolbar read: Toolbar, Analog graph, Digital graph, Time scale and slider. The toolbar contains "Inflow Rates" (Tag Selection), Export icons, "1 Day" (Time Span), Pan & Zoom icons, View icons, and Note icon.

The analog graph displays a sine-wave pattern with a y-axis scale of 0, 50, 100. Below it is a digital graph (square wave). The x-axis shows times: 50, 09:52, 09:54, 09:56, 09:58, 10:00, 10:02, 10:04, 10:06, 10:08, 10:10, 10:12, 10:14, 10:16.

		Pen	Name	Description	Value	Maximum	Minimum	Average	Starts	On Time
👁	⚙	∿	...\Tank Level	Monitor tank level		9.5	91.3	48.6		
👁	⚙	⊓	...\Running State	Pump running status		0	1	0.449	12	809
👁	⚙	▤	SystemNotes							

Plot | Grid | Notes

Choose what to display — Tag list with statistics for the visible time frame.
Access pen properties
Hide pen

Related Information:

...Selecting Pens (Tags) to Plot

...Configuring Pen Display Properties

...Draw HDV Widget

...Open HDV Button

...Open HDV Hotbox

...Plot Data Widget

Selecting Pens (Tags) to Plot

The Historical Data Viewer offers several ways to configure the selection of pens (tags) to be included in a plot. By pre-configuring a set of pen groups, you make it easy for operators to see the information they need without searching and without accidentally selecting the wrong pen. **Terminology**: Since the HDV displays a graph of tag data, tags are referred to as *pens* in the selection. A named set of tags is referred to as a *group*.

Related Information:

...Using the Tag Selector – How pens are chosen.

...Using a Query to Select Tags

Related Tasks:

...Save a Pen Group

...Load a Pen Group

...Modify a Pen Group

...Remove a Pen or Query from the Selection

...Delete a Pen Group

Using the Tag Selector

The Tag Selector dialog is used to add pens to and remove pens from the display. It provides filters that allow you to find tags quickly, and you can create filter-based queries that automatically select all tags matching the filter.

You can save named pen groups so that you do not need to select individual tags each time you open the HDV. Pen groups can be modified, copied and deleted.

Click on the Tag Selector button to open the selection dialog.

Tags shown in the Tag Selection list can be added to the Selected Tags list by either of two methods:

- Double-click on a tag to add it to the list
- Click once on the tag, then once on the Select Tag button near the bottom of the dialog.

Instead of selecting tags, you could set up a filter then, use the Add Query button to select all the tags that match the filter.

Instructions for using all of its features can be found in the following sections.
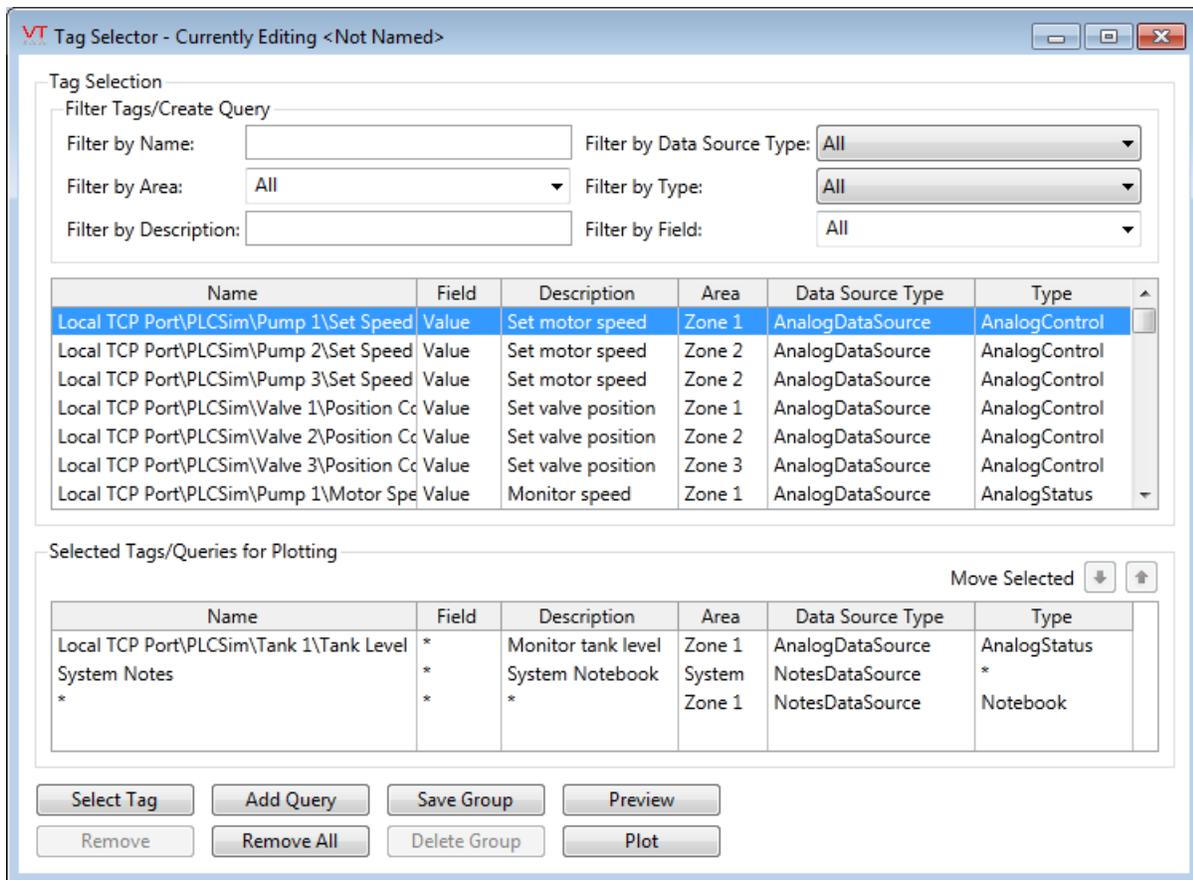
The title bar of this dialog displays more than just its name. Look closely at the following example:
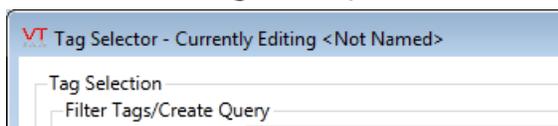
Note the phrase: "Currently Editing..." This shows you the name of the pen group being modified using the selector. If you have not yet saved a pen group then the title will read "Currently Editing <Not Named>". Below the title is the Tag Selection area, headed by the option to Filter Tags or Create a Query. The filtering process is the same for both options.

Use the various Filter By fields to limit the display of tags to only those having matching names, areas, types, etc. Having filtered the list, you can either select individual tags from that list, or you can save the filter as a query in order to take all matching tags. A query can be especially useful on systems where new tags are constantly being created.

Related Tasks:

...Save a Pen Group – Steps to create a group.

...Load a Pen Group – Make use of a group you created.

...Modify a Pen Group – Change the selection of pens in a group.

...Delete a Pen Group – Remove a group that is no longer needed.

Using a Query to Select Tags

A query is a request for all tags that match a given criteria. It is saved as a text representation of the filter created using the fields at the top of the Tag Selector dialog.
For example, the following filter settings:
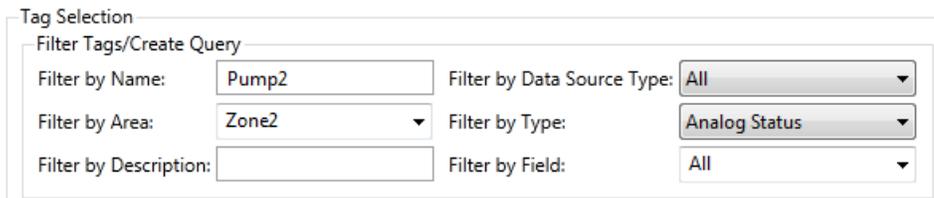
… will be saved as the following query:
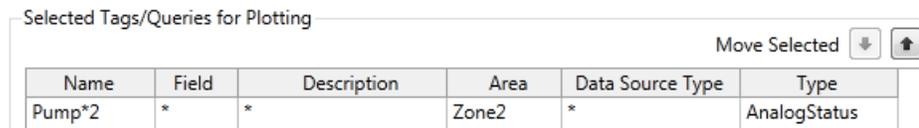
The query will plot all the tags that match the selection filter. Be careful that this is not an excessive number of tags. The Preview button can be used to find out how many tags match the query before you attempt to plot them.
Queries have the advantage of being flexible. If new tags are added to the system that will match the query's filter, you do not need to update the selected tag list – the query will find them the next time the HDV is used.

The steps to create a query are as follows:
1. Open the Tag Selector dialog from the Historical Data Viewer.
2. Define the filtering criteria using the fields in the Filter Tags/Create Query sec-

tion of the Tag Selector.

3. [optional] Click the Remove All button.

While you may combine multiple queries or select queries and tags to get a list which is the sum of both, it is more common to run only one query at a time.

1. Click the Add Query button to add the query to the list of Selected Tags / Queries for Plotting

2. Click the Preview button to check that the tag list from the query is what you expect.

3. Click the Plot button to return to the HDV and plot the tags.

You can view the individual tags that match the query by using the Preview button, as described in the next topic.

Related Tasks:

...Filtering the Tag Selection and Creating a Query

...Preview the Expanded Tag List

Related Information:

...Using the Tag Selector - How pens are chosen.

Filtering the Tag Selection and Creating a Query

When the Tag Selector dialog initially opens, it displays a list of all the tags in the application that could potentially be plotted. This may easily include thousands of tags.

The Tag Selector enables you to filter the tags in your application so that you can find certain ones more quickly. The following instructions are provided to assist you in filtering tags using the Tag Selector.

While instructions are provided for each type of filter individually, you may use several at once to help locate specific tags.

Filter Tags by Name

You may filter the list of tags by the unique name with which they have been configured (i.e. the Name property configured in the tag properties folder for each tag).

1. Click the Tag Selector button appearing in the Historical Data Viewer page's tool bar.

     The Tag Selector dialog will open.

2. Enter one or more characters in the Filter by Name field.
     You may use a combination of characters and the asterisk wildcard character (*) to locate a specific tag by its configured name (e.g. "130*" will find all tags whose name begins with "130" or "*B*" will locate all tags whose name contains the letter "B").

3. Press the Enter key on your keyboard to input the search characters.

The tags matching the name or characters you entered will be displayed in the Tag Selection list. You may now plot the located tag(s) adding them to the Selected Tags/Queries for Plotting list.

> Note: To reset the Filter by Name field (thereby removing name filtering from the list), ensure that no characters are entered in it and then press the Enter key on your keyboard.

Filter Tags by Area

You may filter the list of available tags by the area with which they have been configured (i.e. the Area property configured in the tag properties folder for each tag).

1. Click the Tag Selector button appearing in the Historical Data Viewer page's tool bar.

     The Tag Selector dialog will open.

2. Select the area by which you wish to filter the available tags in the Filter by Area drop-down list.
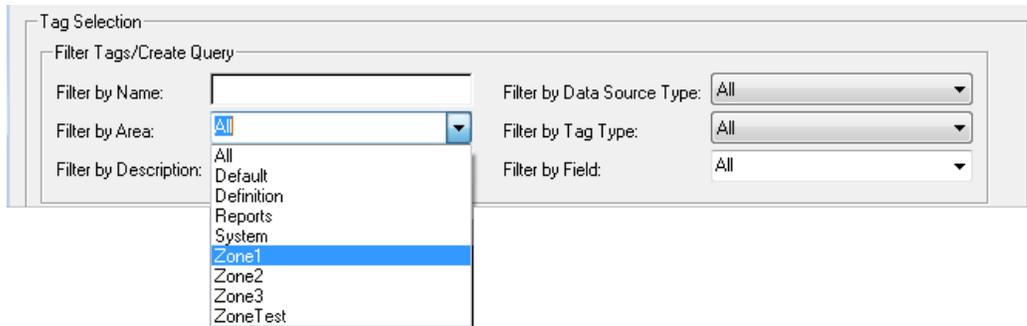
The tags matching the area you selected will be displayed in the Tag Selection list. You may now plot the located tag(s) adding them to the Selected Tags/Queries for Plotting list.

> **Note:** To reset the Filter by Area drop-down list (thereby removing area filtering from the list), select All.

Filter Tags by Description

You may filter the list of available tags by the description with which they have been configured (i.e. the Description property configured in the tag properties folder for each tag).

1. Click the Tag Selector button appearing in the Historical Data Viewer page's tool bar.
   The Tag Selector dialog will open.

2. Enter one or more characters in the Filter by Description field. You may use a combination of characters and the asterisk wildcard character (*) to locate a specific tag by its configured description (e.g. "Well*" will find all tags whose description begins with "Well" or "*3*" will locate all tags whose description contains the number "3").

3. Press the Enter key on your keyboard to input the search characters.

The tags matching the name or characters you entered will be displayed in the Tag Selection list. You may now plot the located tag(s) adding them to the Selected Tags/Queries for Plotting list.

Filter Tags by Their Data Source Type

You may filter the list of available tags by the type of data that they have been configured to read or write (i.e. analog, digital, or notebook).

1. Click the Tag Selector button appearing in the Historical Data Viewer page's tool bar.
    The Tag Selector dialog will open.

2. Expand the Filter by Data Source Type drop-down list in the Filter Tags/Create Query section of the Tag Selector dialog.

3. Select one of the following:

    -   All: Display all data types.

    -   AnalogDataSource: Display only those tags that read or write analog (continuous or decimal) data (e.g. "123.54" or "62.0") (i.e. analog input and analog output tags).

    -   DigitalDataSource: Display only those tags that read or write digital (discrete) data (e.g. "1" or "0") (i.e. digital input and digital output tags).

    -   NotesDataSource: Display only notebook tags.

The tags matching the data source type will be displayed in the Tag Selection list. You may now plot the located tag(s) adding them to the Selected Tags/Queries for Plotting list.

Filter Tags by Tag Type

You may filter the list of available tags by the their type (analog input, digital status, etc.) The list of tag types available will be influenced by the selected Data Source filter. For example, if you are filtering for only

analog data source type, only analog tag types will be available for selection in the tag type filter.

1. Click the Tag Selector button appearing in the Historical Data Viewer page's tool bar.

   The Tag Selector dialog will open.

2. Expand the Filter by Tag Type drop-down list in the Filter Tags/Create Query section of the Tag Selector dialog.

3. Select one of the tag types.

The tags matching the data source type will be displayed in the Tag Selection list. You may now plot the located tag(s) adding them to the Selected Tags/Queries for Plotting list.

Note: To reset the Filter by Tag Type drop-down list (thereby removing tag type filtering from the list), select All.

Related Information:

...Using the Tag Selector – How pens are chosen.

Next Steps:

...Preview the Expanded Tag List – Ensure that the query returns the tags you want.

...Save a Pen Group – Steps to create a group.

Preview the Expanded Tag List

The Historical Data Viewer page enables you to view a dialog that provides details about the tags associated with the pens you've selected for plotting. This is especially useful when a query has been used to

assemble a group of tags for plotting, and you wish to view the details of the tags belonging to the group.

The dialog is opened by clicking the Preview button.

The Preview Expanded Tag List dialog displays a list of the tags whose data will be plotted, along with the following information:

- The name, area, and description for each tag whose data will be plotted
- The field (tag property) to be plotted (typically the value of each tag)
- The type of data to be plotted (AnalogDataSource for analog tags (analog input and analog output tags), DigitalDataSource for digital tags (digital input and digital output tags), and NotesDataSource for notebook tags)
- The tag type for each tag whose data will be plotted

Save a Pen Group

Note: If user realms are in use, then the pen group you save will be available only to members of your security group.
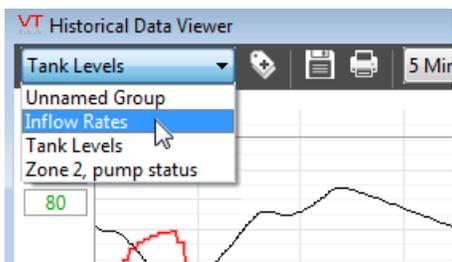
To save a pen group:

1. Open the Tag Selector from the Historical Data Viewer
2. [optional] Define filters to reduce the length of the tag selection list

3. Select tags from the list or click Add Query to save the filter as a query.

4. Click the Save Group button.
    The Save Group dialog will open.

1. Enter a name for the new pen group in the Save Group field.

2. Press the Enter key on your keyboard to input the new name.

3. Click OK.

Troubleshooting:

- Unable to use a particular name
    Group names may not be numeric and may not contain most punctuation characters. Select a new name.

- Group contains unwanted tags, or is missing desired tags.
    Groups can be edited. Add or remove tags are required, then save the group, using the current name.

Related Information:

...Using the Tag Selector – How pens are chosen.

...User Groups – Security namespaces are described in the VTScada Admin Guide

Next Steps:

...Load a Pen Group – Make use of a group you created.

...Modify a Pen Group – Change the selection of pens in a group.

...Delete a Pen Group – Remove a group that is no longer needed.

Load a Pen Group

Once you have created a pen group, you can load it onto the Historical Data Viewer page's graph at any time.

1. Expand the pen group drop-down list in the Historical Data Viewer page's tool bar. (The drop-down list to the far left in the tool bar.)

2. Select the name of the pen group you wish to load.

All pens belonging to the selected pen group will be plotted on the graph on the Historical Data Viewer page.

Troubleshooting:

- Unable to see plotted values in the graph.

Check pen configuration to ensure that all are being plotted with a visible line.

Check that there are values for the selected pens. If all values are a flat line at the top or bottom of the range, they may not be easily visible. If the tags are not logged, it may be necessary to reduce the time scale to see live values being plotted.

If security groups and realm-area filtering are configured, you may not have the right to view the selected pens.

Related Information:

...Using the Tag Selector – How pens are chosen.

...Save a Pen Group – Steps to create a group.

Next Steps:

...Modify a Pen Group – Change the selection of pens in a group.

...Delete a Pen Group – Remove a group that is no longer needed.

Modify a Pen Group

Once you have created a pen group, you can modify the tags belonging to it at any time.

1. Expand the pen group drop-down list in the Historical Data Viewer page's tool bar (i.e. the drop-down list to the far left in the tool bar).

2. Select the name of the pen group you wish to modify.

3. Click the Tag Selector button.
   The Tag Selector dialog will open, and the text Currently Editing <Pen Group Name> will appear in its title bar (where <Pen Group Name> is the name you configured for the selected pen group).

4. Use the Select Tag and Remove Tag/Query buttons in the Tag Selector dialog to modify the Selected Tags list until it contains only the tags you wish to belong to the selected pen group.

5. Click the Save Group button.
   The Save Group dialog opens and displays the name of this pen group.

6. Click OK to modify the referenced pen group.

7. Click the Plot button to close the Tag Selector dialog and plot all pens belonging to the group on the graph on the Historical Data Viewer page.

Troubleshooting:

- Wrong group selected for editing.
  The tag selector's title will state the name of the group being edited.
  If changes have been made to the wrong group and you have Version Con-
  trol as part of your VTScada license, you may reverse the change. Other-
  wise, edit the groups to restore the pen selection.

Related Information:

...Save a Pen Group – Steps to create a group.

...Load a Pen Group – Make use of a group you created.

...Modify a Pen Group – Change the selection of pens in a group.

Next Steps:

...Delete a Pen Group – Remove a group that is no longer needed.

Remove a Pen or Query from the Selection

Tags can be removed from the Tag Selection list either one at a time or
all at once.
To remove a single pen or query from the graph:

1. If the tag selector dialog is not open, click the Tag Selector button to open it.

2. Select the tag to be removed from the Selected Tags list (see image).
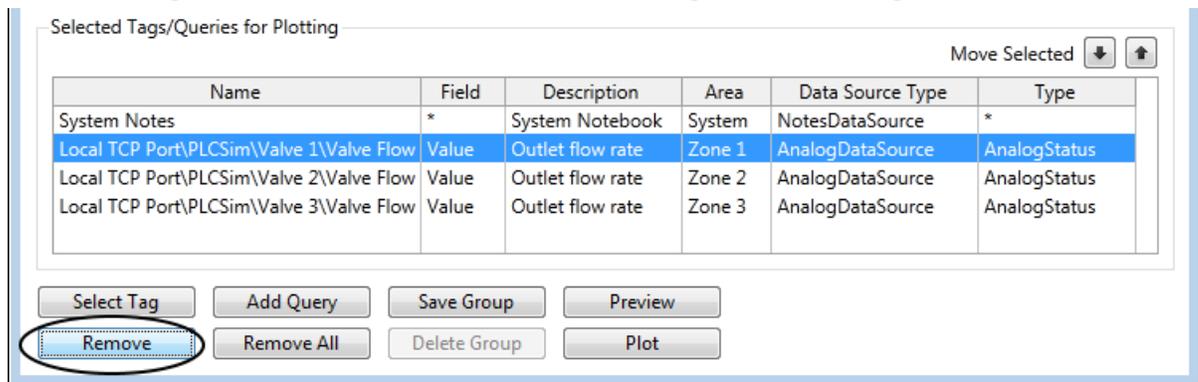
3. Remove it by either double-click on the tag, or clicking on the Remove
   Tag/Query button.

4. Click, Save Group.

The selected pen will no longer will be displayed on the Historical Data
Viewer page's graph.

To remove everything from the list of Selected Tags, click the Remove All button.
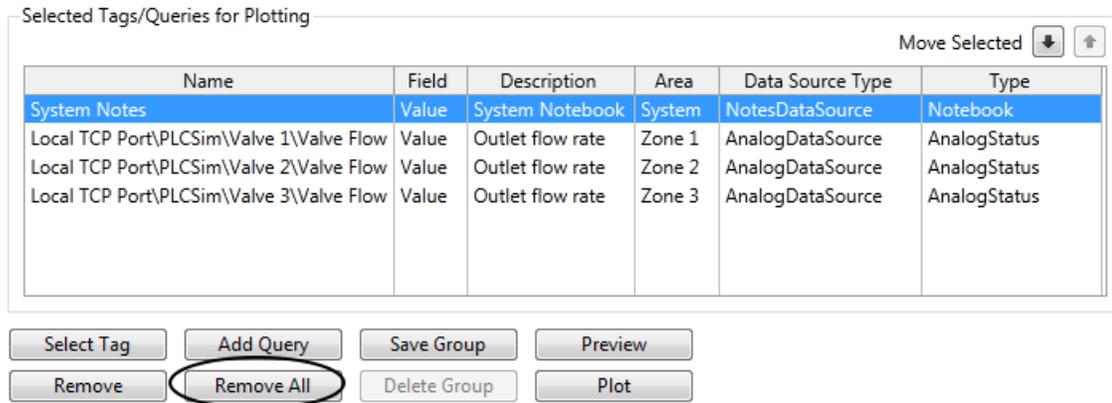
Troubleshooting:

- Pens are still plotted

  Changes were not saved. Click the Save Group button before exiting the tag selector.

Delete a Pen Group

To delete an existing pen group, provided that your account has the required security privilege:

1. Expand the pen group drop-down list in the Historical Data Viewer page's tool bar (i.e. the drop-down list to the far left in the tool bar).
2. Select the name of the pen group you wish to delete.
3. Click the Tag Selector button.

   The Tag Selector dialog will open, and the text Currently Editing <Pen Group Name> will appear in its title bar (where <Pen Group Name> is the name you configured for the selected pen group).
4. Click the Delete Group button.

   The Delete Group? dialog will open and request confirmation that you wish to delete the referenced pen group.
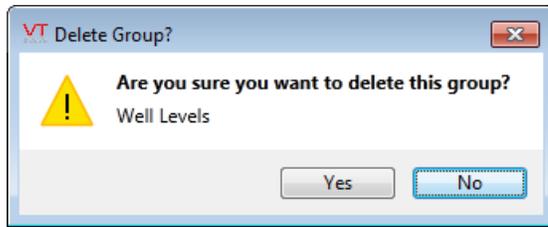
5. Click the Yes button.

   The tags belonging to the selected pen group will be removed from the Selec-
   ted Tags/Queries for Plotting list in the Tag Selector dialog.

6. Click the Close button.

   The Tag Selector dialog will close, and the tags belonging to the deleted pen
   group will removed from the graph on the Historical Data Viewer page.

Related Information:

...Using the Tag Selector - How pens are chosen.

Configuring Pen Display Properties

You can change both the appearance of the Historical Data Viewer's
graph and the information that it contains by adjusting the properties of
the pens being plotted. The list of available properties depends on the
type of pen being plotted. Analogs, digitals and note pens have a small
number of properties in common such as color, line style and width, but
more properties are unique to each pen type.

For any pen, its properties can be adjusted by opening the properties dia-
log. Click the gear icon beside the pen name or by double-clicking its
entry in the legend.
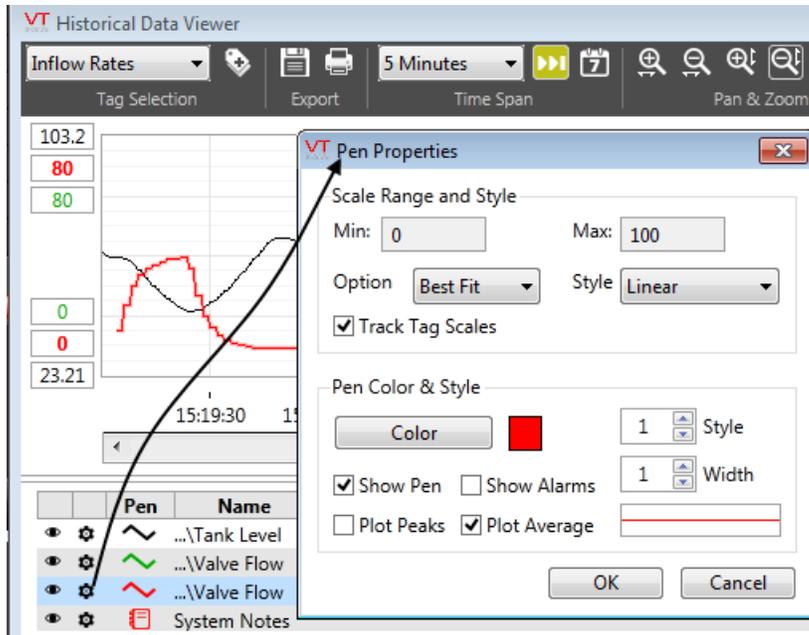
Related Information:

...The HDV Pen Legend – What the legend can show.

Related Tasks:

...Configure Analog Pen Properties

...Configure Digital Pen Properties

...Configure Notebook and Note Properties

...Fix Pen Color Configuration Problems in a Remote Desktop Session

The HDV Pen Legend

The pen legend appears across the bottom of the Historical Data Viewer page. It acts both as a guide to the pens that are plotting data on the graph, and as a means of displaying additional relevant data for each. An example of the pen legend is displayed here:

Analog plots are indicated in the pen legend by the $\sim$ symbol; digital plots are indicated in the pen legend by the ⌐ symbol; and notebooks are indicated in the pen legend by the ▭ symbol.

A pen that is selected in the pen legend appears with a rectangle around it (e.g. the first pen in the example above is selected). To select a pen, click it in the legend. To deselect all pens, click the selected pen again. There are columns for both analog and digital values. Where they do not apply to one type or the other, the entry will simply be blank.

The operator can move the cursor across the graph in order to see a the value of each tag for any time displayed. Note that this may be an average of several logged values covered by one pixel on the displayed graph, rather than each logged value.

Related Tasks:

...Configure Analog Pen Properties

...Configure Digital Pen Properties

...Configure Notebook and Note Properties

Configure Analog Pen Properties

The appearance of the pens that are used to display analog plots can be adjusted as follows:

- Double-click the entry, or click the gear icon to open the properties dialog.
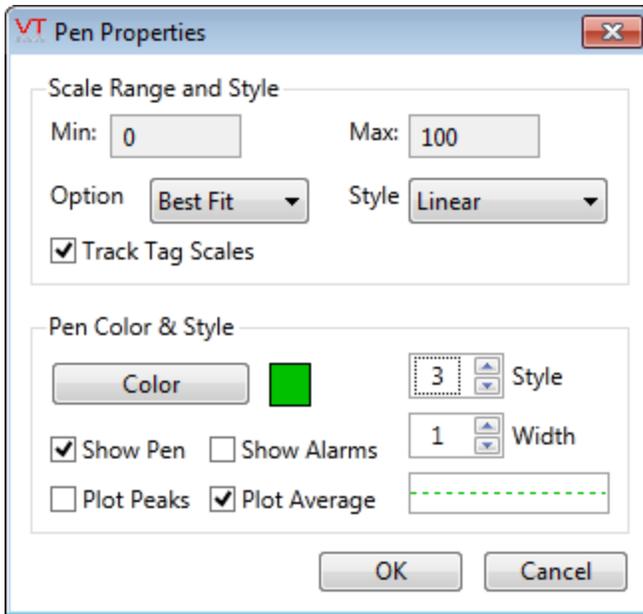
Analog Pen Properties Element	Description
Min & Max	The range of the displayed scale. If the Track Tag Scales check box is not selected, then you can use these fields to provide your own scale ranges for the tag.
Scale Option	Choose whether to display a full scale (all scales), a best fit scale or no scale.
Track Tag Scales	When selected, the scales configured for the associated tag are used for the scale range (see: Min & Max). If not selected, you can provide your own values for the scale range.
Scaling Style	Select one of the three possible graph types: Linear, Logarithmic, Square Root
Color	Opens the Select Color palette, which you can use to select a new color for the pen.
Pen Style	A spin-box that you can use to change the line style (solid, dashed, etc.). Affects the choice of width, in that you cannot select a wide line for any style other than solid.
Width	A spin-box that you can use to set the width of the pen. Affects the choice of style, in that choosing a wider line forces the line style to be solid.
Show Pen	Hides this tag's pen when not checked.

Show Alarms	Add horizontal lines to the graph that show the current value of alarm set-points associated with this tag.
Plot Peaks	When each pixel represents several logged values, the graph is normally an average of the values represented by each pixel. If Plot Peaks is selected, then two lines will be drawn – one for the maximum of those values covered by a pixel and one for the minimum. See also, Plot Average
Plot Average	The normal graph through for a pen – each pixel shows the average of the logged values represented by a pixel since there are usually more logged values than displayed pixels over a time span. See also, Plot Peaks.

Related Information:

...Configure Digital Pen Properties

...Configure Notebook and Note Properties

Configure Digital Pen Properties

The appearance of the pens that are used to display digital plots can be adjusted as follows:

- Double-click the entry, or click the gear icon to open the properties dialog.
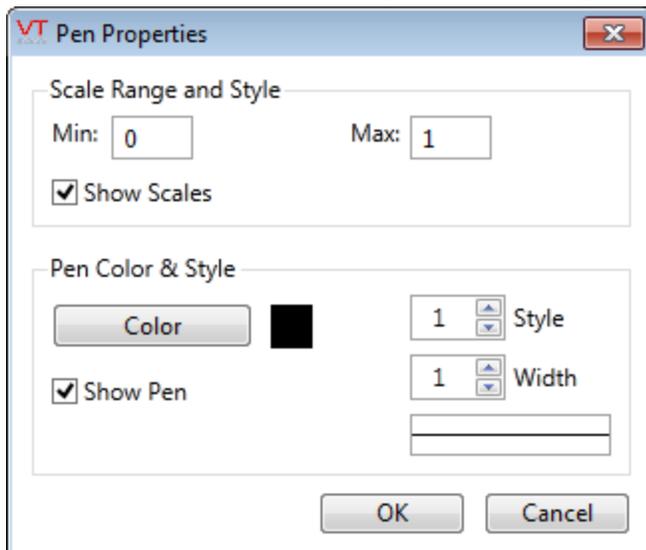
Digital Pen Properties Element	Description

Min & Max	The range of the displayed scale. 0 and 1 for most digitals, but may be set 0 to 3 for two-bit digital inputs.
Show Scales	Select whether the minimum and maximum scale values are to be displayed along the Y-axis of the graph.
Color	Opens the Select Color palette, which you can use to select a new color for the pen.
Style	A spin-box that you can use to change the line style (solid, dashed, etc.). Affects the choice of width, in that you cannot select a wide line for any style other than solid.
Width	A spin-box that you can use to set the width of the pen. Affects the choice of style, in that choosing a wider line forces the line style to be solid.
Show Pen	Hides this tag's pen when not checked.
Average	Affects the legend display. If checked, the average of all plotted values across the displayed time span will be calculated for the tag.

Related Information:

...Configure Analog Pen Properties

...Configure Notebook and Note Properties

Configure Notebook and Note Properties

The appearance of the markers, placed on the graph when you add a note, can be adjusted as follows:

- Double-click the entry, or click the gear icon to open the properties dialog.
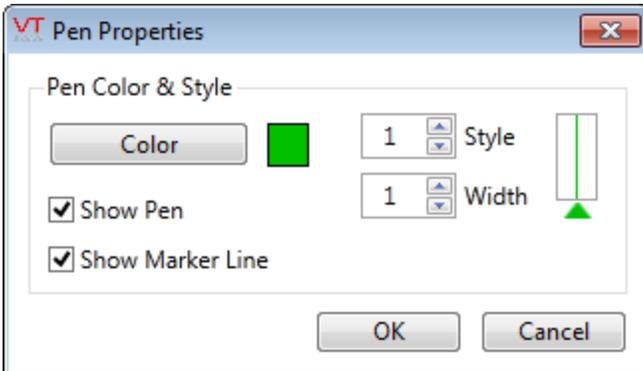
Digital Pen Prop-	**Description**

erties Element	
Show Pen	Hides this tag's pen when not checked.
Show Marker Line	When checked, the note's placement on the graph is shown with both an arrow and a vertical line. Deselect to remove the vertical line.
Color	Opens the Select Color palette, which you can use to select a new color for the pen.
Style	A spin-box that you can use to change the line style (solid, dashed, etc.). Affects the choice of width, in that you cannot select a wide line for any style other than solid.
Width	A spin-box that you can use to set the width of the pen. Affects the choice of style, in that choosing a wider line forces the line style to be solid.

Related Information:

...Configure Analog Pen Properties

...Configure Digital Pen Properties

Fix Pen Color Configuration Problems in a Remote Desktop Session

When connecting to VTScada through a remote desktop, you may find that the colors used for each pen in the legend appear to be the same. In particular, with remote desktop sessions accessing VTScada on a Windows Server 2008™ system, colors in the HDV may appear incorrectly, as the Windows Remote Desktop Session host, in its default setting, will limit remote desktop sessions to 16-bit color.
You can remedy this as follows:

1. From the Windows Start button, click to open "Administrative Tools".
2. Click to open, "Remote Desktop Services".
3. Click to open, "Remote Desktop Session Host Configuration".
4. Under the list of connections, right-click on the connection and select Properties from the menu that opens.
5. Open the Client Settings tab of the properties dialog.

6. Clear the Limit Maximum Color Depth check box.
7. Click OK to save your work and close the dialog.

If a remote session desktop is running, both it and any running VTScada program on the remote machine, must be restarted before the change will go into effect.

Report Configuration

One of the most important jobs of a SCADA system is to generate reports on the data being collected. VTScada provides several reporting tools that you can configure to work the way that you need.

Most reports will be generated using the Reports Page, but this tool requires Operators to perform several steps in order to generate a report. As a VTScada Developer, you can save Operator's time and reduce errors by pre-configuring some of those options. An even better solution is to use a Report tag, which can be fully-configured and has the ability to run automatically on a schedule, in response to an event such as an alarm, or in response to an operator pressing a button.

The Report Button widget (used for drawing Report tags) also provides an option to re-run the last scheduled report, in the event that a power-failure or other error caused the last automatic report to be missed.

You also have the ability to configure new types of report output, adding calculations or visual formatting.

Related Information:

...Report on Collected Data – Operator's Guide instructions for using the Reports page. Includes a description of each report type.

...Save and Load Tag Groups – The Operator's Guide describes how to save time by creating and using named groups of tags.

...Report Tags – Fully pre-configured reports that run automatically or at the press of a button.

...Run-Time Configuration of Report Tags – Allow operators to modify certain settings in a Report tag, without needing to have configuration privileges.

...Report Generation Properties – Settings to control the defined start of week, default output directory, email server and more is described in the VTScada Admin Guide

...Email Setup for Alarms and Reports – Configuration required to send reports by email is described in the VTScada Admin Guide

...Applying an Excel Template to Report Output – Perform calculations on, and formatting of the report output.

...Build Custom Reports – The VTScada Programmer's Guide describes how to create new types of report.
In addition to reports, VTScada provides the following options for viewing and exporting logged data:

- The Historical Data Viewer for viewing data trends.
- VTScada can be configured so that third-party programs can read the log files as if they were an SQL database.
- VTScada can be configured as an OPC Server.

Applying an Excel Template to Report Output

VTScada enables you to send report data to a customized Microsoft Excel™ template file. Excel template files can be used to apply formatting, and to generate summary calculations.

> Note: In order to make use of this feature, you must:
>   1) have Microsoft Excel installed on your PC,
>   2) you must use the Screen Display report output option,
>   3) you must provide the name of the template file that you wish to use.

The VTScada installation comes with two sample template files, named Example1.xltm and Example2.xltm. These templates are stored in the Example directory within the VTScada installation directory (e.g.

C:\VTScada\Example\). *.xlt versions of these files are also provided for customers with versions of Excel older than version 2007.

Within each of these template files is a macro named, Complete. A macro is a mini-program you write or record to store a series of commands that you may run to automate complex or time-consuming tasks. In the case of these template files, the Complete macro sets the visibility of the report to true or visible, and saves the report.

The Complete macro is written using the Visual Basic programming language. In the following code, comment lines start with an apostrophe, '.

```
Sub Complete()

'This is an example of an Excel Template macro that can be used with
VTScada reports.
'At this point the report module has already dumped the report data
onto the spreadsheet.
'Your macro must be called Complete.
'To uncomment line, remove the preceding'

'If the macro makes any changes to the spreadsheet, then the user
will not be able to
'close the workbook until they answer the save query dialog.
'To prevent this, set the Saved property of the workbook to 1
'Excel.Application.Visible = TRUE
'Uncomment the above line if you want your spreadsheet to be dis-
played on report completion
'Leave it commented if you want report execution to take place
silently (suitable for automatic reports)

'ActiveWorkbook.SaveAs Filename:="Output.xls"
'The above line is an example on how to save your file with a hard-
coded filename.
'Default saving location for Excel is My Documents.

'ActiveWorkbook.SaveAs Filename:="C:\" + Replace(Worksheets
("Sheet1").Range("A1").Value + " generated at " + Time$ + " on " +
Date$, ":", "-") + ".xls"
'The above line is an example on how to save the file on your C drive
with the filename being
'the report name and range concatenated with the time at which it was
generated.

End Sub
```

The macro, "Complete" in file, "Example1" contains code to display the spreadsheet when the report is complete. (As noted in the comments, the relevant line of code must be uncommented.) An example of how to save the report spreadsheet with a hard-coded file name is included.

Additionally, an example is provided of how to save and name the report spreadsheet with the report name and range, concatenated with the time at which it was generated.

The contents of the Complete macro in Example2:

```
Sub Complete()
'This is an example of an Excel Template macro that can be used with
VTS reports.
'At this point the report module has already dumped the report data
onto the spreadsheet.
'Your macro must be called Complete.
'To uncomment line remove the preceeding '
'If the macro makes any changes to the spreadsheet, then the user
will not be able to
'close the workbook until they answer the save query dialog.
'To prevent this, set the Saved property of the workbook to 1
'This sample Excel template file automatically determines the extents
of the data sent to it
'& calculates the total, average, standard deviation, and variance at
the end of the report.
'This will work for any number of columns in any number of sheets in
a workbook file provided
' that the title is always 2 rows high and that the time & date each
occupy a column each.

' View the spreadsheet when it opens
Excel.Application.Visible = TRUE

'Save file name is set to current time & date
If Dir("C:\Reports", vbDirectory) = "" Then
  MkDir ("C:\Reports")
End If

If Val(Application.Version) < 12 Then
  ActiveWorkbook.SaveAs Filename:="C:\Reports\" + "VTSCADA Report " &
Replace(Time$ + Date$, ":", "-") + ".xls"
Else
  ActiveWorkbook.SaveAs Filename:="C:\Reports\" + "VTSCADA Report " &
Replace(Time$ + Date$, ":", "-") + ".xlsm", FileFormat:=52
' 52 = xlOpenXMLWorkbookMacroEnabled = xlsm (with macro's in 2007-
2010)
End If

' Trap errors & ignore
On Error GoTo ErrorCheck
' Number of title rows
Trows = 2
' Number of timestamp columns
TCols = 3

' Cycle through all worksheets
For Each Sheet In ActiveWorkbook.Worksheets
  ' Go to the sheet
  Sheets(Sheet.Name).Select
  ' Record the active cell on the sheet
```

```
    ActiveCellPos = ActiveCell.Address
    ' Determine the area used on the worksheet using Range format
    UsedArea = Sheet.UsedRange.Address(ReferenceStyle:=xlA1)
    ' Get the number of rows & columns filled in
    C = Sheet.UsedRange.Columns.Count
    R = Sheet.UsedRange.Rows.Count
    ' Set comments Under Date/Time
    Cells(R + 1, TCols).Value = "Total"
    Cells(R + 2, TCols).Value = "Average"
    Cells(R + 3, TCols).Value = "Standard Deviation"
    Cells(R + 4, TCols).Value = "Variance"
    ' Formulas for each column
    For I = (TCols + 1) To C
      ' Formula for total
      Cells(R + 1, I).Activate
      ActiveCell.FormulaR1C1 = "=SUM(R[-" & R - Trows + 0 & "]C:R[-
1]C)"
      ' Formula for average
      Cells(R + 2, I).Activate
      ActiveCell.FormulaR1C1 = "=AVERAGE(R[-" & R - Trows + 1 & "]C:R[-
2]C)"
      ' Formula for Standard Deviation
      Cells(R + 3, I).Activate
      ActiveCell.FormulaR1C1 = "=STDEV(R[-" & R - Trows + 2 & "]C:R[-
3]C)"
      ' Formula for variance
      Cells(R + 4, I).Activate
      ActiveCell.FormulaR1C1 = "=VAR(R[-" & R - Trows + 3 & "]C:R[-
4]C)"
    Next I

    ' Select the calculated data & set to bold
    Rows(R + 1 & ":" & R + 4).Select
    Selection.Font.Bold = TRUE
    ' Bold the title line & set color
    Rows("1:" & Trows).Select
    Selection.Interior.ColorIndex = 35
    Selection.Font.Bold = TRUE
    ' Adjust column widths to fit data
    Rows("2:" & R + 4).Select
    Selection.Columns.AutoFit
    ' Set to freeze pane on first data cell
    Cells(Trows + 1, 1).Select
    ActiveWindow.FreezePanes = TRUE

    Rows(Trows + 1 & ":" & R).Select
    Selection.Rows.Group TRUE
    ActiveSheet.Outline.ShowLevels RowLevels:=1
Next

' Exit here to prevent running error trap when complete
Exit Sub

' Do nothing error trap
ErrorCheck:
Resume Next
```

```
End Sub
```

The macro, "Complete" in the file, "Example2" calculates the total, average, standard deviation and variance of the report data.

Related Tasks:

...Create an Excel Template

...Use an Excel Template

Create an Excel Template

You must be familiar with using Visual Basic for Applications (VBA) in order to create or modify a macro within Microsoft Excel. The menus to open and therefore the steps to follow within Excel will vary depending on the version you are using. The following procedure is provided as a guideline rather than a set of steps to be followed.

To create a custom Microsoft Excel template file:

1. Navigate to the Example directory within the VTScada installation directory (e.g. C:\VTScada\Example\).

2. Open the template file you wish to use.
   "Example1" or "Example2". The extension will be .XLT or .XLTM, depending on which version of Excel you use.
   Be careful that you do not create a new file based on this macro instead of opening the macro for editing. Right-click on the file name and be sure to select Open rather than the default option, New.

3. If prompted to enable macros, do so.

4. Press the Alt-F11 key combination to open the VBA development environment.

5. Locate Module1 in the Project area.
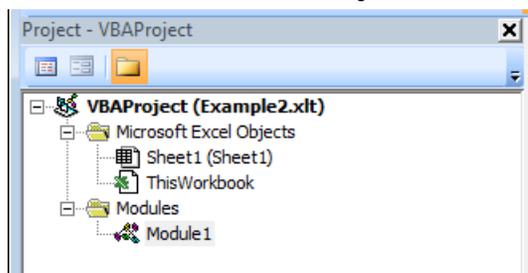
6. Click to open the code window to edit the macro named Complete().

7. Modify the macro as you require.

   A way to do this that requires only a cursory knowledge of VBA pro-gramming is to start with a report that is saved to an unformatted Excel file. Turn on the macro recorder and adjust the report as you like. When the report looks the way you want, stop the macro recorder and examine the code that was created to learn how each step of the formatting process is coded into VBA.

8. Expand the File menu and click, Save.

9. Close the VBA development environment window.

10. In Excel, select Save As from the file menu.

11. Navigate to your application directory (e.g. "C:\VTScada\MyApplication\).

12. Enter a name for the template file in the File name field.

13. Close the custom template file.

If trouble-shooting is required, there are many resources available in bookstores and on-line for both Excel and VBA. *Trihedral does not offer support for either Microsoft Excel or the VBA language.*

Next Steps:

...Use an Excel Template

Use an Excel Template

Once you have created a custom template file, you may use it to format your VTScada reports. The following instructions will help to guide you through this process.

> Note: The use of Microsoft Excel template files is available only to those users who have Microsoft Excel installed on their PC, and who are using the Screen Display report output option.

1. Navigate to the Reports page.

2. Set the report parameters as you require (i.e. select the report type, the tags whose data you wish to be included in the report, and configure a time period for the report).

3. Select Screen Display in the Output Types drop-down list on the Reports page.

4. Select the Use Excel to Display Screen Reports check box. The Template field above this check box becomes enabled.

5. Click the Browse button. The Template File dialog opens, with which you can browse your hard drive for your template file.

6. Browse to your application directory and select your custom template file.

7. Click the Open button. The Template File dialog closes and you are returned to the Reports page where the path to your custom template file is displayed in the Template field.

8. Click the Run Report button.

Note: Excel will not be used for reports viewed over VIC connections. All configuration options that would create a report in Excel will be disabled on the Internet client..

Run-Time Configuration of Report Tags

The Report Tools Folder

The components of the Report Tools folder of the Widgets palette can be used to add flexibility to Report tags. Report tags are extremely useful in that they hold all the configuration required to generate a report. This saves operators time by avoiding the need to work through the steps of the Reports page.

However, it may be useful to provide certain configuration options for a report tag, such as allowing an operator to choose the output destination, or the time span covered in the report. This can be achieved, without granting tag configuration privileges to the operator, by using the elements in the Report Tools folder.

Note: This library duplicates the widgets available by drawing any Report tag. The exception is the Report Iterations widget, since Report tags cannot include iterations as part of their configuration. The Report Iterations widget is included for the sake of completeness as one of the report page tools, and might be used if you were creating a custom report page.

The Report Tools folder can found in the Widgets palette of the Idea Studio. All the components that make up the Reports page are available here. Note that, it is not expected that you would use these to duplicate the Reports page. The usefulness of these tools lies in the ability to provide selected configuration options for a given Report tag.

After dragging each tool from the palette to a page, link it to the Report tag instance that it should modify.

Related Information:

...Report Button Widget – Allow a report to be generated (or re-generated) at any time.

...Reset Target Widget – Choose the output format and location of the report.

...Report Iterations – Repeat the report for consecutive time frames.

...Report Options Widget – Choice of Excel or Access options to enhance selected output formats.

...Report Tag List Widget – Selection of tags to be included instead of the original named group.

...Report Type Widget – Enable selection of the report type.

...Reporting Period Widget – Modify the time period that will be covered by the report.

...Reporting Period (Enhanced) Widget – Provides a greater range of options for controlling the report's time period.

SQL Queries of VTScada Data: The ODBC Server

You can generate reports of VTScada data using Structured Query Language (SQL) queries in third-party programs such as Dream Report™, Crystal Reports™, Microsoft Access or Excel™. These queries are handled by the ODBC Server, an optional component of VTScada that must be purchased with your license agreement.
A significant advantage of the ODBC Server is that it can query VTScada from computers that do not have VTScada installed. Managers, engineers, operators etc. can view up-to-date reports from any location that has network access to your VTScada server.
Using this interface, you can treat a VTScada application as if it were a database containing logged tag values, aggregate tag data, and alarm data. Once the connection is configured, your reporting program can send SQL queries to VTScada to retrieve tag values that are being logged. (Tags that are not being logged will have no stored history to query.)

Requirements:

- A license key that includes the ODBC Server option. Check by opening the VTScada About button in the VAM.
- Install the VTScada ODBC Driver on any computer that is to generate the queries. To do so, run the program VTSODBCDriverInstall.exe. This will install either a 32-bit or 64-bit version of the driver, matching your operating system.
- Secure your application, and grant the Internet Client Access privilege to at least one account.
- A configured realm, in which the application has been selected.
- A system DSN, configured using the Microsoft Windows™ ODBC Administrator program, or knowledge of the configuration so that you can build your own connection string.

Related Information:

...Data Available to the ODBC Interface - Reference

...ODBC Interface: Table Structure and Notes

...SQL Queries: Reference and Examples

Related Tasks:

...Configure Your Application to be an ODBC Server – Server con-figuration and connection

...Monitor the ODBC Interface – Capture in-coming queries as they arrive.

Configure Your Application to be an ODBC Server

To create an ODBC connection to your application, at least one server must be configured to be an ODBC server.
The steps are much the same as those to set up VTScada Internet Server and Realm, with a few small exceptions.
The Max Clients setting has no effect in this context. This setting is rel-evant to VIC and MIC connections, rather than ODBC clients.

1. In the VTScada Application Manager (VAM), select the button labeled Internet Setup. The VTScada Internet Client/Server Setup dialog will open.
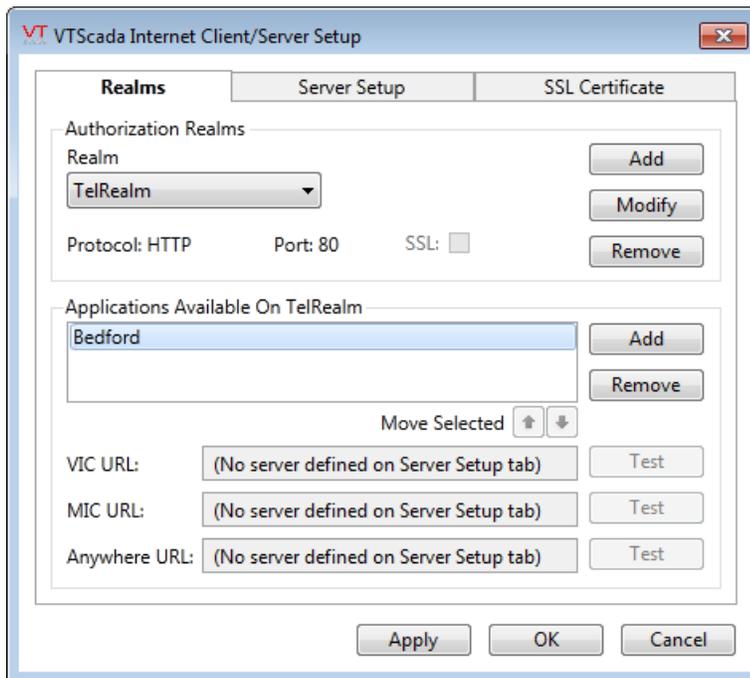
2. Select the Add button to open the Add Realm window where you will provide a name and a connection protocol for your server.
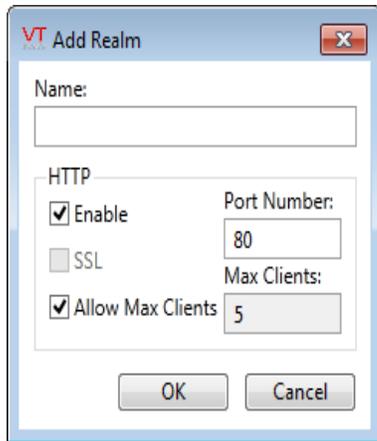
3. Provide a name

    The name should clearly indicate what the server is. As a suggestion: by
    including the letters "ODBC" you will make the name easier to recognize later
    when configuring the ODBC connection.

4. Configure the Protocol and Port Number

    For the purposes of the ODBC Interface you must enable the HTTP protocol.
    The Port Number will normally be 80, the standard HTTP port, unless your
    network administrator directs otherwise.

5. Check the SSL option, if using.

6. Click, "OK" to save the new Realm

7. Click the second Add button in the Internet Client Window in order to select
    which application will be available to this server.
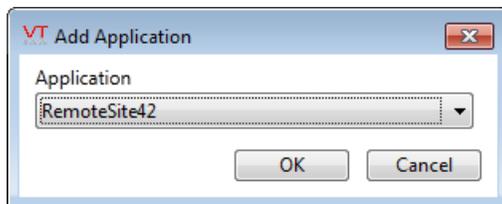
8. For the application selected, add the following to your application properties,
    in the System section.
    Use the name you created for your own realm here. You may use the Applic-
    ation Properties dialog or edit the Settings.Dynamic file. Multiple realm
    names may be included, separated by semicolons.

```
SOAPServicesRealmName    = Your_ODBC_Realm_Name_Here
```

Note: You will need to re-start the application after adding this prop-
erty in order for the changes to take effect.

Optional Step:

You can take advantage of VTScada's ability to group data records by time interval. By adding the application property, SQLQueryTableTPPs, you can retrieve tag data from specific time intervals. The format of the statement is:

```
SQLQueryTableTPPs    =    <time frame specifier>
```

where the time frame specifier takes the form of a digit and a letter. The letter indicates the units in which time interval is measured (hours: h , weeks: w, etc.) and the digit specifies the number of units of each interval that should pass between each record.

Two separate examples follow:

(Note that you can have only one TPP statement in a config file, but that statement can have multiple values, separated by semi-colons and no spaces).

```
; two hours...
SQLQueryTableTPPs    =    2H
```

...or, using the Add Property dialog and configuring two TPP values:
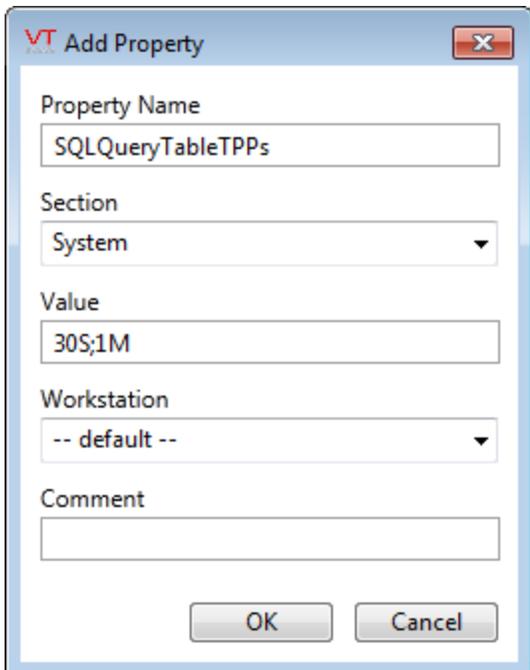
The available time interval units are:

- MS – milliseconds
- S – seconds (this is the default; the S may be omitted)
- M – minutes
- H – hours
- D – days
- W – weeks
- Y – years

Connect to the ODBC Server

Any computer that has a network connection to the ODBC Server may con-
nect. There is no need for VTScada to be installed on that computer, but
the Trihedral-supplied ODBC Driver must be.
Connection parameters must be supplied to the program that will run
the query. This can be done using a data source name (DSN), configured
using the Microsoft ODBC Data Source Administrator, or you may use an
ODBC connection string.

1. On each computer that is to connect to the server, run VTSODBCDriver-
   Install.exe.
   This will install the ODBC driver needed to connect to the server.

If using an ODBC connection string, it will take the generalized form:

```
DRIVER=VTS ODBC Driver;SERVER=Your_Server_Address;REALM=Your_
Realm;UID=Account_Name;PWD=Password
```

If using a DSN, open the Microsoft ODBC Administrator utility on your
computer. There will be two such utilities: one for 32-bit connections
and one for 64-bit connections. Choose based on the program you will
be using to generate the queries.

- The 64-bit ODBC Administrator utility can be found in the Administrative
  Tools section of your Control Panel.
- The 32-bit ODBC Administrator utility can be opened using, \Win-
  dows\SysWOW64\odbcad32.exe.

Open the System DSN tab of the Administrator utility and create a new
connection, selecting the VTS ODBC Driver.

Examples:

- A configured DSN:
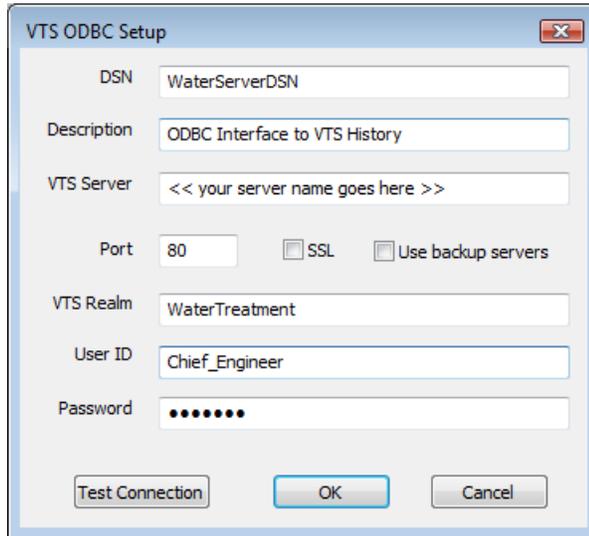
- Connection string for a secured application:

```
DRIVER=VTS ODBC Driver;SERVER=127.0.0.1;REALM=[RealmName];UID=[User-name];PWD=[Password]
```

- Connection string for a realm configured to use a port other than the default 80, or to use SSL:

```
DRIVER=VTS ODBC Driver;SERVER=127.0.0.1;REALM=[RealmName];UID=[User-name];PWD=[Password];PORT=443;SSL=YES
```

Notes:

- The DSN name you enter should be as descriptive as possible. Be aware that long names might not save properly.

Note: Do not use the name of your VTScada application as the DSN name.

- For the VTScada Server field, enter the fully qualified domain name of the machine running the VTScada application.
  If response times are excessively slow when using a workstation name, you may obtain significantly better performance by the server's IP address instead.
- The Port should match what was configured for the Realm.
- Check the SSL box only if the Realm has been configured for SSL.

- Selecting 'Use Backup Servers' causes a list of VTScada Internet Servers to be retrieved and stored, from the VTScada ODBC server any time a successful connection is made. If future attempts to connect to this server fail, the list of VTScada Internet Servers will be tried one-by-one for a connection

Note: The VTScada application must be secured with a user name having internet privileges. Enter that user name and password here.

- The Test Connection button should succeed if the application is running.

Troubleshooting:

- No connection.

  Confirm that all the following are in place:
    - Server and Realm configuration completed on the VTS server.
    - SoapServicesRealmName line added to the server's Settings.Dynamic file and file changes have been imported.
    - Security is active and the user account has the VTS Internet Client privilege.
    - The querying workstation is able to connect to the server.
    - If using a DSN, you are using 32-bit or 64-bit as required by your querying program.
    - The connection string or DSN uses the correct names and addresses as configured on the Server.

Next Steps:

...SQL Queries: Reference and Examples – Create and run queries.

...Monitor the ODBC Interface – Monitor the in-coming queries.

Data Available to the ODBC Interface

The ODBC interface exposes the following information for you to query:

- Alarms - You can inquire as to the current status and the configuration of all alarms in your system
- Alarm History - Provides a record of all alarm events at all priorities
- Notes - All notes by notebook.

- Logged Tags – All data collected from tags that have loggers attached. (This includes Status and Control tags since these are automatically logged.)
- Values from Logged Tags, grouped by time period.
  Requires a value for the setting, "SQLQueryTableTPPs".

Time stamps use UTC values in the tables. Your queries will need to take this into account. For queries against the table, History, you can query two available time stamp values:

- Timestamp – The time reported by the I/O device, along with the value. If a time stamp is not provided with the value, then the time at which the value was read or calculated will be used.
  This value is indexed, and should be used in all WHERE clauses.
- StorageTimestamp – The time when the value was written by the Historian. This value is not indexed and should not be used in a WHERE clause. Doing so may result in a very slow query response time. StorageTimestamp is not available to be queried in the derivative, data-aggregated _TPP tables.

Legacy tables stored tag values in separate tables and used the time stamp of the current server. These legacy tables still exist but are hidden by default, using the property: SQLQueryHideLegacyTables. Newer code all tag values from the table, History or a "_TPP"derivative such as "History_1d"

Related Information:

...ODBC Interface: Table Structure and Notes - Reference

...Configure Your Application to be an ODBC Server - Includes notes for SQLQueryTableTPPs.

ODBC Interface: Table Structure and Notes

The following tables will be relevant for your queries.

Note: If a tag is not logged, its history cannot be queried.

Prior to release 11.1 of VTScada, queries of logged tag data would refer to a separate table for each tag. These legacy tables may still be viewed if the application property SQLQueryHideLegacyTables is set to 0 (FALSE). :Alarms and :AlarmHistory are also legacy tables, the only difference

between these and "Alarms", "AlarmHistory" being that timestamps used the server's time zone instead of a UTC value.

Note: The current logging system records all data using UTC time stamps. Legacy tables, noted above, use the server's time zone. You must take the time zone of the data into account when designing your queries.

Table name – History

- Contains time stamps, values from all logged tags, notes recorded in all notebook tags and recorded statistics from each driver tag in your application.
- Other than the column Timestamp, the column names will vary depending on the tags in your application.
- Column names will be the full name of each tag, with the name of the logged property appended. For most I/O tags and notebooks, this suffix will be ":value".
- The timestamp value will always be stored using UTC[1].
- If your query includes two or more tag names, the result set will be sparse if viewed as a table. For any given timestamp, it is likely that there will be only one logged tag value. Other tag values are marked as "null" for that timestamp. In this case, there is no way to distinguish between values that were not logged at a given time, and values that were logged as null.
- You are strongly advised to include a time range in the where clause of every query. The BETWEEN clause is recommended.

Examples of possible column names for the table History:
- Timestamp
- Primary TCP Port\Modbus1:ErrorValue
- Primary TCP Port\Modbus1:FailedCount
- Primary TCP Port\Modbus1\Primary Pump\Flow Rate:value
- Primary TCP Port\Modbus1\Secondary Pump\Flow Rate:value
- Operator Notes:value
- System Notes:value

---

[1]Coordinated Universal Time

Multiple Values per Time Stamp

In most situations, if there are multiple values from one tag with the same time stamp, it is because duplicates are being recorded. VTScada will store all the values (each with its own sequence ID), but a query will return only one.

There are situations where different events can be logged to the same tag with an identical timestamp as provided by the I/O device. If this matches your situation, you can tell VTScada to return all the values for that timestamp by adding the property, SQLQueryHistoryNoOverridesTableName to your list of application properties and setting the value to a name you wish to use for the table in your queries. For example:

Note: if using this option, it will usually not make sense to query TPP values other than minimums or maximums. Add the property SQLQueryHistoryNoOverridesTableSupportsTPP as well, setting the value to FALSE in order to explicitly deny TTP value queries when retrieving duplicates.

Table name - History_TPPvalue

Values that appear to come from a History_TPP table are calculated in response to your queries.

Use care to avoid querying a time span that extends into the future. The last known value will be projected for all time periods after the current date.

Four default values for TPP are specified in the application property SQLQueryTableTPPs, but this is a legacy setting. Your query can be for any TPP value. For example, you may query from the History_15m even though it is not specified in the SQLQueryTableTPPs property.

Example table names: History_1M,   History_1H

For each logged tag in the application, and for the recorded statistics of every driver, this table will contain a column named after the tag and having a suffix identifying the statistic calculated for each time period.

Column name showing suffix	Data type	Description
Timestamp	Date/Time	UTC date and time at beginning of each time span
Tag Name:Value:Average	Double	Average value within each time span
Tag Name:Value:Minimum	Double	Minimum value recorded in the time span
Tag Name:Value:Maximum	Double	Maximum value recorded in the time span
Tag Name:Value:Delta	Double	Change in value over the range
Tag Name:Value:ValueAtStart	Double	Value at the beginning of the time span
Tag Name:Value:TimeOfMin	Date/Time	Time of the minimum value recorded in the range
Tag Name:Value:TimeOfMax	Date/Time	Time of the maximum value recorded in the range
Tag Name:Value:ZToNZCount	Long	Count of zero to non-zero transitions during the span
Tag Name:Value:NonZeroTime	Double	Total time within the span that the value is not zero or invalid
Tag Name:Value:Total	Double	Arithmetic sum of the recorded values

		in the time span
Tag Name:Value:Interpolated	Double	A value interpolated across a range of time spans.

Columns in the History and History TPP tables that are related to drivers

For a complete reference to the following columns, please refer to Communication Driver Log-Enabled Variables.

For History TPP tables, the statistic suffixes will be added to these

Column Name	Data Type	Description
Tag Name:Timestamp	Date/Time	UTC date and time of last update
Tag Name:ErrorValue	Integer	An error value or code associated with a driver's communication error
Tag Name:FailedCount	Integer	Incremented on each communication error.
Tag Name:FailedRetryCount	Integer	Incremented if failure occurs on a retry
Tag Name:SuccessCount	Integer	Count for successful reads and writes
Tag Name:Quality	Double	Used to show the driver's overall "health"
Tag Name:ResponseTime	Double	The time it takes the PLC/RTU to receive a command, process it and send a response
Tag Name:ErrorAddress	Text	If an address was associated with the error, it will be recorded here

Table name - Alarms

Contains the current status of all configured alarms. There is no Timestamp column for this table, as all information is current. Note the leading colon in the table name.

Column name	Data Type	Description
ID	Text	Unique ID of the alarm
Name	Text	Name of the alarm
SubName	Text	Sub-name of the alarm (e.g. LoAlarm, HiAlarm)
Status	Text	Text description of alarm status
Active	Integer	0 for inactive, 1 for active

Unacked	Integer	0 if unacknowledged, 1 if acknowledged
Disabled	Integer	0 if enabled, 1 if disabled
Timestamp	Date/Time	Time of the last event for this alarm. May be null
Message	Text	Message associated with last event
Priority	Text	Priority attached to alarm (text, not numeric)
Type	Integer	May either denote whether the alarm was reported, or may be used to differentiate between different alarm types.
HookPointValue	Double	Numeric value of tag triggering the alarm
Area	Text	The area the alarm belongs to
HookPointUnits	Text	The engineering units matching the HookPointValue
Operator	Text	The operator logged in at the time of the last event.

Table Name – AlarmHistory

Contains a record of all alarms that have occurred in the application, including event alarms for operator actions.

Column Name	Data Type	Description
Timestamp	Date/Time	UTC date and time of each alarm event
Name	Text	Name of the alarm
SubName	Text	Sub-name of the alarm (e.g. HiAlarm, LoAlarm)
Event	Text	Text description of the alarm event. (e.g. Enabled, Acknowledged, Cleared, Event, etc.)
Disabled	Integer	1 or 0 according to the alarm disabled status
Message	Text	Message associated with the event
PriorityNum	Integer	Numeric value of the alarm priority.
Priority	Text	Priority of the alarm (as text, not numeric)
Type	Integer	Obsolete
HookPointValue	Double	The value of the tag triggering the alarm at the time of the event.
Area	Text	The area the alarm belongs to.
HookPointUnits	Text	Engineering units associated with the HookPointValue.

Operator	Text	The operator logged in at the time of the event.

Related Information:

...Query a 3rd-Party DBMS

Refer to the VTScada Admin Guide for the following properties:

...SQLQueryHideLegacyTables

...SQLQueryTableTPPs

SQL Queries: Reference and Examples

Limit on result set size:

The size of the result set is limited by the application property, SQLQueryMaxResultRows. This variable defaults to 10,000.

If you are changing SQLQueryMaxResultRows to be higher than its default value and are using a WHERE clause that filters on something other than time ranges, you may also need to increase SQLQueryMaxResultRowsPreFilter. This specifies the maximum number of records that can exist in memory before the WHERE filter is applied. SQLQueryMaxResultRowsPreFilter defaults to 1,000,000.

Note: You are strongly advised to include a time range having both a start date and end date in every query. Doing otherwise may return an extremely large data set.

All time stamp values use UTC. You will need to adjust the date and time values in your query for your time zone and possibly for daylight savings time.

History_TPP tables are treated as real tables, but are actually calculated on the fly in response to a query. When querying the History_TPP table, use care not to set the end time of the query in the future. The last known value will be carried forward.

Supported SQL syntax follows the format:

 SELECT [DISTINCT | ALL] 'columnspecifier-1', 'columnspecifier-2', ...
 FROM 'tablename-1', 'tablename-2'
 [WHERE where-expression]
 [ORDER BY 'columnspecifier-1' [ASC | DESC], 'columnspecifier-2'...]
[LIMIT [offset,] row_count]

For the above ...

- Where the column specifier refers to a tag's value, it must include the full tag name with the suffix, ":value".

- In the alarm tables and the TPP tables, you may use a * to indicate all columns, but you cannot use it as a wildcard to select columns having a common root.

- Column name aliases may be used in the SELECT clause to make the result more readable, but they are not supported in the WHERE clause. All column names in WHERE clauses must be fully specified.

- The table name is either the table name or 'table-name AS alias-name'. Quotes may be used around table or column names, and must be used if the names contain special characters such as a space or colon.

- When a table-alias name is specified it may be used in place of the table-name in the column specifier. Table names are only required in the column specifier when querying multiple tables.

- The where-expression filters the data that will be retrieved by the query. Refer to the following table for functions and operators that may be used in the WHERE expression.

- Use the limit clause to restrict the number of rows that will be returned, and optionally add an offset (0-based), which allows pagination. LIMIT is applied after WHERE filtering and after sorting. If there is no explicit LIMIT clause in the query there is an implicit "LIMIT SQLQueryMaxResultRows" (Settings.Dynamic setting) added.

When more than one table is specified, the tables are automatically joined based on their Timestamp columns and *must* have matching time periods.

A JOIN expression may be used in place of a table list, but it is only parsed to extract the tables specified; the actual join expression is ignored.

Functions, Operators and Expressions

Only WHERE clauses may use functions, operators and expressions. In the SELECT clause, mathematic operators will be ignored and functions will return a "column not found" error.

Remember that the History table permits queries on only one column in addition to the Timestamp column.

Entries in the following table are grouped in order of decreasing precedence, where entries in each group have the same level.

Operator, function or keyword	Description
First Level	
()	Use parenthesis to enclose the parameters of a function and to group expressions, thereby controlling the precedence of operations.
Constant (including sign)	Any constant value to which you are comparing a field.
Column Reference	Tag name and logged property. WHERE 'Tank1Level:Value' > 10
Second Level	
ABS	Absolute value function.
CASE	The CASE statement is a standard SQL command, but in VTScada only the following form is supported: CASE WHEN condition THEN result1 ELSE result2 END
CONCAT	String concatenation. Limited to two parameters. WHERE CONCAT(Area, Operator) = "Western zoneFred"
INTEGER	Function to truncate a numeric value to the integer portion. WHERE INTEGER('TankLevel:Average:Value' / 'TankVolume:Average:Value') > 50
LENGTH	Function to count the number of characters in a field. WHERE LENGTH(HookPointUnits) > 5

LOWER	Function to convert a string value to lowercase letters. WHERE LOWER(Area) = "water tower"
SQRT	Function to calculate the square root.
UPPER	Function to convert a string value to uppercase letters. WHERE UPPER(Area) = "WATER TOWER"
Unary + and −	Use the positive or negative sign before the value that is to be incremented or decremented by one.
Third Level	
*	Multiplication operator.
/	Division operator.
%	Modulus division operator.
Fourth Level	
+ and −	Regular addition and subtraction operators.
Fifth Level of Precedence	
>	Greater than comparison operator.
>=	Greater than or equal to comparison operator.
<	Less than comparison operator.
<=	Less than or equal to comparison operator.
BETWEEN ... AND ...	Example: WHERE x BETWEEN y AND z. Functionally equivalent to WHERE x >= y and x < z
=	Equivalency comparison operator. Note that any comparison to NULL will return NULL, or false. Use IS NULL or IS NOT NULL instead.
!= and <>	Not equal comparison operator.
IS NULL	Test for a null value. WHERE 'Tagname:value' IS NULL
IS NOT NULL	Test for a value that is not null. WHERE 'Tagname:value' IS NOT NULL
Sixth Level	
NOT	Negation clause for comparisons.
Seventh Level	

AND	Both clauses must be true in order for a true response to be true.
Eighth Level	
OR	Either clause must be true in order for a true response.

Related Information:

Data Available to the ODBC Interface

ODBC Interface: Table Structure and Notes

Related Tasks:

Configure Your Application to be an ODBC Server

Monitor the ODBC Interface

Note: Remember that Timestamp in the History and History_TPP tables is stored using UTC. You will need to adjust the data and time in your query for your time zone and possibly for daylight savings time.
In alarm and in legacy tables, the timestamp column uses the server's local time zone.

Example 1:

To retrieve error values from a driver called ModDriver1 over a particular one-minute period, which has been adjusted for UTC time.

```
SELECT Timestamp, 'ModDriver1:ErrorValue' from History
WHERE Timestamp BETWEEN '2014-03-01 16:54:00' AND '2014-03-01
16:55:00'
```

Example 2:

To retrieve several driver statistics from a driver named ModDriver1, over a day, using a TPP of 1 minute: This query is being run in the Eastern Standard time zone (GMT – 5), during standard time.

```
SELECT Timestamp, 'ModDriver1:ErrorValue:Average', 'ModDriver-
1:ErrorValue:ValueAtStart,
'ModDriver1:Quality:Average, 'ModDriver1:Quality:ValueAtStart' from
History_1M
WHERE Timestamp BETWEEN '2014-03-01 05:00:00' AND '2014-03-02
05:00:00'
```

Example 2:

To retrieve weekly minimum values from two Analog Input tags called ai1 and ai2 over the course of November 2013, sorted with most recent values first: (Given a SQLQueryTableTPPs property that includes the 1W flag.) This query is being run in Greenwich, England.

```
SELECT Timestamp,
       'ai1:Value:Minimum' AS ai1Min,
       "ai2:Value:Minimum' AS ai2Min FROM History_1W
WHERE  Timestamp BETWEEN '2013-11-01' AND '2013-11-30'
ORDER BY Timestamp DESC
```

This will return a result set with columns Timestamp, ai1Min, and ai2Min.

Example 3:

To select all the values recoded by a tag named Tank1Level for one hour on April 1 2008.

```
SELECT Timestamp, Tank1Level:Value
FROM History
WHERE Timestamp BETWEEN '2008-04-01 13:00:00' AND '2008-04-01
14:00:00';
```

or

```
SELECT Timestamp, Tank1Level:StorageTimeStamp, Tank1Level:Value
FROM History
WHERE Timestamp BETWEEN '2008-04-01 13:00:00' AND '2008-04-01
14:00:00';
```

Example 4:

To retrieve a list of all active alarms:

```
SELECT Name, Priority
FROM   Alarms
WHERE  Active = 1
```

Example 5:

To retrieve the names of all disabled alarms in the area "MyArea":

```
SELECT  Name
FROM    Alarms
WHERE   Disabled = 1 and Area = 'MyArea'
```

Example 6:

To retrieve all events associated with the operator, Bob, on April 10, 2008. This query is being run in the Eastern Standard time zone (GMT – 5), during standard time.

```
SELECT  Timestamp, Name, SubName, Event
FROM    AlarmHistory
WHERE   Timestamp >= '2008-04-11 5:00:00' AND Timestamp < '2008-04-12
5:00:00'
        AND Operator = "Bob"
```

Example: Java to ODBC Connection

You must have release 1.3 or later of the VTScada ODBC driver in order to connect from a Java® program.

The following Java code sample assumes that ODBC access has been configured on the same workstation, as described elsewhere in this chapter, that your application is available via a realm named "SQLServicesRealm", and that your application has a logged Analog Input tag named "ai1".

```java
import java.sql.*;
public class VTScadaJdbcDemo {
  public static void main(String args[]) throws Exception {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection con = DriverManager.getConnection
("jdbc:odbc:DRIVER=VTSODBC Driver;SERVER=127.0.0.1;PORT=80;REALM=SQLS-
ervicesRealm;");
    Statement stmt = con.createStatement();

    System.out.println("");
    System.out.println("All configured alarms:");
    ResultSet rs = stmt.executeQuery("SELECT * From :Alarms");
    while (rs.next()) {
      String name = rs.getString("Name");
      String pri = rs.getString("Priority");
      System.out.println(name + ": " + pri);
    }

    System.out.println("");
    System.out.println("All notes from SystemNotes:");
    ResultSet rs2 = stmt.executeQuery("SELECT * From SystemNotes");
    while (rs2.next()) {
      Timestamp time = rs2.getTimestamp("Timestamp");
      String value = rs2.getString("Value");
      System.out.println(time + ": " + value);
    }

    System.out.println("");
    System.out.println("All Security events from alarm history:");
    ResultSet rs3 = stmt.executeQuery("SELECT * From :AlarmHistory
WHERE Area=\"Security\"");
    while (rs3.next()) {
```

```
        Timestamp time = rs3.getTimestamp("Timestamp");
        String message = rs3.getString("Message");
        System.out.println(time + ": " + message);
    }

    System.out.println("");
    System.out.println("Weekly interpolated values for ai1 for Novem-
ber 2006:");
    ResultSet rs4 = stmt.executeQuery("SELECT * FROM ai1:1W where
Timestamp > '2006-11-01 0:00:00' and   Timestamp < '2006-12-01
0:00:00'");
    while (rs4.next()) {
        Timestamp time = rs4.getTimestamp("Timestamp");
        Double message = rs4.getDouble("Value:Interpolated");
        System.out.println(time + ": " + message);
    }

    con.close();
  }
}
```

Monitor the ODBC Interface

Through the Trace Viewer application, you can monitor incoming SQL queries. The Trace Viewer is a diagnostic application that ships with VTScada and can be found in the list of applications loaded in the VTScada Application Manager (VAM). Note that you may need to add this as an existing application in order to see it in the applications list.

In brief, to monitor SQL queries from the ODBC interface:

1. From the VAM, start the Trace Viewer application.

2. Click on the lighting bolt button in the upper left corner.

3. Click on the View option beside SOAP Message Tracing in the Live Data Capture Selection dialog that opened in the previous step.

4. Close the Live Data Capture Selection dialog.

Related Information:

...Trace Viewer Application – Instructions and reference provided in the VTScada Programmer's Guide

Filtering Tags, Alarms and Realms

Using a filtering protocol defined in configuration files, you can prevent the viewing of certain information by unauthorized users or set rules that control which tags will load depending on the workstation where the application is running.

There are three types of filtering that can be applied to tags. In all cases, filtering rules involve a tag's area property.

Tag Area Filtering:

Prevent tags belonging to one or more specified areas from loading on a given workstation when your application runs. This type of filtering is generally used in applications where memory is limited, and certain workstations do not require access to all of your application's tags (e.g. in a large plant where a certain plant section is beyond the responsibility of operators in other sections). Tag area filtering is typically applied to individual workstations.

Note: Tag Area Filtering is an older feature, and can be used only with areas that are a single word. Spaces and wildcards cannot be used in this type of filter.

Alarm Area Filtering:

Display only the alarm tags belonging to one or more specified areas on a given workstation. This type of filtering is generally used in applications where certain users do not need to be aware of certain types of alarms on their workstation (e.g. in a large plant where alarms belonging to equipment in a certain plant section is beyond the responsibility of operators in other sections). Alarm area filtering is typically applied to individual workstations.

Realm Area Filtering:

Limit the display of alarm tags belonging to one or more specified areas for operators who belong to a given security userrealm. Additionally, (if those operators have the configuration privilege, necessary to open the tag browser) they will not see the tags having the specified areas, listed in the tag browser. Note that they will still see all tags drawn on any page.
Unlike tag area filtering and alarm area filtering, realm area filtering "follows" the user from workstation to workstation, as it is linked to the user's security account instead of to an individual workstation.

Related Information:

...Alarm Area Filtering – Hide alarms that have been configured with one or more specified areas on a given workstation.

...Realm Area Filtering – Prevent tag selection by unauthorized users.

...Tag Area Filtering – Prevent tags that have been configured with a specified area from loading on a given workstation.

Tag Area Filtering

Tag area filtering will prevent tags that have been configured with a specified area from loading on a given workstation when the application runs.

Note: Tag Area Filtering is an older feature, and can be used only with areas that are a single word. Spaces and wildcards cannot be used in this type of filter.

Why Use Tag Area Filtering?

Tag area filtering can be used to reduce the demand for memory on a given computer. It is particularly useful on older PCs with limited memory, or on PCs where tag licensing is restricted.

How Does Tag Area Filtering Differ From Alarm Area Filtering and Realm Area Filtering?

Tag area filtering prevents tags configured with specific areas from loading on a given workstation. Alarm area filtering hides alarms that have been configured with specific areas on the Alarm page on a given workstation. Realm area restricts the display of alarms based on the tag area and the security-realm membership of the operator.

> Note: Tag area filtering does not affect the data presented on the Alarm page Users will be able to view alarm data corresponding to the unloaded tags in the History and Configured lists, and the filtered areas will still be visible in the Area Filtering drop-down list. If you wish to hide these areas from view on the Alarm page, you must use alarm area filtering. See: Alarm Area Filtering.

Where is Tag Area Filtering Configured?

Tag area filtering is stored in Workstation.Startup or Workstation.Dynamic files, where "Workstation" is the name of the computer on which the properties will apply.

The preferred method of configuring these is to use the Application Properties dialog. This will ensure that your configuration options are stored to the correct location.

How is Tag Area Filtering Configured?

The following elements are involved in tag area filtering:

- A Workstation.Dynamic file named for the PC to which it should apply (e.g. MyPC.Startup),
  This is generated automatically when you provide a workstation name for any of the following properties when working in the Edit Properties page of

the Application Configuration dialog. It will be located in the Work-stationSettings sub-folder of your application.

- The AreaFilter property must be added and set to 1.
- The AreaExclude property should be added and may be set to 1 or 0 define default behavior for areas not explicitly listed.
- The [Areas] section and subsequent area declarations define which areas are to be included and which to be excluded.

Note: You cannot add declarations for areas that contain a space. You cannot use wild cards in area names.

All of these tasks should be done using the advanced mode of the Edit Properties page in your Application Configuration dialog. Sections are created and used when you set the section name in a property. The properties here will be Area names, with values set to 1 or 0 to include or exclude them. Areas not listed will be included or excluded according to the value of the property AreaExclude.

AreaFilter Property

The AreaFilter property enables (or disables) tag area filtering. A copy of the AreaFilter property must be created for the workstation, and should be set to 1 to enable tag are filtering at that station.

AreaExclude Property

The AreaExclude property defines what will be done with any areas not explicitly referenced in your Workstation.Startup file.

- If set to 1, then any area not explicitly included, will be excluded.
- If set to 0, then areas not explicitly excluded will be loaded at startup.

Note: Caution: If using AreaExclude, be careful that system areas are included in the list of tag areas to load, otherwise you may find that ports, drivers, fonts, etc. will not be loaded, thereby preventing the application from running properly on the workstation.

[AREAS] Section

The [AREAS] section is the heading under which you may specify which tag areas to include or exclude from loading at start up. You will need to add each area as a property.

- To load tags configured with a specific area when the application runs, set the name of the property to the area, and the value to 1.
- To prevent tags configured with a specific area from loading when the application runs, set the name of the property to the area, and the value to 0.
- Tags in areas not specifically listed are loaded or not depending on the AreaExclude setting.

Note: Wildcard characters and spaces are not permitted in area names when applying tag area filtering.

Once you have completed the configuration click Apply and restart your application. (Workstation.Startup files are only read when the application initially runs.)

Related Information:

...Tag Area Filtering Example

...Alarm Area Filtering – Hide alarms that have been configured with one or more specified areas on a given workstation.

...Realm Area Filtering – Prevent tag selection by unauthorized users.

Tag Area Filtering Example

In this example, a large plant has two main areas (AreaA and AreaB). Each area employs a group of skilled personnel that operate and monitor only the equipment processes in their own area. Because each area has

thousands of tags, memory is limited on the plant workstations there-fore, the plant management has requested that the workstations in each area load and display only those tags that correspond to the equipment in that area. The one exception is the plant manager who, because he must oversee the entire plant, wishes to view all tags for the application on the main server in his office. I

Given workstations:

- MainControlStation
- StationA
- StationB

Remember, the application must be stopped and restarted on the work-stations in order for these properties to go into effect, even thought the Restart column will not show a check mark.

MainControlStation Configuration

Because the workstation named MainControlStation must load all tags for the application so that the plant manager has full access, it requires no special configuration at this time.

StationA Configuration

The following configuration is required for the workstation named Sta-tionA. Note the values for Workstation and Section in each of the fol-lowing properties.

Property Name	Section	Value	Workstation	Restart	OEM
AreaA	Areas	1	StationA		
System	Areas	1	StationA		
AreaFilter	System	1	StationA		
AreaExclude	System	1	StationA		

The AreaFilter property in the section, System, is set to 1 so that tag fil-tering is enabled this workstation.

The AreaExclude property is set to 1 so that any areas not specified will not be loaded.

The two properties in section, "Areas", have names matching the areas that will be loaded (System and AreaA), and values of 1 to specify that these areas are to be loaded.

StationB Configuration

The workstation named StationB requires the following configuration. Note that these properties are added using the Application Configuration dialog, just as the properties for StationA were. You can save time by copying existing properties, changing values of the fields as required. What matters most is the value in the Workstation field.

Property Name	Section	Value	Workstation▽	Restart	OEM
System	Areas	1	StationB		
AreaFilter	System	1	StationB		
AreaExclude	System	1	StationB		
AreaB	Areas	1	StationB		
System	Areas	1	StationA		
AreaFilter	System	1	StationA		
AreaExclude	System	1	StationA		
AreaA	Areas	1	StationA		

The result of the above configuration is that the workstation named StationB will only load tags whose Area property has been configured as "AreaB" and the "System" area tags .

Alarm Area Filtering

Alarm area filtering will show alarms that have been configured with one or more specified areas on a given workstation, and hide those associated with areas that are not included.

Why Use Alarm Area Filtering?

Alarm area filtering is typically used in applications where some users do not need to be aware of some alarms (e.g. in a large plant where alarms belonging to equipment in a certain plant section is beyond the responsibility of the operators in other sections).

How Does Alarm Area Filtering Differ From Tag Area Filtering and Realm Area Filtering?

Alarm area filtering hides alarms associated with specific areas from a given workstation. The alarm tags will still be visible to developers who have access to the Tag Browser and other application development tools.

Where is Alarm Area Filtering Configured?

Alarm area filtering is best achieved using workstation-specific application properties.

How do I Configure Alarm Area Filtering?

The following elements are involved in alarm area filtering:

- A workstation.startup file named for the PC to which it should apply (e.g. MyPC.startup), and
- The <Alarm_Manager> section and subsequent area declarations.

Workstation.Startup File

Create a workstation.startup file for each PC for which you wish to apply tag area filtering. To create a workstation.startup file:

1. Create a text file that is named for the PC to which its settings should apply, and which has the .STARTUP extension (e.g. MyPC.startup).
2. Save this text file to the WorkstationSettings folder of your application and use either the File Manifest or Import File Edits to make it part of the application's working set of files. (You may need to create the WorkstationSettings folder)

Note: A PC will only obey the workstation.startup file that carries its name.

Once you have created your workstation.startup file, configure it for alarm area filtering as follows:

<Alarm_Manager> Section

The <Alarm_Manager> section is the Workstation.Startup section heading under which you may specify which alarm areas should be visible when the application is run.

Note the use of angle brackets rather than the more common square brackets for this section. These are required.

The following instructions will guide you through the process of successfully configuring the <Alarm_Manager> section.

1. Create or open a WorkstationName.startup file, where WorkstationName is the name of the computer for which the alarm list should be filtered.

2. Save this file into the WorkstationSettings folder of your application. (Create that folder if it does not exist.)

3. Within the file, add a section named <Alarm_Manager>.

4. In this section, enter AlarmArea followed by an equals sign, followed by the name of the alarm area to be displayed (as in the following example).

Note: Alarm area filtering allows the asterisk wildcard character. For example, to view all alarms whose area ends with "1" you could enter *1 for the AlarmArea property. To view all alarms whose area begins with "a" you could enter a* for the AlarmArea property.

```
<Alarm_Manager>
; The alarm areas to show
AlarmArea = North
AlarmArea = South
```

Note: Once you have completed the configuration of your Workstation.Startup file(s), save the files, import file changes at the VAM, and then restart your application.

Related Information:

...Alarm Area Filtering Example

...Realm Area Filtering – Prevent tag selection by unauthorized users.

...Tag Area Filtering – Prevent tags that have been configured with a specified area from loading on a given workstation.

Alarm Area Filtering Example

In this example, a large plant has two main areas (AreaA and AreaB), each of which has a set of skilled personnel that operates and monitors only the equipment processes in that area. Because the plant operators are specialized only in their own area and need only respond to the alarms in their area, the plant management has requested that the work-stations in each area display only those alarms that are relevant. Given workstations:

- WorkstationAreaA
- WorkstationAreaB

WorkstationAreaA Configuration

The workstation named WorkstationAreaA requires the following configuration:

1. A text file named, WorkstationAreaA.startup must be present in the WorkstationSettings sub-folder of the application.

2. The <Alarm_Manager> section added to the WorkstationAreaA.startup file.

3. The AlarmArea property should be set to AreaA to indicate that only alarms configured with an Area property of AreaA should be visible on this workstation.

The completed WorkstationAreaA.Startup file should appear as follows:

```
<Alarm_Manager>
; The alarm areas to show
AlarmArea = AreaA
```

The result of the above configuration is that the workstation named WorkstationAreaA will only display alarms whose Area property has been configured as "AreaA".

WorkstationAreaB Configuration

The workstation named WorkstationAreaB requires the following configuration:

1. A text file named, WorkstationAreaB.Startup must be present in the WorkstationSettings sub-folder of the application.

2. The <Alarm_Manager> section must added to the WorkstationAreaA.Startup file.

3. The AlarmArea property should be set to AreaB to indicate that only alarms configured with an Area property of AreaB should be visible on this work-station.

The completed WorkstationAreaB.Startup file should appear as follows:

```
<Alarm_Manager>
; The alarm areas to show
AlarmArea = AreaB
```

The result of the above configuration is that the workstation named Work-stationAreaB will only display alarms whose Area property has been con-figured as "AreaB".

Note: Alarm area filtering does not prevent tags from loading; there-fore, the alarm tags associated with the hidden alarm areas will con-tinue to appear in the Tag Browser (for those users who have the security privileges to access the Tag Browser).

Realm Area Filtering

Realm area filtering is based on a combination of user groups (as defined using the security manager) and a realm of tag area names. It will affect your application in the following ways:

- Members of a user group can see and acknowledge only the alarms from areas matching their designated realm list.

- When working with reports, members of a user group can select only the tags from areas matching their designated realm list.

- When working with the Historical Data Viewer, members of a user group can select only the tags from areas matching their designated realm list.

- For VIC connections, a realm having the same name as each security-group, must be provided. Users will be able to log in to only the realm that matches their security group. Users who are not part of a security group may only log on to the realm designated by the RootNamespace property.

- The tag browser will also be affected, such that users in a given security group will see only the tags that match their designated tag areas. Note that this applies only to the tag browser – not to tags drawn on application pages. All tags will be visible on all pages to all users. Any tag's trend window may be viewed.
- Security managers who are members of a realm are able to see only those accounts and roles that are also members of the same realm.

Note: Realm Area Filtering will affect how you configure the roster so that users can acknowledge alarms via email or SMS-text message. Each contact user name in the roster (other than the admin account) must include the full group qualifier.

Why Should I Use Realm Area Filtering?

Realm area filtering is most often used for larger applications where there are groups of users for various areas of the application. Use Realm area filtering to specify:

- What alarms should be visible to a user, based on their security-group.
- What tag areas should be shown in the tag browser (if any) when no user is logged on to the application.
- What tag areas should be shown in the tag browser, reports screen and historical data viewer when a user is logged on to the application.
- What tag areas should be shown in the tag browser when a super user is logged on to the application.

Note: A super user (aka, administrative user) is one who does not belong to any security-group.

While realm area filtering can prevent users from acknowledging alarms in areas they are not authorized for and can also prevent them from drawing tags having those areas, it does not affect any page displays other than the alarm list, and does not restrict access to controls.
If you wish to restrict user access to pages or to control tags, use application-specific security privileges.

How Does Realm Area Filtering Differ From Tag Area Filtering and Alarm Area Filtering?

Realm area filtering affects alarms configured with specific areas, hiding them from given users according to their security-group, and is not limited to any one workstation (i.e. the user may logon to any workstation and they will still only have access to the alarms permitted by the filter). Tag area filtering prevents tags that have been configured with specific areas from loading on a given workstation. Alarm area filtering hides alarms associated with specific areas on the Alarm page on a given workstation.

Where is Realm Area Filtering Configured?

Realm-area filtering is primarily configured in your application's Settings.Dynamic file. You will also need to configure user accounts such that they belong to a specific security group.

> Note: If using Windows Security Integration and Realm Area Filtering, you must add the realm name and prefix to the account using the VTScada accounts dialog. (e.g. realm:username@company.com)
> If a user needs multiple realm logons, they will require multiple Windows accounts.

How do I Configure Realm Area Filtering?

The following elements are involved in realm area filtering:

- One or more realms.

> Note: One or more realms are required if VTS/IS is installed and realm area filtering is to be applied for VTScada Internet Clients
> If realm area filtering is to be applied to clients who do have VTScada installed and who access their application over a local area network (LAN), realm configuration is not required.

- NameSpaceDelimiter property in Settings.Dynamic.
- GroupLogin property in Settings.Dynamic.

- One or more super-user security accounts. (Accounts with no group designation.)
- One or more user security accounts. (Accounts with a group designation.)
- [RealmAreas] section in Settings.Dynamic.
- [*-RealmAreas] configuration section.
- [<Area>-RealmAreas] section in Settings.Dynamic.
- Area properties in the above section.
- RootNameSpace property defined for users who are not part of any security group, but who will logon to a VTScada Internet Server.

Realm Configuration

A realm is a set of one or more VTScada applications that run on a VTScada Internet Server (VTS/IS). VIC users may access one or more applications contained within a realm over the World Wide Web, provided that they are able to provide credentials (i.e. a valid username and password) when requested by the VTS/IS.

Realm Areas and the VTScada Internet Server

Realm Areas also affect VTScada Internet Client access to an application. On the VTScada server, add one realm for each user group. Each realm must be given the same name as the group, and must include a reference to this application.

An operator can then connect to the application using a URL that includes the name of the Realm he is connecting to. For example, members of the Western group would use the address: http://www.yourdomain/Western.

Note: Restricting group access to areas does not mean that the operators cannot see the tags belonging to areas outside of their group's defined realm area. It does mean that they cannot see or acknowledge alarms resulting from those tags. It also restricts their ability to use the tags in reports and the trends screen.

NameSpaceDelimiter Property

The NameSpaceDelimiter property is a application property that sets the character that should be present to separate group names from user names. The NameSpaceDelimiter property is used when a manager is creating new user accounts if the GroupLogin property is also set to TRUE (1).

To specify a namespace delimiter:

1. Open your application's Settings.Dynamic file.
2. Locate the NameSpaceDelimiter property under the <SECURITYMANAGER-Admin> section.
3. Enter the character(s) you wish to use as your namespace delimiter following the equals sign, as shown:

```
NameSpaceDelimiter = :
```

The usual choice of character for a namespace delimiter is a colon :

4. Save and close the Settings.Dynamic file.
5. Start VTScada if it is not already running
6. In the VAM, select the application and click on Properties.
    You may need to logon if security has been enabled.
7. Click Import File Edits.
8. Click on the button, Import.

GroupLogin Property

The GroupLogin property in your application's Settings.Dynamic file can be set to 1 to enable group logons. When the GroupLogin property has been set to 1, the Please Logon dialog (that opens when the Logon button in the Display Manager's title bar has been clicked) will include a Group field into which users must enter their group name.

To set the GroupLogin property:

1. Open your application's Settings.Dynamic file.
2. Locate the GroupLogin property under the <SECURITYMANAGER-Admin> section.
3. Enter a 1 following the equals sign, as shown:

```
GroupLogin = 1
```

4. Save and close the Settings.Dynamic file.

5. Start VTScada if it is not already running.

6. In the VAM, select the application and click on Properties.
    You may need to logon if security has been enabled.

7. Click Import File Edits.

8. Click on the button, Import.

Super User Security Account

A super user is one who does not belong to any group, and who has the ability to administer their own user base by adding new user accounts to a given group.

To create a super user account, simply add a new user account to the application.

Group User Security Account

A group user is one who is associated with a specific security-group.

To create a group user account, add a new user account to the application, making sure to specify the group name, followed by the namespace delimiter (see NameSpaceDelimiter Property above), followed by the username in the New Username field of the Add Account dialog (e.g. GroupA:JSmith).

> Note: When a group user logs on, they will have to enter the group (e.g. GroupA) in the Group field, their account name (e.g. JSmith) in the Username field, and their password in the Password field of the Please Logon dialog.
>
> If logging into a VTScada Internet Client, they must log into a realm name that matches their security-group. You must ensure that the realm has been configured in the VTScada Internet Server.
>
> Super users, who are not members of any group , will not be able to log on over the internet unless the application property RootNamespace has been added, and its value set to the name of a realm created for the use of these accounts.

[REALMAREAS] Section

The [REALMAREAS] section of Settings.Dynamic is used to define the alarm areas that should be visible when no user is logged on. This applies to AlarmList banners that may have been drawn on the default page – the Alarms page will not be accessible when no user is logged on. To create a RealmAreas section:

1. Open your application's Settings.Dynamic file.
2. Add the [REALMAREAS] section heading at the end of the file.
3. Enter "Area = " followed by the area you wish to be visible to users who are not logged on to your application.

(You may wish to protect your system by not allowing users who are not logged on to view any alarm areas.)

You may enter as many areas as you require, with each area on a separate line. For example:

```
[REALMAREAS]
Area = Overview1
Area = Overview2
```

4. Save the Settings.Dynamic file and load it into your application.

[*-REALMAREAS] Section

The Settings.Dynamic [*-REALMAREAS] section is used to select the tag areas that should be visible when a super user is logged on. (A super user is one who does not belong to any group.)

To specify the tag areas that should be visible when a super user is logged on:

1. Enter the [*-REALMAREAS] section heading beneath the [REALMAREAS] section heading and corresponding properies at the end of your application's Settings.Dynamic file.

2. Enter "Area = " followed by the area you wish to be visible to super users who do not belong to any group.

You may enter as many areas as you require, with each area appearing on a separate line. You may also use the asterisk (*) wildcard character. (For example, to view all areas, you could enter *. To view all areas ending with "1" you could enter *1. To view all areas beginning with "a" you could enter a*.) For example:

```
[*-REALMAREAS]
Area = *
```

3. Save the Settings.Dynamic file and load it into your application.

[GroupName-REALMAREAS] Section

The Settings.Dynamic [GroupName-REALMAREAS] section to define the alarm areas that should be visible when a user in a given security-group user is logged on.

To specify the visible alarm areas:

1. Open the application's Settings.Dynamic file.

2. Add the [GroupName-REALMAREAS] section heading at the end of the file.

3. "GroupName" should be replaced by the name of the security group for which you are granting access to alarm areas.

4. On successive lines below the heading, enter "Area = " followed by the area you wish to be visible to users who belong to this security group.

You may enter as many areas as you require, with each area appearing on a separate line. You may also use the asterisk (*) wildcard character.

(For example, to view all areas ending with "1" you could enter *1. To view all areas beginning with "a" you could enter a*.) For example:

```
[GroupA-REALMAREAS]
Area = North
Area = South
[GroupB-REALMAREAS]
Area = East
Area = West
```

5. Save the Settings.Dynamic file and import it to your application.

Related Information:

...Realm Area Filtering Example

...Internet Realms

...Tag Area Filtering – Prevent tags that have been configured with a specified area from loading on a given workstation.

...Alarm Area Filtering – Hide alarms that have been configured with one or more specified areas on a given workstation.

...Refer to the VTScada Admin Guide for the following property definitions:

...RootNamespace

...GroupLogin

...NameSpaceDelimiter

...RealmAreasExcludeInvalid

Realm Area Filtering Example

The Acme Pasta Company has a large food processing plant that is run using a VTScada application. This application is large enough that two sets of operators are employed – one set in the Pasta division and the other set working in the Sauce division.
On-site, there is one stationary workstation named, MasterControlStation. The system managers wish this workstation to be a VTScada Internet Server that will run the application, allowing employees access to the application using VTScada Internet Clients on laptop PCs

when they are on-site and when they are away from the plant to ensure smooth, round-the-clock monitoring and control of the system.
Operators in the Pasta division need never see or acknowledge Sauce division alarms and vice versa.
The managers would also like to have a group that enables them to logon and monitor both the Pasta and Sauce divisions.

VTS/IS Configuration

A VTS/IS (VTScada Internet Server) is a PC with VTScada installed and for which a VTS/IS license has been purchased. A VTS/IS enables VTScada Internet Clients to access its running VTScada applications over the World Wide Web using the Microsoft Internet Explorer web browser.
In this example, the VTS/IS is to be the on-site workstation named, "MasterControlStation".

1. Select the VAM's Internet Setup button on MasterControlStation.
   The VTScada Internet Client/Server Setup dialog will open.
2. Open the Server Setup tab.
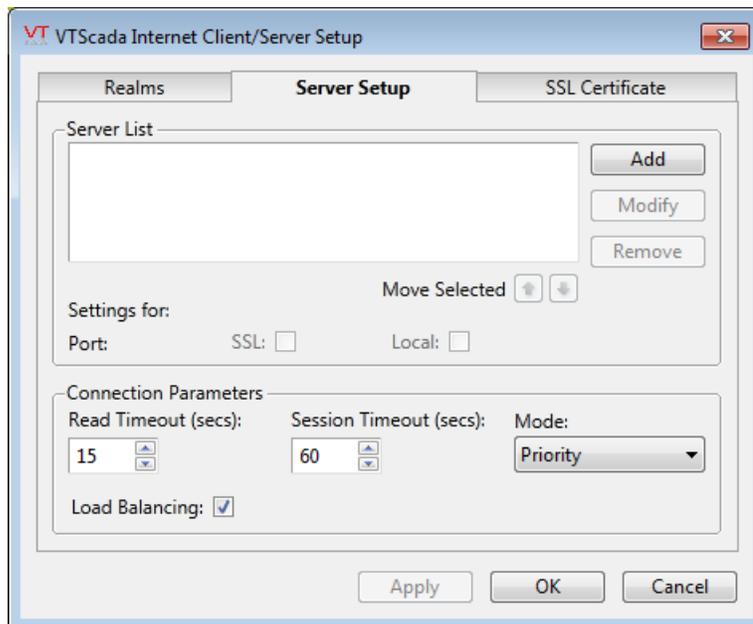
3. Click Add to open the Add Server dialog.
4. Enter MasterControlStation in the Server field.
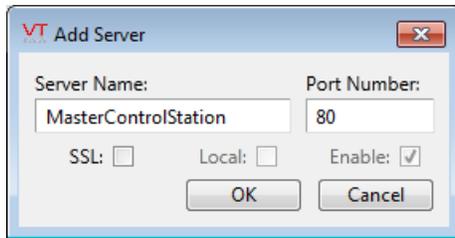
5. Click OK. You will be returned to the Server Setup tab where the VTS/IS will be referenced.
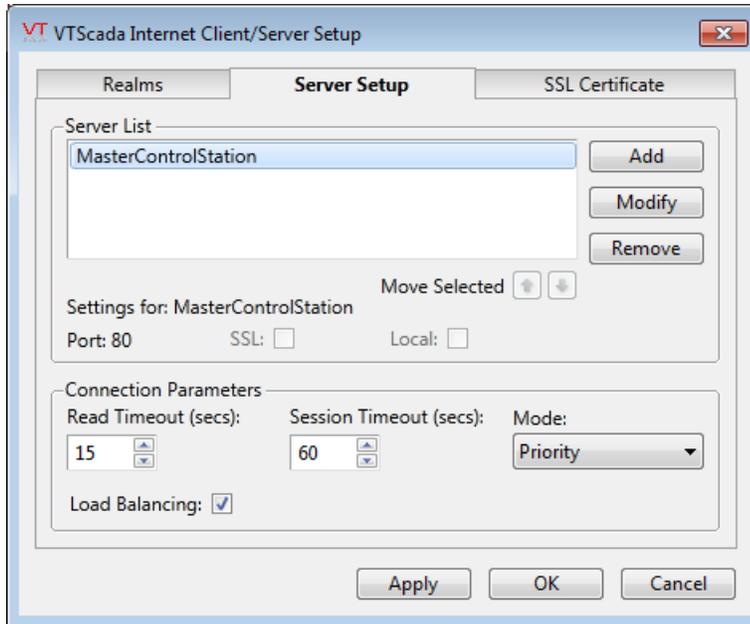
6. Click OK.

After configuring the server, you can configure a realm.

Realm Configuration

A realm is a set of one or more VTScada applications that run on a VTScada Internet Server (VTS/IS). VIC users may access one or more applications contained within a realm over the World Wide Web using Microsoft Internet Explorer, provided that they are able to provide credentials (i.e. a valid account name and password) when requested by the VTS/IS.

In this example, three realms are required: a Pasta realm, a realm, and an All realm (for managers).

1. Click the VAM's Internet Setup button on the VTS/IS.
2. Click Add in the Authorization Realms section of the VTScada Internet Client dialog. The Add Realm dialog will open.
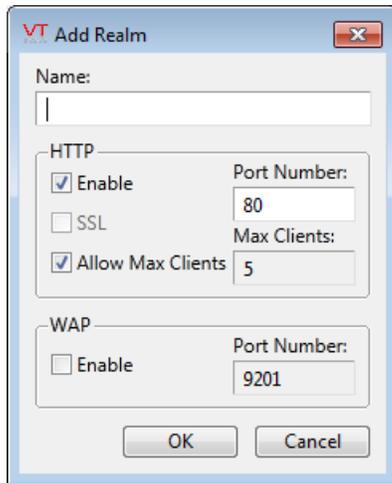
3. Enter "Pasta" as the name for this realm in the Name field.
4. Click OK.
   The Add Realm dialog will close, and you are returned to the VTScada Internet Client/Server Setup dialog where the name of the new realm will be displayed in the Realm drop-down list.
5. Click Add in the Applications Available On section. The Add Application dialog will be displayed.
6. Select the name of the application.
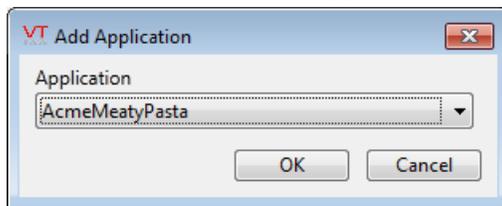
7. Click OK.
   The Add Application dialog will close, and the selected application will be displayed in the realm's list of applications.
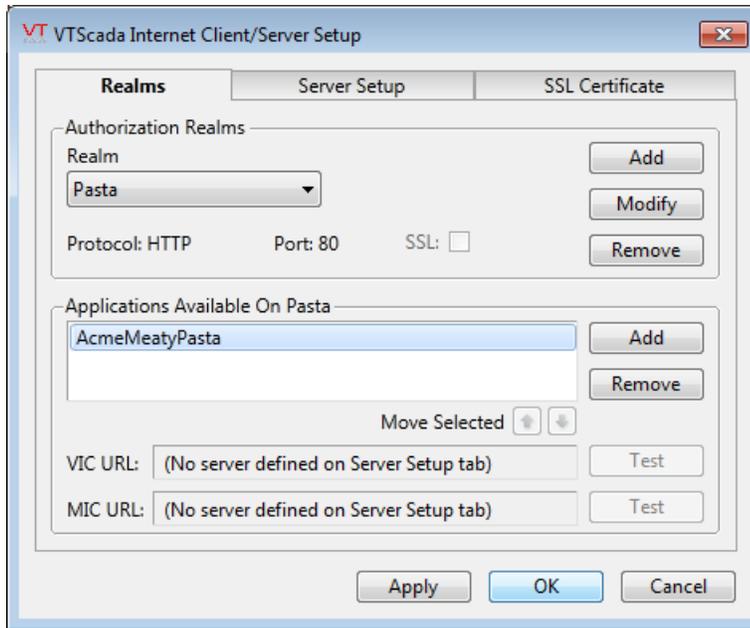
8. Repeat steps 2–7 to add a Sauce realm and an All realm.

9. Click OK. The VTScada Internet Client dialog will close.

Settings.Dynamic Configuration

Configure your application's Settings.Dynamic file for group logons as follows:

1. Open your application's Settings.Dynamic file using a text editor.

2. Locate the GroupLogin property under the <SecurityManager-Admin> section.

3. Set a value of 1 for GroupLogin.

4. Locate the NameSpaceDelimiter property under the <SecurityManager-Admin> section.

5. Set a value of : for NameSpaceDelimiter.

```
<SECURITYMANAGER-Admin>
GroupLogin = 1
NameSpaceDelimiter = :
```

6. Enter the following at the end of the Settings.Dynamic file:

```
[*-REALMAREAS]
Area = *
[Pasta-REALMAREAS]
Area = Pasta
[Sauce-REALMAREAS]
Area = Sauce
[All-REALMAREAS]
```

```
Area = Pasta
Area = Sauce
```

7. Save and close the Settings.Dynamic file.

8. In the Application Configuration dialog for the application, use the Import File Edits screen to import the changes that you have made to the Settings.Dynamic file.

Group User Security Accounts

The last step in configuring this system would be to create group user security accounts for the operators and managers who will be using the system. The following steps assume that you are working inside the running application, but apply equally to the security dialogs of the Application Configuration dialog.

1. Logon to the application with an existing user account that does not belong to any group and that has at least the Manager privilege.
   If security has not yet been enabled, click on the Logon button and create such an account after activating security.

Note: The Please Logon dialog will now include a Group field, since the GroupLogin property has been set to 1. Leave the Group field blank on this occasion since you are logging in as a "super user".

2. Click the Logon button (which is now labeled with the account name of the account that is logged on).
   The options dialog will open.

3. Click the Users button.
   The Accounts dialog will open.

4. Click Add.
   The Add Account dialog will open.

5. Enter the group to which this user should belong, followed by the NameSpaceDelimiter, followed by the account name in the Username field (e.g. Pasta:JSmith).

6. Enter a password for this user in the New Password field.

7. Enter the password in the Confirm Password field.

8. Select the privileges you wish this user to have.

You must grant at least the Internet Client Access privilege to allow the user VIC access.

9. Click OK.

You will be returned to the Accounts dialog where the new user account will be referenced.

10. Repeat steps 4 through 9 to create as many user accounts as you require for this application for the Pasta, Sauce, and All groups.

Users must now connect to the application using the realm matching their realm-area. For example: http://MasterControlStation/Pasta

The VIC logon does not prompt for security-group as this is implied by the realm. Users will connect to their realm using just their account name and password.

Client / Server Configuration

An application can be configured to run on multiple servers. Backup servers will take over when the primary server goes offline (redundancy). Separate services within VTScada, such as the Alarm Manager and the Historian can be configured to run on different primary servers simultaneously, (load distribution) each with its own list of backup servers. Applications that run on multiple computers may also have client workstations – those where operators can work, but which are neither primary or backup servers.

Note: All computers must be running the same version of VTScada

Note: Before configuring for multiple servers, take time to become familiar with how version changes are deployed.

All that is required is:

- A VTS license for each computer, where those installed on servers have the license options, Server Capability and Redundancy Capability.
- A list of the servers, configured within the application. The list will show which computer is primary and which is backup for each part (or all parts) of VTScada.

It is assumed that there is some type of network, connecting the servers. There are two steps, as described in the following topics.

1. Load the application onto another computer with VTScada installed (preferably by doing a Get-From-Workstation operation).
2. Configure the Server list.

Optionally, you may wish to create an advanced server list for load distribution.

Note: VTScada assumes that you have a properly configured DNS for your network. This may not be the case in all situations. In the case of a local network with no DNS, set the Setup.INI property RPCConnectStrategy to a valid value.

> Note: VTScada relies on a Windows® operating system function call to obtain the NetBios name of workstations. That function will truncate names that are longer than 15 characters. You are advised to limit workstation names to be 15 characters or less.

Related Tasks:

...Install an Application on Client (or Backup) Computers – Ensure that the application is running on more than one computer.

...Local Changes versus Deployed Changes – Describes when and how version changes are deployed to other workstations.

...The Configuration Deployment Process – Illustrated reference.

...Designate Primary and Backup Servers – Configure for redundant operation.

...Designate Servers for Specific Services – Configure for load distribution.

Related Information:

...Refer to "RPCConnectStrategy" in the VTScada Programmer's Guide for advanced configuration options.

Designate Primary and Backup Servers

Primary and backup servers may be chosen before or after the application has been installed on remote computers. The process can be completed from any computer on which the application is running. The only restriction is that the application must be installed on whichever computer(s) are to be designated primary before that designation is made.

1. Open the Application Configuration dialog.
2. Select the Edit Server Lists option from the menu.

3. Click on the list of available servers so that you can view the list
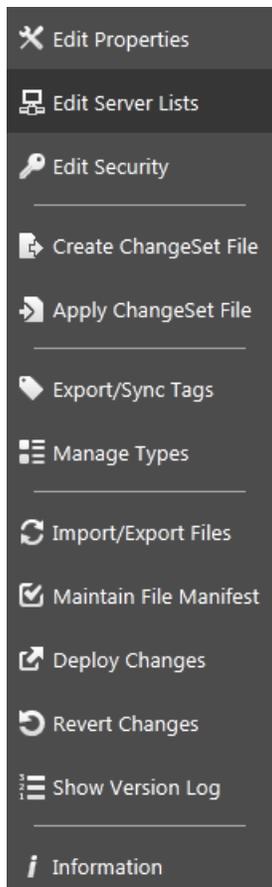
4. Select the workstation that is to be the Primary Server

Note: All workstations on your network will be listed. Select only that workstation which is running the application and which you want to designate as the Primary Server.

5. Click on the Plus button to add it to the list

6. Repeat the last three steps to add as many backup servers as required.

The first server shown in the Common Server List will be the primary server. Having created the list, you can use the up and down arrows to re-order it. In the event of server failure, backup servers will take over as the primary in the order that they are found in the list (from top to bottom).

Note: Any server that is running the application, but that is not entered in the server lists is referred to as a "pure client".

Related Information:

...Examples of Server Configuration

...Designate Servers for Specific Services

...Workstation-Specific Server Lists

...Install an Application on Client (or Backup) Computers

Install an Application on Client (or Backup) Computers

At some point in the process of setting up a **remote application**[1], you must install the application on those computers.
The Get From Server option is preferred for distributing an application between computers that are connected on a network. By using this method, you ensure that the most up-to-date version of the application is installed, and that future updates are distributed automatically. (Assuming a properly-configured server list.)
If the application is based on an OEM layer, and that OEM layer is present on the remote workstation, it will also be retrieved at the same time as the application.

---

[1]An application that simultaneously runs on two or more computers. Usually configured with at least one primary server and one backup server.

After opening the *Add Application Wizard*, choosing *Advanced*, then *Get from Workstation*, you will arrive at the *Get from Workstation* page.

1. Enter the machine name or IP address of the workstation containing the application.
   On a network with a properly configured domain name server, all local machines that have VTS installed will be available for selection in the drop-down list.

2. Click Next.
   Applications available on the selected workstation will be shown. An asterisk

will mark any that already exist on your workstation.

3. Select the application you want.

4. Click Next.

   A confirmation page will open, where you may review your choices before
   clicking Finish.

After clicking Finish, the application will be copied to your workstation
and added to your list of available applications in the VAM.

Troubleshooting:

- No servers are shown in the Choose Server dialog.
  This may indicate that your network domain name server (DNS) is not prop-
  erly configured. You can ignore this problem and type in the name of the
  computer where the VTS configuration server is running.

- The Find Servers dialog reports that the requested computer is unavailable.
  This may be preceded by a "Contacting Workstation..." message.
  The specified workstation is not available, or VTS is not running on that com-
  puter.

- The application you expected to find is not listed.
  Applications that exist on both computers are not shown. If it is not listed in
  your VAM, try the Find Existing option.

Next Steps:

...Designate Primary and Backup Servers

Designate Servers for Specific Services

In larger applications, there may be reasons to divide the load of running VTScada between more than one server. For example, you may wish to have CPU-intensive services such as the alarm manager handled by one computer while another looks after disk I/O tasks such as logging. Another example is to define primary and secondary servers for the I/O device driver. Load distribution is configured using the advanced tab of the Edit Server Lists dialog.
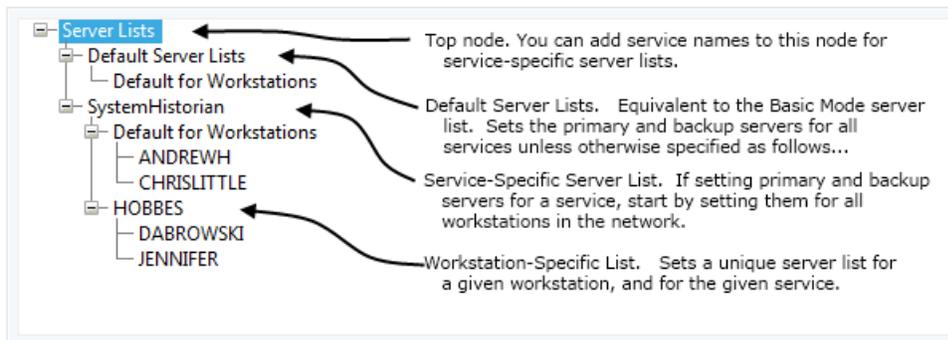
Note: The following instructions use an existing tag (System Historian) to define a service-specific list. When creating a service-specific list for a driver or Historian, there is an advantage to creating the list first, typing in a name rather than selecting the existing tag. After the list has been created, create the matching tag using the same name. By doing so, you ensure that the service uses the custom server list immediately rather than using the default list for a period of time.

To add a service for which you intend to designate primary and backup servers:

1. Open the Application Configuration dialog.
2. Select the Edit Server Lists option from the menu.
3. Click on the button labeled "Advanced Mode"
4. Right-click on the node, Server Lists
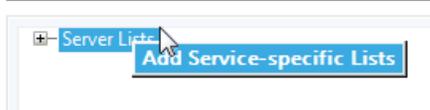
5. Click on Add Service-Specific Lists

The Add Service-specific Lists dialog opens
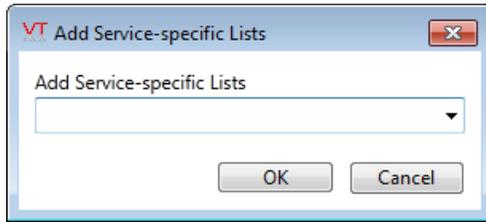
6. Select the service for which you want to define Primary and Backup Servers. The example shows a service-specific list being created for the System Historian.
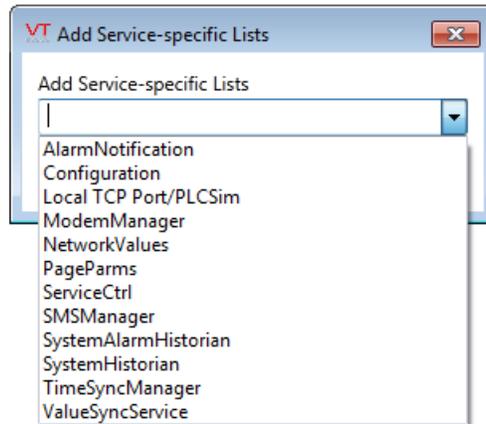
Note: I/O services are selected by the name of the attached Device Driver. These will only appear in the server list while the application is running. Also, the System Historian will be shown in the list only while the application is running

7. Click on OK.
The service is listed in the menu.
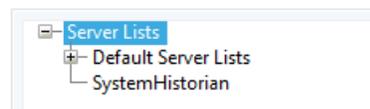
Note: Warning: Do not leave the configuration in this state. As configured, this means that there is no Primary Server for the System Historian.

8. Right-Click on the service name to open the context menu.
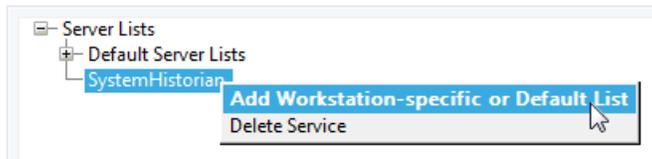
The Add Workstation-specific or Default List dialog opens
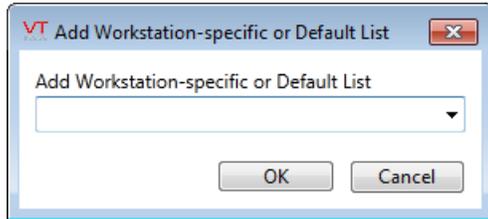
9. Select the option, "Default for Workstations" and click OK. [1]

    The entry Default for Workstations is added to the tree.

10. Right-click on Default for Workstations
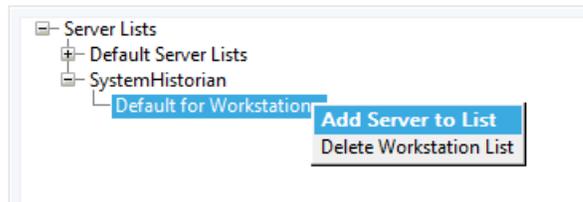
11. Click on Add Server to List

12. Select the computer that is to serve as the Primary Server for this service.

13. Click OK

    Repeat the last three steps for the backup servers you want to designate for this service.

14. Click Apply
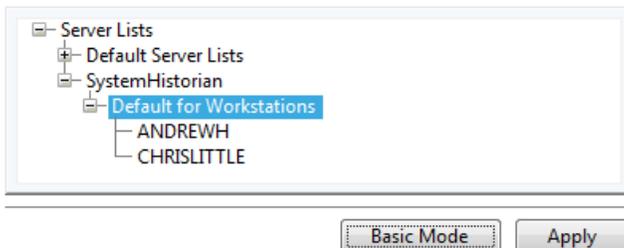
[1]It is possible to define a server list that is in effect only for a named workstation.

Related Information:

...Examples of Server Configuration

...Designate Primary and Backup Servers

Workstation-Specific Server Lists

There are two reasons for creating a workstation-specific server list:

- There may be an advantage for one or more servers to work independently. For example, in an application that spans many widely-distributed sites, the local sites might maintain their own server lists.

- Developers may be working in an environment where they do not have an active connection to the primary server. By adding a server list that applies only to their computer, then setting their computer as their own primary server and no back-up servers, they can work without VTScada attempting to synchronize with the real primary server.

To create a workstation-specific server list:

1. Open the Application Configuration dialog.

2. Select the Edit Server Lists option from the menu.

3. Click on the Advanced Mode button.

4. Create a default server list that will apply to all workstations other than the ones you are about to configure.
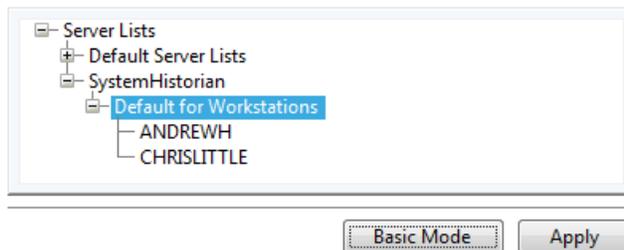
5. Right-click on the service name.
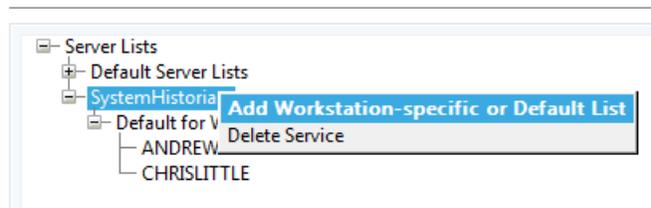
6. Click on the Add Workstation-specific or Default List option.

7. Select the name of the first workstation for which a workstation-specific list will be defined.
   In this example, the workstation is named "Hobbes"

8. Right-click on that workstation

9. Select the primary server for the service on that workstation.

10. Repeat as required for other services and for backup servers.
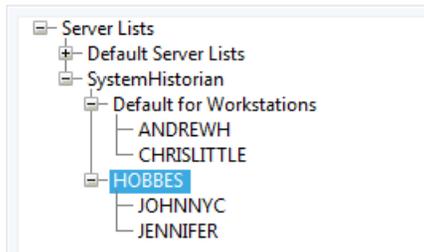
Related Information:

...Examples of Server Configuration

...Designate Primary and Backup Servers

...Designate Servers for Specific Services

...Install an Application on Client (or Backup) Computers

Examples of Server Configuration

The following examples are provided to help you see how server configuration might be used to your advantage.

Simple Configuration for Redundant Operation

ServerA is designated Primary for all services. ServerB is designated as backup.
In the event that ServerA is offline, ServerB will take over all operations.
Detail shown from the Basic Interface of the Edit Server Lists page of the Application Configuration dialog.
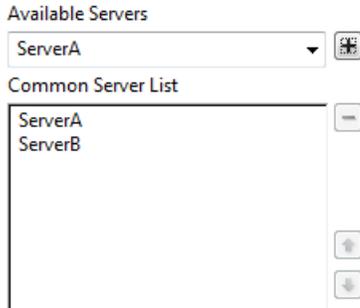
Simple Configuration for Load Distribution and Redundant Operation

ServerA is the Primary for the application's driver (tag I/O) and the Alarm Manager and is backup for all other services. ServerB is Backup where ServerA is Primary and is Primary where ServerA is Backup.

This balances the load so that I/O operations and Alarming are concentrated on one server while disk-intensive operations including logging are handled by the other.

In the event that either server fails, the other will continue running all services.

Detail shown from the Advanced Interface of the Edit Server Lists page of the Application Configuration dialog.
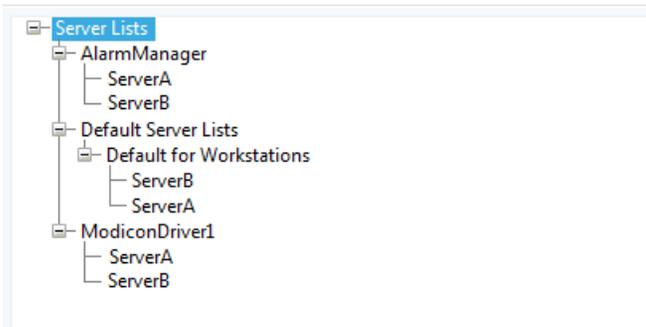
Historian Load Distribution

This example is designed to handle an extremely high data-logging load by providing three Historians, each with one redundant server. It is assumed that each Historian is logging data from 1/3 of the tags.
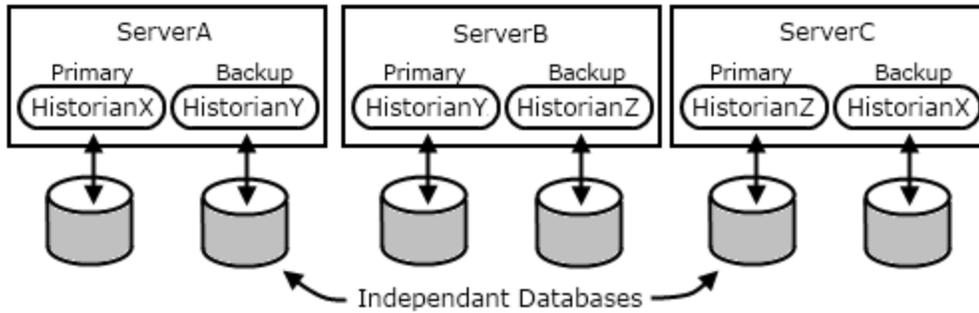
In this configuration, each of the machines will carry 2/3 of the disk activity load.

Server detail shown from the Advanced Interface of the Edit Server Lists page of the Application Configuration dialog.
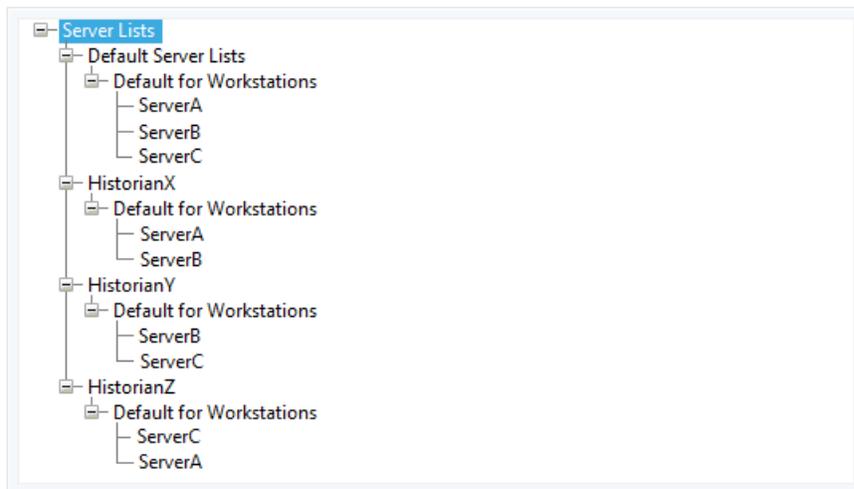
Related Information:

Historian and Logger Configuration

Client/Server Reference

Server configuration is done within the Application Configuration dialog of an application. After an application has been configured to run on more than one server, updates to the configuration list can be done from any of the servers or workstations on which it is running.

You build the list of the primary and backup servers by selecting computer names from the list, where each computer will run a copy of the application on a licensed VTScada installation. The computer name at the top of the list is the primary, and each subsequent one is the next backup in order.

Click the Advanced tab in order to configure separate lists for different services of VTScada, including configuration.
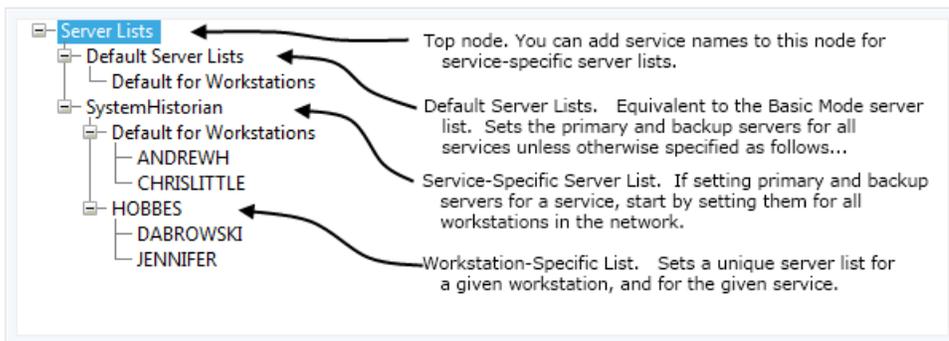
Related Information:

...Server Redundancy – Description of the fail-over process when the primary server is lost.

...The Configuration Deployment Process – How configuration changes are distributed.

...Operational Changes vs. Configuration Changes – Practical differences.

...VTScada Services Reference – Reference list

Server Redundancy

Server redundancy means that, if a primary server is unavailable for any reason, a designated backup immediately takes over and enables the application to continue to run. There is no limit to the number of backup servers you may designate, other than the fact that each must have VTScada license that enables them to function as a server.
Backup servers behave as clients until their designated primary fails. The following example shows how this process works.
Given 3 workstations:

- Primary configuration server A,
- Backup server B, and
- Client C.

While A is functioning, both B and C operate as clients of A. Data from the PLC or RTU is sent to A, then distributed to B and C. Operator control actions, initiated at B are sent to A and from A are sent to the hardware. Similarly, configuration changes made at B are sent to A, and from A are sent to C (with a confirmation sent back to B).

Primary Configuration Server

Client (Backup Server)
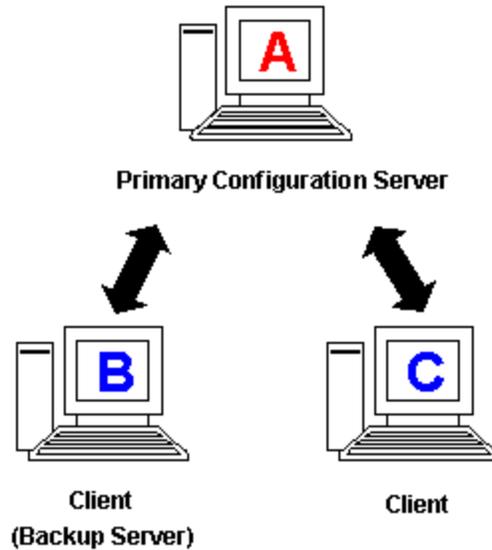
Client

If A is shut down, B immediately assumes the role of server, and enables the application to continue to run. C is now a client of B. Everything that was done by A is now done by B.

When the primary configuration server restarts, it synchronizes all of its application and data files with the backup server, and then re assumes its role as primary server, while the backup server resumes its role as a client of the primary server. While this process is underway, A is acting like a client to B.

Backup Server

"A" synchronizes data with "B"

"C" synchronizes data with "B"
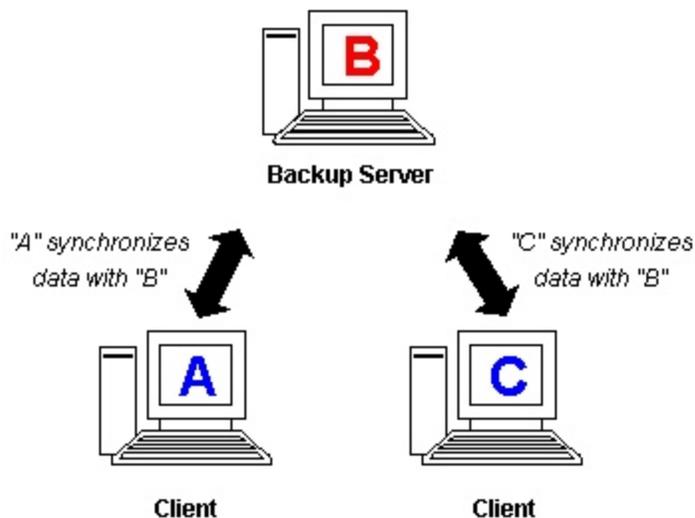
Client

Client

Once synchronized with B, A and B reverse roles; A assumes the role of primary server, and B is once again a client to A.

Primary Configuration Server

"B" and "A" are synchronized and reverse roles

"C" synchronizes data with "A"

Client (Backup Server)
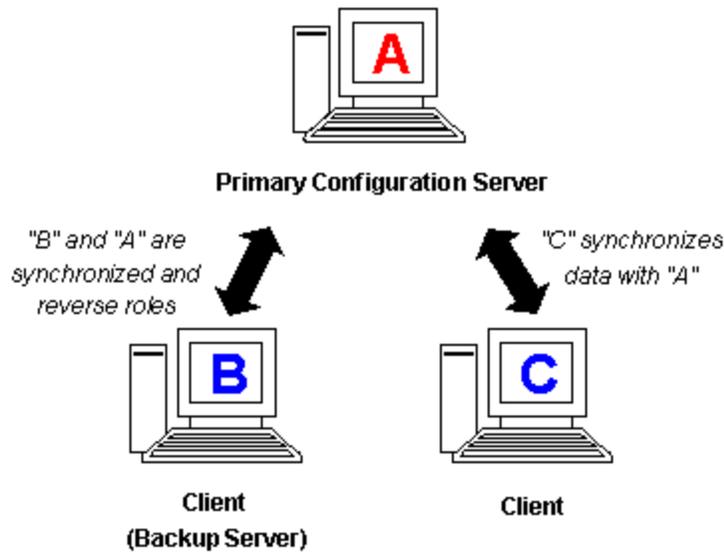
Client

Each time a server changes, all clients resynchronize their application data and files with that of the server.

The Configuration Deployment Process

Every change to an application is considered to be a version change. If the Auto-Deploy option is not selected, changes are "Local" and not distributed to other computers until you choose to deploy them. Otherwise, all changes are immediately sent to all computers on which the application is active.

Step 1: A developer makes a change to an application on workstation Wks1.

Wks1   Srv1   Wks2   Wks3
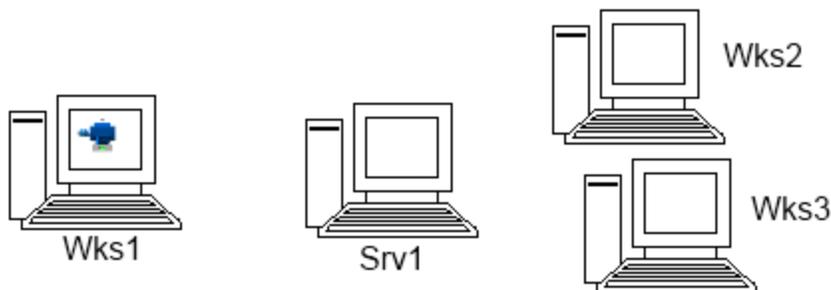
Step 2: The change is deployed. It is sent to the current configuration server, Srv1 which then sends it to workstations Wks2 and Wks3
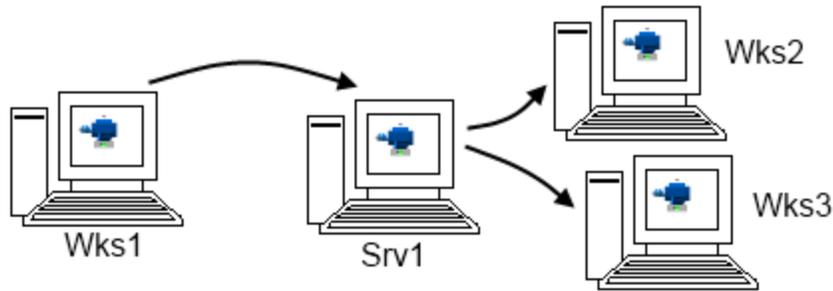
Step 3: Workstations Wks2 and Wks3 send an acknowledgment back to the server. This acknowledgment shows the current version number on those workstations.
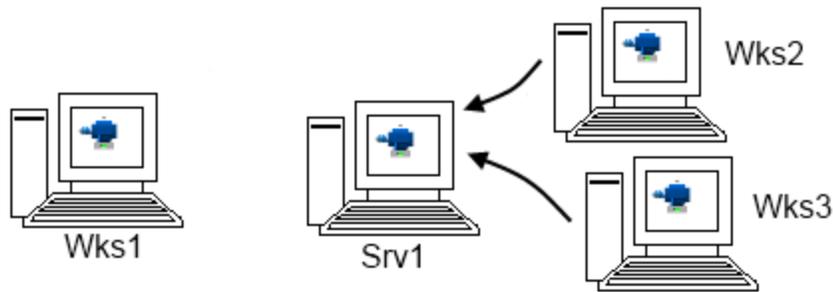
Step 4: After a few moments, the server distributes those acknowledgments to all the workstations. The result is that every workstation always contains a complete local copy of every other workstation's change history.
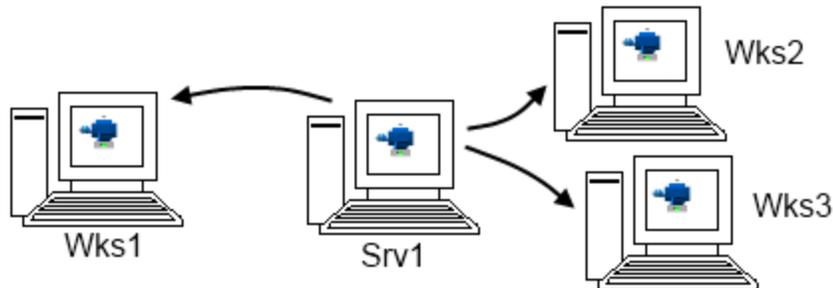
Should the primary configuration server go offline, the backup configuration server will immediately take over all of its duties.

Operational Changes vs. Configuration Changes

In versions of VTScada (VTS) prior to release 10, it was important to know the difference between operational and configuration changes to tags. That difference has become almost negligible.

- Operational changes are deployed immediately, regardless of whether the Auto-Deploy option is set.
- Operational changes can be made by an authorized operator without using the Tag Browser or the Idea Studio.
- Configuration changes must be deployed manually if the Auto-Deploy is off.

Operational changes are done while an application is in operation mode using the shortcut menu. To access the shortcut menu, right-click a tag's widget on an application page. Any of the options chosen from the short-cut menu will allow you to make operational changes to the tag's configuration that will immediately be propagated to all workstations running the remote application.
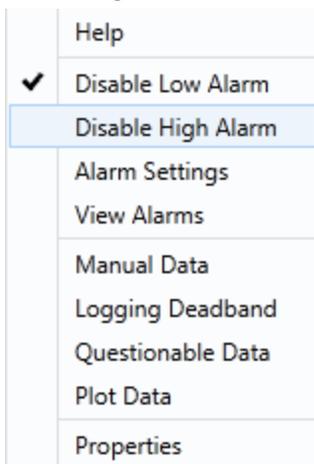
Configuration changes are those made using the configuration panels, or by importing source and tag files that have been edited outside of VTScada.

VTScada Services Reference

When planning your server configuration, you should consider the balance between services that are I/O intensive and those that place a heavy load on hard drive access. In larger systems, there can be significant benefit to distributing the load to separate these types of services. Your design should also consider system redundancy for risk management planning.

Each of the following VTScada services can be assigned its own server list. Some place a heavier load on data I/O while others require more

disk access. Some of the services listed here place minimal load on a system, and therefore may be assigned to servers at your discretion.

Note: every system will have different requirements and different priorities. No single guideline can be used for the design of all systems.

Alarm Manager

The service responsible for logging alarms and maintaining all alarm information. This is often considered a critical system and grouped with the primary server for tag I/O.

Alarm Notification

Used to decide which workstation should dispatch alarm notifications. This service is registered only on workstations that have the alarm notification feature enabled in their license key. Backup servers will assume the role if the higher-level servers don't have alarm notification feature enabled.

Configuration

The service responsible for distributing deployed changes to all workstations. In a working system, configuration changes usually do not occur often, but may place some load on disk access when they do. Some developers prefer to keep the configuration server separate from the primary server for tag I/O in larger applications.

Modem Manager

Used in VTScada installations that include the Alarm Notification System option. Provides data and voice telephony services for standard VTScada applications and enables modems connected to different machines to be managed as a common pool. This service seldom places a heavy load on a system.

Network Values

This service is in charge of updating changes on all PCs for remote applications when a change to a Network Value variable has occurred. Involves data transfers to all PCs in the network therefore there is little advantage to separating this service from others.

Operator Notes

This service is responsible for storing information entered on the Operator Notes page. The load from this service is primarily disk access, but is rarely large.

Page Parameters

Responsible for (in part) synchronizing taskbar page buttons. Places minimal load on a system.

Service Control

May be used by custom code to provide application-level control of RPC servership.

SMS Manager

Used in VTScada installations that include the Alarm Notification System option. Provides SMS messaging services for alarm notifications. This service seldom places a heavy load on a system.

System Historian

The System Historian service takes care of data storage and retrieval. This service is disk access intensive and is often separated from services that involve tag I/O and alarming.

User-Defined Historians

Developers may create their own Historian tags in addition to the SystemHistorian, thereby distributing the task of logging data. In a large application, setting different servers for each Historian is an effective way of distributing the load of disk access.

Tag I/O Services (Device Drivers)

Shows in the list only when the application is running. These are identified by the driver tag name. Tag I/O services are often considered a critical service along with alarm management. You might separate these services from those that require intensive hard drive activity.

Time Synchronization Manager

This service is responsible for ensuring that the system clocks on all client computers are synchronized with the computer that this service is running on. Load is minimal, involving the CPU and RPC communication rather than disk I/O.

Value Synchronization Service

Used only in the Totalizer tag, Counter tag, Selector Switch and (possibly) user-defined custom tags. Places minimal load on most systems.

Internet and Mobile Device Configuration

The VTS Internet Server, built into every copy of VTScada, provides remote access to selected applications using any of:

- VTScada Anywhere Client. A zero-footprint, JavaScript-based client that is compatible with all major web browsers on most platforms.
- Mobile Internet Client (MIC). Provides access to alarms, I/O, maps, sites and stations, in a format that is optimized for use on a mobile device.
- VTS Internet Client (VIC). An ActiveX-based client that enables operator access to the full application, much like a run time only license. May be used within versions of Windows Explorer that support ActiveX, or as a stand-alone client.

Two other technologies also make use of the VTScada Internet Server: The ODBC interface to query VTS history, and the Web Services feature that enables machine-to-machine linkages to your application.

Your license agreement controls the number of connections (if any) that you are able to make to the server.

To allow or deny specific IP address (or a range of IP addresses) at the VTScada server level instead of your firewall, create the properties, HTTPAllow and HTTPDeny in your SETUP.INI file.

When configuring your server, ensure that you select a port that is not in use by any other process on your server. Port 80, the default HTTP port, is often used by other programs. You can check your server by opening a command prompt as an administrator, then running "netstat -ao". You can then check the process ID value in the Windows task manager. For example:

```
C:\WINDOWS\system32>NETSTAT -AO

Active Connections
Proto Local Address Foreign Address State PID
TCP 0.0.0.0:80 MYSTATION-PC:0 LISTENING 4
TCP 0.0.0.0:81 MYSTATION-PC:0 LISTENING 10916
```

This example shows that port 80 is in use by process id #4, which belongs to Windows. Port 81 is in use by process #10916, which happens to be a configured VTScada server in this example.

VIC/MIC clients transmit the user logon credentials (username and password) using Basic Authentication, which is a simple, non-encrypted, Base 64 encoding of "username:password", and which can be decoded by network snooping tools if they can capture the message content. Wireshark as one example, will show the decoded credentials if it connected to a local machine or switch that performs the communications (Since switches don't broadcast network traffic for all to hear, the "listener" must be local to the communications path versus being anywhere on the network.) If Windows Security Integration mode is enabled, then the potential consequences will extend beyond the SCADA system should the operator's Windows credentials be stolen.

To secure user credentials against listeners that may have access to switches or workstations carrying this traffic, it is essential that you encrypt VIC/MIC sessions with SSL by installing an SSL certificate on the VTScada Internet server. The SSL certificate can be obtained from a 3rd party issuer such as Verisign, Thawte, GoDaddy, from an organization's own Certificate Authority (CA) infrastructure, or via local ad-hoc creation. (e.g. By using Open SSL tools.) To allow clients (browsers) to verify the certificate and not display an untrusted warning, they need the "root" certificate from the CA to be installed. The root certificate for most third-party issuers is already installed in most Web browsers.

If you have more than one VTScada Internet Server, then upon the loss of one server, both the VTScada Anywhere client and the VIC will fail-over automatically to the backup server. The mobile client does not have this ability, and must be directed to connect to the backup server.

> Note: Note: If your pages include tabbed folders, then the Unique Key property of the folder must be set in order for tabs to function when viewed using a Mobile Internet Client.

Related Information:

...Connecting over the Internet – Examples and instructions for operators, provided in the VTScada Operator's Guide.

Capabilities of Internet Clients

Internet clients do not provide all the tools and features that are available on a workstation with VTScada installed. But, only one feature is unavailable to all forms of client connections: the Idea Studio can be opened and used only on workstations with a full VTScada installation.

Capabilities of each client are listed here, starting with the most restrictive connection option:

The Mobile Internet Client provides the following additional features:
- For any application page, operators may choose between an minimal display optimized to reduce bandwidth, or a full display including all widgets and graphics.
- Operators can view I/O tag values on a page by page basis.

- Operators can set output tag values.
- View and acknowledge current alarms.
- View the alarm history.
- View a trend graph (HDV display) for any I/O tag.
  Graphs are images, updated only when recreated using a refresh. Data cannot be exported
- View a slippy map.
- If the mobile device supports the "pinch" user interface, operators may scale the HDV display.
- No access to the following system pages: Reports page, Operator Notes, the full HDV page, Internet Client Monitor.

The VTScada Anywhere Client adds the following features. Operators may:

- View and use all pages.
  - Subject to each operator's security privileges.
  - Reports can be generated to the display but not saved to file, printer or email.
  - HDV data cannot be exported.
- Perform limited configuration operations to a standard application, including:
  - Copy existing tags
  - Modify the properties of an existing tag, excepting font selection within a Font tag.
  - Delete a tag
  - Add new tags
- Modify Alarm Notification System options and pronunciation.
- Work with security
- Access the Application Configuration dialog to:
  - Edit properties
  - Edit server lists
  - Edit security
  - Import/Export Files

- Maintain the File Manifest
- Deploy changes
- Revert changes
- Show the version log
- Open the Information page.
- Add and edit page notes.
- Managers may work with security accounts.
- Administrators may configure security.

The VTScada Internet Connection (VIC) adds the following features to those listed above:

- View a list of available VTScada applications (as permitted by the VTS/IS)
- Create and save reports.
- Export HDV data.
- Use the VTScada print icon to print pages.
- Run a script application. Note: only one instance of a script application may run at a time – if it is running on a workstation, it may not also run on a VIC connection.
- Create and apply a ChangeSet.
- Export/Sync tags.
- Manage Types.
- Maintain logged-off sessions.

Server Requirements and Licensing

A minimum recommended configuration for a VTScada/IS is:

- Installation of VTS
- Windows® (Vista or later) . A server class operating system is recommended for production systems.
- Access across a network.

- In the event that the clients connect to the **VTScada/IS**[1] via proxy servers (common where a firewall is used), it is advisable that the proxy servers support the HTTP 1.1 protocol, rather than only HTTP 1.0.
- The server name should contain only letters, numbers and hyphens (as defined in Section 3.5 of RFC 1034 and Section 2.1 of RFC 1123 of the Internet Engineering Task Force). Characters such as underscores are not recognized by this standard and may cause unexpected behavior.

Licensing for VTScada Internet Servers

VTScada Internet Servers and the connections made to them are subject to the following licensing restrictions:

- The server is enabled only on a computer with either a full development or runtime VTScada license
- The number of concurrent VTScada Internet Connections you can make to a server is subject to your VTScada license. Depending on the licensing options you purchased with VTScada, you might have the privilege of connecting from zero to unlimited VIC workstations to your server.
- Mobile Internet Client access is permitted using the same license as VIC connections, with each mobile client being equivalent to one internet client.
- Licenses are managed based on a cluster of configured servers. For example, if one server in the cluster is licensed for 5 connections and another is licensed for 10, you will always have 15 available connections. If the 10-license server goes offline temporarily, your available license count will not drop to the remaining 5 but will remain at 15 on the server that is still running. No existing connections will be lost. For complete details, see: VTS/IS License Clusters.

If you are not sure how many VIC licenses you have available, you can find this information in the About VTScada dialog.

The number of licenses in use is counted on the basis of concurrent sessions, regardless of whether these are run from one computer or from many computers. Each connection to a VTScada Internet Server counts as one license in use. If, for example, you opened two tabs in Internet Explorer and connected to your VTScada Internet Server from each tab,

---

[1]VTScada Internet Server

then that would count as two licenses in use. Pop-up windows do not consume extra licenses.

An exception is that for support purposes, certain Trihedral-supplied debugging tools (such as the "Debugger", "Profiler", etc.) can always be accessed via the VTScada Internet Client, even if their use would exceed the license allocation.

License Clusters

A cluster is defined as the list of machines configured in the VTScada Internet Setup dialog. These must be computers running VTScada that can see each other via VTScada Remote Procedure Calls (RPC). For example, the following image shows a cluster consisting of two servers: Bridge-PC and EngineRoom-PC.
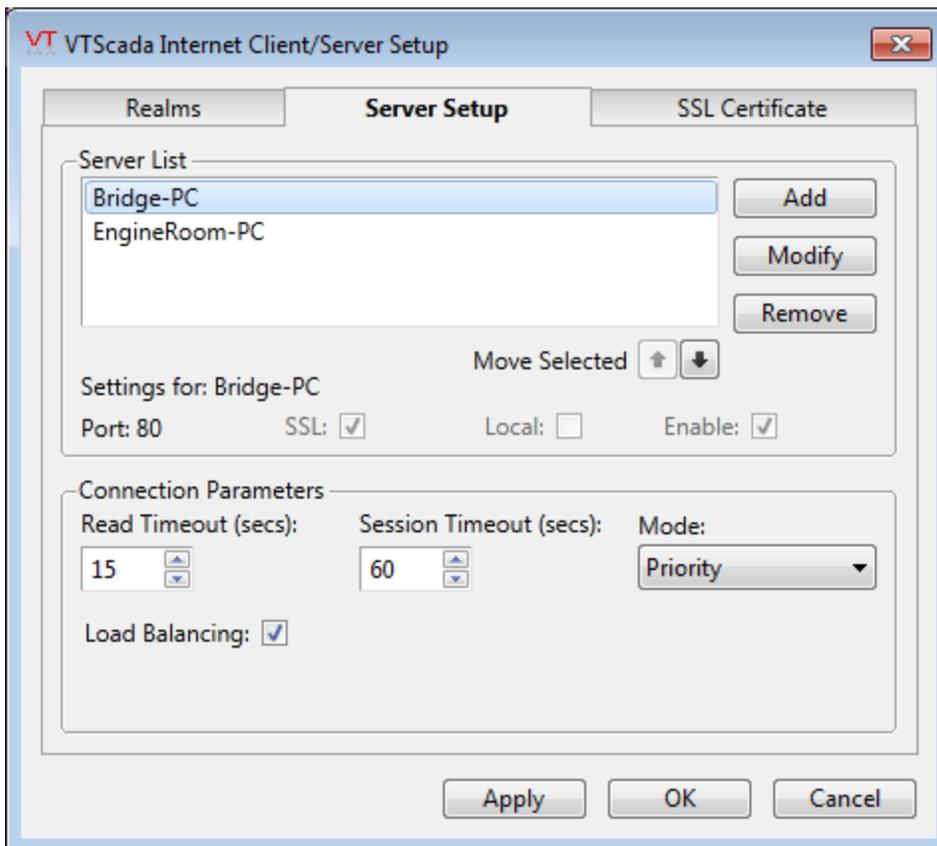
To configure a cluster, you need only add the server computers to the list. All other configuration and operation tasks are handled automatically.

> Note: You can switch off this behavior and manage connections on a machine-by-machine basis by setting the Setup.INI variable, ManualVICServerLists to 1.

Cluster members share information about all the connected VIC sessions across the cluster, along with license and configuration details. Updates to either the Realms or the Server Setup list on any one machine in the cluster are automatically propagated to all. Changes in timeouts or mode are automatically sent to all VIC sessions.

License counts are combined within a cluster. You can connect more clients to a server than would otherwise be allowed by its license limit, so long as the total number of connections within the cluster does not exceed the cluster's limit. In theory, any one machine could support all the connections available within the cluster's limits, although in practice this may exceed that machine's CPU or RAM limitations.

If a server fails, you do not lose its license count from the total during the time it is offline. All VIC connections to that machine are immediately re-routed to the remaining servers in the cluster that are running the application, even if this would exceed the licenses available on just those remaining machines.

If sub-networks within the organization become isolated such that the servers are temporarily unable to communicate with each other, each server will maintain the full license compliment of the cluster, even though this may exceed the license limit of that one machine.

When isolated servers re-join the cluster, if the combined total number of connections exceeds the cluster limit, no new connections can be made, but existing connections in excess of the limit are maintained. While the cluster maintains excess connections, an event will be logged in the alarm history every 10 minutes, warning of this state.

Realm session limits are maintained across a cluster such that the total number of connections to a realm throughout the cluster may not exceed the configured limit.

If the Load Balancing option is checked, client connections will be distributed between servers automatically, reducing the load on any one particular server.

Related Information:

...Client Requirements

Configure a VTScada Internet Server

The following is a generalized list of the major tasks required to configure a VTScada Internet Server. Each of these tasks is fully described in following topics.

Note: The VTScada Security Options dialog includes an option that controls whether VIC connections are maintained after a user logs out, but leaves the browser open. Choosing that option allows a faster re-connection time, at the cost of tying up a license for an undetermined time. Maintaining the VIC connection upon log out is not recommended.

Step 1: Grant the Internet Client Access Privilege to Selected Operators

VTScada does not permit any Internet access to an application in which security has not yet been enabled – your unsecured applications are always secure against remote access over the Internet. Within each application to which remote users will have access, you must enable security and grant the Internet Client Access privilege to selected accounts. If script applications are to be available, they must be added to a realm that includes at least one standard application. Access to the script application will be managed by the security configuration of the standard application.

1. Run the application on the VTS/IS.
2. Enable security and logon to the application with a user account that has at least the Manager privilege.

3. Create a user account for each of your intended remote clients. Ensure that each user account has at least the Internet Client Access privilege granted to it

Step 2: Establish the VTS/IS Security parameters

Access to the VTScada Internet Server is protected by the account name and password credentials held for each application by the VTScada Security Manager. This is the sole protection that is afforded to a VTScada Internet Server from unauthorized access. Therefore, these credentials must be guarded.

Note: When credentials are transmitted between the VIC and the server, the account name and the password are both transmitted using Base64 encoding. This encoding system is public knowledge and is entirely reversible, therefore the account name and password can be extracted by knowledgeable persons who are able to intercept the communication.

Security is provided implementing a virtual private network (VPN), or by using Transport Layer Security (TLS). This establishes an encrypted communications connection that is secure against decryption, replay attacks and many other hacking attempts. It is generally accepted that 128-bit SSL is sufficiently secure for financial transactions.
All systems that use the VIC over a WAN should use a VPN or SSL to secure their communication.

Step 3: Configure one or more VTScada Servers as VTScada Internet Servers

To establish your PC as a VTS/IS, you must assign it a port through which communications may occur. If you have a remote application with more than one VTS/IS server configured, you may also set up redundant operation parameters.
Note: You may configure a realm or a VTS/IS on any port you desire; however, if operating over a public network (e.g. the Internet), you will likely have to traverse firewalls and other security mechanisms. Configuring a

realm or VTS/IS to operate on other than the standard ports (port 80 for plain text HTTP, or port 443 for SSL-secured HTTPS), will likely require special configuration of such interposing security mechanisms. It is therefore advisable to operate on the standard ports whenever possible.

Step 4: Establish a Realm Containing One or More Applications

In short, a realm is a named list of one or more applications that are to be made available to VIC and Mobile Internet Client connections.
Note: If a realm is to contain script applications, it must also contain at least one standard application. Clients cannot connect to a realm composed entirely of script applications.
If Realm Area Filtering is enabled, you must create one realm for each security group, matching the realm name to the group name. Operators may connect only to the realm that matches their group. Super-users (those not belonging to any security group) may not connect to any realm unless you also add the application property, RootNamespace, setting its value to the name of a realm to which those users should connect.

Step 5: [Optional] Add a Realm Display Setup tag

You may choose to add one of these tags for each realm. Realm Display Setup tags allow you to define the level of control that operators will have over their display parameters.

Next steps:

...See: Secure Your Application in the VTScada Admin Guide. Covers steps 1 and 2. Activate security, create user accounts and assign privileges, including the Internet Client Access privilege, generate an SSL certificate.

...Setup the Internet Server – Covers step 3, configuring the server.

...Configure a Realm – Covers step 4, configuring the realm.

...Realm Display Setup Tags – Covers step 5, adding a Realm Display Setup tag.

Related Information:

...Options for Security Settings – Choice of action on log out.

Network Configuration

This topic is for those who want to allow VIC connections from outside their firewall to their VTScada Internet Server, where their site has a simple router and no internal network infrastructure services (e.g. DHCP/DNS). *Trihedral strongly advises that you use SSL for accessing your VTScada Internet Server from outside your firewall.*

Typically, in its default configuration, the router blocks all incoming connections. The router will probably have an embedded DHCP[1] server (for allocating local, non-routable IP addresses on the local network) and a DNS[2] forwarder (for resolving names using public, or ISP[3]-provided DNS servers). Most such routers allow the DHCP server to be disabled if address allocation is done by another means (such as an internal DHCP server or static IP assignments), but rarely provide DNS capabilities other than forwarding. This means that name resolution requests that cannot be directly resolved by the workstation are passed to the router, which forwards the request to an external DNS server

Name resolution is typically done by a machine inside the network perimeter by two means:

- A machine name without a domain being specified [e.g. VTSCADA1] results in a NetBIOS name resolution and yields a non-routable internal IP address [e.g. 192.168.0.5].
- A fully-qualified domain name (FQDN) being specified [e.g. vtscada1.trihedral.com] results in a DNS query being made to the router which forwards it to a DNS server on the internet to obtain a publicly-accessible IP address. This is normally the external IP of the router.

So long as the servers and clients (thick or thin) are within the network perimeter and these machines use NetBIOS-type names, this works. It will not work when someone wants to access a VIC server from outside

---

[1]Dynamic Host Control Protocol

[2]Domain Name Server

[3]Internet Service Provider

the network perimeter. As the router's firewall blocks all incoming con-
nections by default, to access the VIC server from outside the local net-
work perimeter requires:

- Port forwarding must be configured on the firewall such that the ports used
  by the VTScada Internet Server are forwarded to the VTScada Internet Server.
- Install a DNS server on the internal network that resolves the server's fully
  qualified domain name (FQDN[1]) to its local IP address.
- Configure all servers and clients to use that DNS server when inside the net-
  work perimeter. Typically this is done by disabling the DHCP server on the
  router and installing a DHCP server on the same machine as the DNS server.
  DHCP can then provide the correct IP addresses to use for DNS, the domain
  suffix to apply internally (e.g. trihedral.com) and the gateway (router)
  address.

This is sometimes referred to as "split DNS", "split horizon DNS", "split
view DNS" or "split brain DNS". With this setup, you configure the VIC
server list to use just the server's FQDN. When a machine is inside the
network perimeter, the internal DNS server resolves the server FQDN to a
local IP address and when outside the network perimeter, public DNS
resolves the FQDN to the router which port-forwards to the server.
In order for SSL to work, the FQDN of the server must be listed on the
SSL certificate and resolve to the server whether the client machine is on
the internal or external network. In other words, you need the name res-
olution of vtscada1.trihedral.com to yield two different addresses depend-
ing on the location of the client – hence the use of a split DNS in this
network configuration is recommended.

Securing a VTScada Internet Server

Before configuring the Internet Server, you must understand the relevant
security issues.
Access to the VTScada Internet Server is protected by the account name
and password credentials held for each application by SecurityManager.

---

[1]Fully Qualified Domain Name

This is the sole protection that is afforded to a VTScada Internet Server from unauthorized access. Therefore, these credentials must be guarded. When credentials are transmitted between the client and the server, the account name and the password are both transmitted using Base64-encoding. The encoding is public knowledge and is entirely reversible, therefore the name and password can be easily extracted.

Credentials are secured by using Transport Layer Security (TLS). This establishes an encrypted communications connection that is secure against decryption, replay attacks and many other hacking attempts. It is generally accepted that 128-bit SSL is sufficiently secure for financial transactions.

Note: These notes will refer to SSL (secure socket layer) security. SSL is an older technology and the term has become the de-facto name for internet security. While the term "SSL" is used here by convention, VTScada does use the more modern Transport Layer Security.

It is strongly recommended that all systems that use the VIC over a WAN use SSL to secure their communication.

To use SSL on a VTS/IS, you must purchase and install an SSL certificate. SSL security involves the server providing an X.509-compliant digital certificate to the client, permitting the client machine software (web browser) to positively validate that the server is truly genuine, and not a fake, before engaging in encrypted communication with the server. Only a Certifying Authority (CA) can issue an SSL certificate that a web browser or VIC will accept without warning you that the certificate cannot be properly validated. This means that you may use a "test" SSL certificate (available from most CAs), but will receive conspicuous warnings by both your web browser and VIC, ensuring that you cannot be accidentally "fooled" by a fake SSL certificate. Only a properly issued certificate for the correct host and domain name will be accepted silently. The domain name must usually be registered to your company and will be verified to be so by the CA.

The asymmetric keys used by SSL ensure that your VIC (only your VIC – no other VIC or other program) can decrypt the communication stream

from the server. The keys also ensure that only the server to which you are talking can decrypt any communication from your VIC. Only after secure communication has been established are your authentication credentials supplied to the server.

In addition to securing the line of communication, it is necessary to address some additional security issues related to your VTS/IS[1]:

- Remote users must have a valid user account within the VTScada applications they wish to access remotely using a VIC.

- User accounts of clients wishing to access VTScada applications remotely on a VIC must have the "Internet Client" security privilege, which allows the user to remotely view the application.

- You should consider whether you wish some pages within a standard VTScada application to be protected from specific users, through the use of application privileges.

- It is possible to create an AutoLogon account that does not require any authentication. This should only be done if you have a closed intranet with no internet access, or if you wish to provide open access to a guest account.

- If User Groups have been enabled, then only users in a group that has the same name as the VTScada Realm will be able to connect.

- If the VTScada Internet Server is a workstation that has been configured for read-only access, then all VTScada Internet Client connections to that server will have read-only access.
  Note that if you are running a multi-server environment, the read-only server should be configured in Legacy mode to prevent automatic fail-over to a server that is not read-only.

Related Information:

...Read-Only Workstations – Describes what a read-only workstation is, and how one is configured in the VTScada Admin Guide.

...Redundant SSL-Protected Servers

---

[1]VTScada Internet Server

...Privileges Related to Internet Servers – Reference. List of related security privileges.

Related Tasks:

...Generate a Request for an SSL Certificate

...Process an SSL Certificate

Generate a Request for an SSL Certificate

The SSL tab of the VTScada Internet Client/Server Setup dialog is used to request new SSL certificates for your VTS/IS.

> Note: The process will fail unless the following conditions are met:
> You must be using the same account as VTScada will use when running as a VTS/IS
> As the same user, you must then export the SSL certificate and its private key from the user certificate store to a file.

After requesting the certificate, you must then import the SSL certificate and its private key into the computer certificate store. You must be logged into your computer (not VTS) as a user with administrative rights in order to do this.

1. Ensure that VTScada is running using the same user account that it will use when running as a VTS/IS.

2. Open the VTScada Application Manager (VAM)

3. Click the Internet Client button.
   The VTScada Internet Client/Server Setup dialog opens.

4. Click the SSL Certificate tab. The dialog appears as shown.
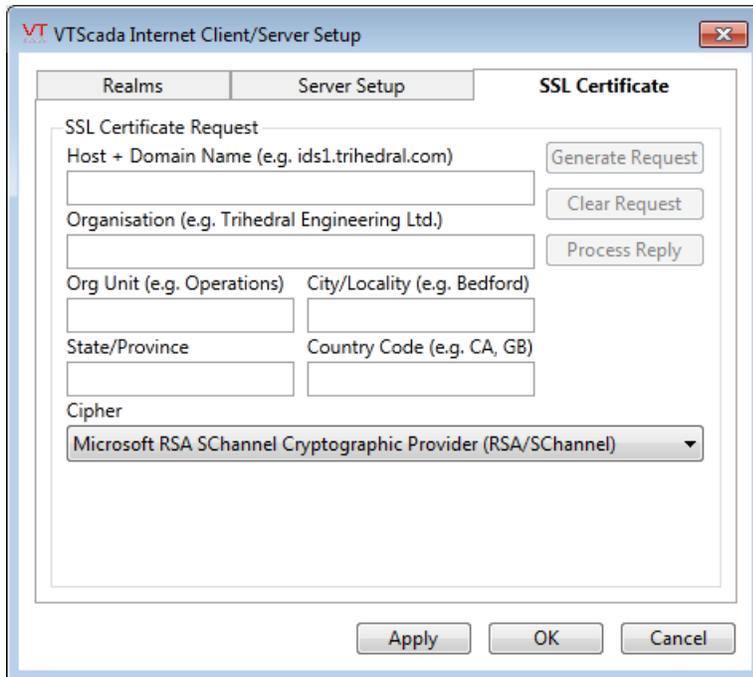
5. Enter the host and domain name in the Host + Domain Name field. The name you supply must exactly match the host + domain name supplied in the URL used to access the VTS/IS. For example:

```
myserver.trihedral.com
```

    Where, "myserver.trihedral.com" is the host + domain name, "myserver" being the host, and "trihedral.com" being the domain name.

6. Enter the name of your organization in the Organization field.

7. Fill in the remaining identifying fields.

8. Select the cipher you wish to use for cryptography from the Cipher drop-down list.
    Note that there is normally only one cipher ("Microsoft RSA SChannel Cryptographic Provider (RSA/SChannel)") available for selection, unless you have installed a custom cryptographic package.
    Once all fields have been entered, the Generate Request button becomes enabled.

9. Click the Generate Request button. VTScada compiles the data you've entered and displays the following dialog:
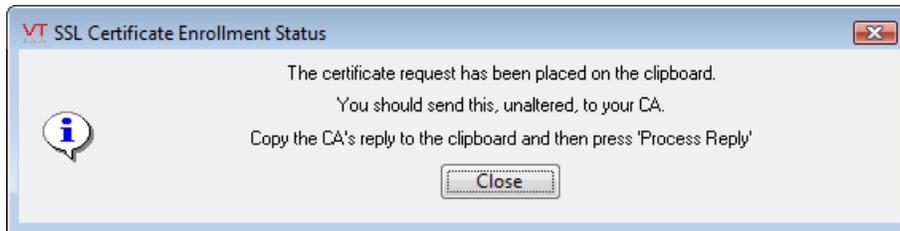
If the request failed, the Enrollment Status dialog will display an error message:

10. Open the text editor of your choice (Notepad, UltraEdit), or paste (Ctrl + V) the clipboard contents into your CA's online form (if available).

11. Select Edit | Paste (or press Ctrl + V) to paste the certificate request into your text editor. The pasted request begins with "-----BEGIN CERTIFICATE REQUEST-----" and ends with "-----END CERTIFICATE REQUEST-----".

12. Save the text file.

13. Send the text file to your Certificate Authority (such as Thawte or VeriSign). It is not essential to perform the certificate request on the same PC for which the request is being made. You can request a certificate for any PC; however, the CA's reply must be processed into the store of the PC on which the request was made.

Note: When VTScada generates the certificate request, a matching public/private key pair is generated. The public key is encoded into the certificate request and is sent to the CA. The private key is placed in a secure, encrypted key store. If you do not take a backup of your private key, a system restore operation or catastrophic computer failure will cause loss of the private key. This will require that you generate a new key and a new request. Therefore, it is highly recommended that you use the Microsoft Management Console (MMC) to backup the certificate request and private key. Once you receive your SSL certificate and subsequently process it, you should again do a backup using MMC.

Next Steps:

...Process an SSL Certificate – Use the certificate you have generated.

Process an SSL Certificate

Once you have sent an SSL certificate request to a Certificate Authority and the CA has validated your request, you will receive a reply containing the certificate. To process this information, use the following steps.
The SSL certificate will be generated in the certificate store for the logged in user (which should be the same account that the VTS/IS will use). After processing the certificate reply, the certificate (and its private key) must be exported from the user's certificate store and imported into the local computer's certificate store using the Microsoft Management Console and an account that has administrative rights.
For further information on importing and exporting certificates, please refer to Microsoft's Website: http://windowshelp.microsoft.com/Windows

1. Copy the CA's reply to the Windows clipboard by selecting the certificate including the "--------BEGIN CERTIFICATE--------" and "--------END CERTIFICATE--------" lines, and pressing Ctrl + C.

2. Open the SSL Certificate tab of the VTScada Internet Client/Server Setup dialog.

3. Click the Process Reply button.

> Note: VTScada processes the reply by removing the request from the request store and placing the SSL certificate into the VTS/IS' certificate store, binding it to the correct private key. The CA's reply must be processed into the store of the PC on which the request was made. After you've done so, you should back up the SSL certificate and copy it to the PC for which the request was made.

4. Run the Microsoft Management Console (MMC) under the user account that VTScada is running in.

   a. From the File menu, add the Certificates snap-in for "My user account".

   b. In the tree on the left, expand "Certificates – Current User"

   c. Expand "Personal" and click "Certificates".

   d. In the right-hand pane you will see the SSL certificate.

   e. Right-click on the SSL certificate and from the "All Tasks" sub-menu, select "Export".

   f. Follow the Wizard, selecting to export the private key and to delete the private key if the export is successful.

   g. Complete the wizard, exporting the certificate to a *.PFX file.

   h. If the export is successful, right-click the SSL certificate in the right-hand pane of MMC and select Delete to delete the certificate from the user account certificate store.

5. Stop VTScada and log out from your Windows account.

6. Log into Windows using an account that has administrative privileges.

7. Run the Microsoft Management Console (MMC).

   a. From the File menu, add the Certificates snap-in for "Computer account".

   b. In the tree on the left of the MMC, expand "Certificates – Current User" then right-click on "Personal".

   c. From the "All Tasks" sub-menu, select "Import".

   d. Follow the wizard selecting the certificate file that you saved earlier.

8. Locate the newly added certificate in Personal\Certificates, right-click it and pick Manage Private Keys... from the All Tasks sub-menu.

9. Click Add and add the user account you will be running VTScada as to the security list, granting Read access.

10. Run REGEDIT as administrator.

11. Navigate to HKEY_LOCAL_MACHINE\Software\Microsoft\SystemCertificates\MY.

12. Right-click the tree-node and select Permissions… from the menu.

13. Click Add and add the user account you will be running VTScada as to the security list, granting Full Control access.

14. Locate the Subject field of the certificate, noting the line: CN = YourServer.Name.Com.

15. Edit the VTScada SETUP.INI file. In the [SYSTEM] section, add *SSLCertName = YourServer.Name.Com* where YourServer.Name.Com should be replaced by the actual text from your certificate.

16. Reboot the workstation. (This step may not be required in all situations.)

17. Start VTScada and configure a secure realm.

As an alternative to setting permissions to a specific user account in MMC and REGEDIT, you can set them for a group, such as Domain Users. VTScada will now be able to access the SSL certificates and run HTTPS connections for VICs, once configured

Note: It is highly recommended that you use the Microsoft Management Console (MMC) to backup the certificate and private key. If the registry has to be restored or is lost, the private key will be permanently lost as well. The key will be encrypted to protect it during the backup process; you will therefore be required to enter a password.

Related Information:

…Configure a VTS Internet Server – See the VTScada Developer's Guide – Configuring a VTScada Internet Server? Go back to the generalized steps.

…Redundant SSL-Protected Servers

Redundant SSL-Protected Servers

If you want redundant SSL-protected VIC servers, you can use one of the following options:

- One certificate per server, in the case that each server uses a different URL. For example, if you had two servers named vtscada1.countyscada.com and vtscada2.countyscada.com, then you will need one SSL certificate per server name.
  Each server has its own SSL certificate installed and the server list contains both server names.
- You can use a wildcard SSL certificate (which costs more) and install the same certificate on both servers. Do this if you need to increase the number of servers, while continuing to use the same certificate.
- One certificate installed on multiple servers, in the case that the URL resolves to a networking device and that networking device automatically detects failure of one server and port forwards new connections to the other server or contains some sort of load balancer.

Related Tasks:

...Generate a Request for an SSL Certificate

...Process an SSL Certificate

Privileges Related to Internet Servers

VTScada has several security privileges that enables users of a VIC to view and perform limited configuration operations to VTScada applications running on a VTS/IS.

Internet Client Access privilege

Must be granted before any account or process can access the server.

Internet Client Tools Access privilege

Must be granted for users to view and work with script applications.

Internet Client Monitor Access privilege

Must be granted for users to open the internet client monitor

page.

Internet Client Monitor Admin privilege

Required for users to enable or disable connections via the Internet Client Monitor page.

Auto Logon Accounts

You have the option of creating a realm for which the user name and password are pre-entered, thus allowing immediate access to an application for all VTScada internet clients.

Note: Warning: AutoLogon accounts can potentially create a significant breach of security and should only be used in special circumstances: For example, if your application is running on a company-wide intranet with no access from the outside, you may wish to provide free access without account name and password prompts. You might also want to provide a guest account (presumably with very restricted security privileges) for general access.

AutoLogon accounts are created by entering the realm, name and password information for an existing account, into VTScada's setup.ini file. Once the AutoLogon account has been defined, you may then open a VIC connection to that realm and expect the VTScada internet server to immediately log you in as the defined user and open the application.
You may create as many AutoLogon accounts as you need by creating a realm for each. It is assumed that you have:

- An application on a licensed VTScada Internet Server.
- The application is secured and has a user account with the internet client privilege set.
- A realm has been configured to reference the application.

The following should be added to the [SYSTEM] section of the Setup.INI file in your VTScada installation folder:

```
AutoLogon_RealmName = AccountName:Password
```

where

- "AutoLogon_" is a keyword
- "RealmName" should be replaced by the name of your realm
- "AccountName" should be replaced by the name of the user account
- the colon ":" must appear as shown after the user name
- "Password" should be replaced by the user account's password

Note that changes to the SETUP.INI file will take effect only when VTScada starts, therefore you must re-start the VTScada Application Manager.

Example:

If your realm is named "WaterTreatment" the user account is named "Guest" and the password is "ABCD" then the line to add to the SETUP.INI file would be:

```
AutoLogon_WaterTreatment = Guest:ABCD
```

Note: If you are using VTScada's user groups feature, do not add the name of the group or use the group delimiter in the configuration line above. Use simply the user name, the single colon and the password.

Setup the Internet Server

Use the VTScada Internet Client/Server Setup dialog to create server lists, to identify each VTS/IS in your system, the port on which they listen, whether the port will accept unsecured connections and whether the server is local so that the correct socket listener may be started.

Note: These instructions are the same no matter how operators will connect to the application. It is the presence of the word "/mobile" or "/anywhere" in the connection address that distinguishes mobile device and Anywhere client connections from an Internet client connection.

When configuring your server, ensure that you select a port that is not in use by any other process on your server. Port 80, the default HTTP port, is often used by other programs. You can check your server by opening a command prompt as an administrator, then running "netstat -ao". You can then check the process ID value in the Windows task manager. For example:

```
C:\WINDOWS\system32>NETSTAT -AO

Active Connections
Proto Local Address Foreign Address State PID
TCP 0.0.0.0:80 MYSTATION-PC:0 LISTENING 4
TCP 0.0.0.0:81 MYSTATION-PC:0 LISTENING 10916
```

This example shows that port 80 is in use by process id #4, which belongs to Windows. Port 81 is in use by process #10916, which happens to be a configured VTScada server in this example.

Note that you must configure the same port for both the server and the realm.

1. In the VAM, click on the VTScada Internet Client/Server Setup button.
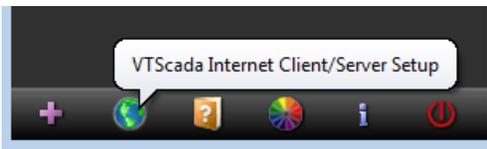
The Internet Client/Server Setup dialog opens:
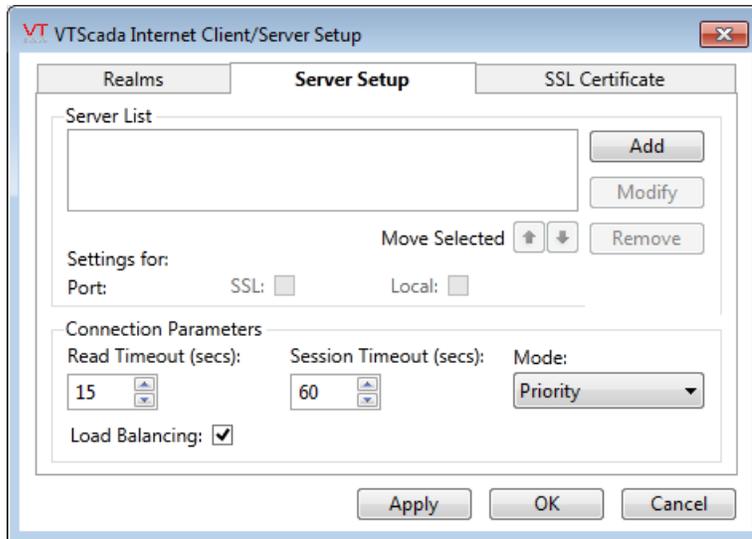
2. Click Add. The Add Server dialog opens.
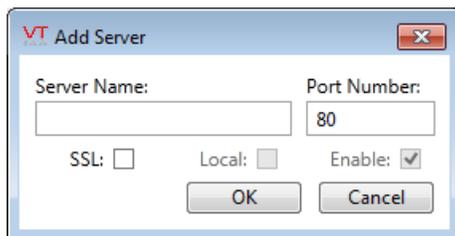
3. Enter the name of the first server you wish to add to the VTScada Internet Server list in the Server field.

The name must be one that all VICs can successfully name-resolve to an Internet Protocol (IP) address. In general, it is better to use a name than an IP address.

If you are using SSL, this must be the host + domain name, exactly as it appears in the "CN=" field on your SSL certificate.

4. Enter the port number on which you wish this server to listen in the Port Number field.

The default port number is 80, the standard Hypertext Transfer Protocol (HTTP) port.

If you are using SSL, the standard port number is 443.

5. Select the SSL check box if you wish to use the Secure Sockets Layer (SSL) protocol to secure data that is transferred over the SSL connection.

If this check box is not enabled, you will need to supply the host + domain name for the SSL certificate in VTScada's Setup.ini configuration file (located in the installation directory). Add the following line to the [SYSTEM] section:

```
SSLCertName = <host+domain>
```

where <host+domain> is the host and domain name you specified when obtaining an X.509-compliant SSL certificate. After modifying the Setup.ini file, you must stop and restart VTScada for your change to take effect.

The Local check box will (later) be automatically checked if your network has a working name resolution system, and if the server name can be resolved to an IP address owned by this computer. Local is used so that the system can activate the correct socket listener.

The completed Add Server dialog for a remote server (not the local PC) should appear similar to the one displayed here.
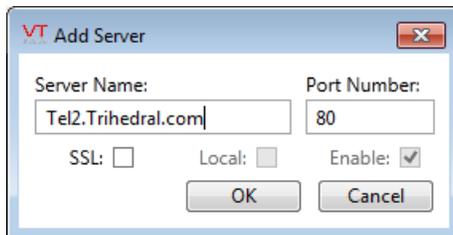
6. Click OK. The Add Server dialog closes, and you are returned to the Server Setup tab, where the server has been added to the server list.

7. Repeat steps 2 through to 6 to add as many servers as you require.

The order that the server names appear in is significant. Should a con-
nection be lost, the VIC will automatically attempt to connect to the next
server in the list.

8. Click OK to save the server list and close the VTScada Internet Client/Server
Setup dialog.

Troubleshooting:

- Port conflicts.
  You may configure a realm or a VTS/IS on any port you desire; however, if
  operating over a public network (e.g. the Internet), you will likely have to tra-
  verse firewalls and other security mechanisms. Configuring a realm or
  VTScada Internet Server to operate on other than the standard ports (port 80
  for plain text HTTP, or port 443 for SSL-secured HTTPS), will likely require
  special configuration. It is therefore advisable to operate on the standard
  ports whenever possible.

Related Information:

...Reference: Server Setup Details – Provides full details for the choices in
the configuration dialog.

...MIME Types Supported – List of types that are available to anyone who
can access your server.

...Error Dialog (Add Server)

Next steps:

...Maintain Two VIC Connections

Reference: Server Setup Details
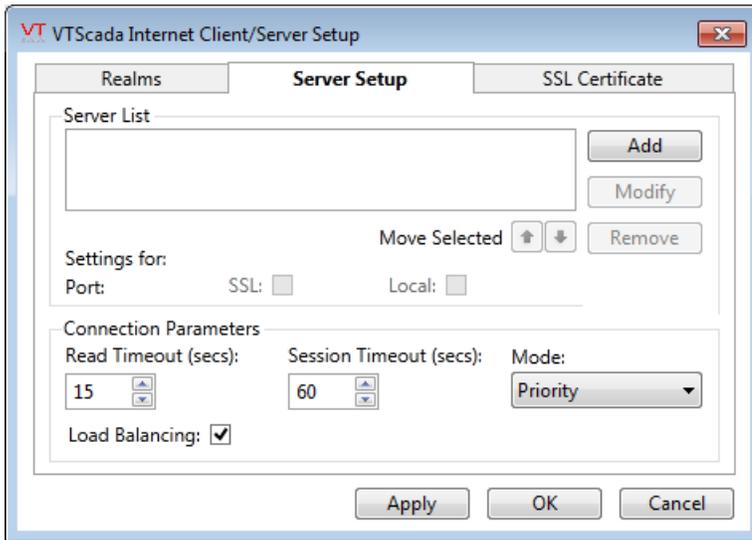
Server List

The list of servers you have added. This may be a list of aliases for one server, or it may include the other VTS/IS servers that are part of this license cluster.

The VTScada Internet Servers are listed in order of descending connection priority. Under redundant operation, when the connection is lost to the current server, it will be picked up by the next in order.

After you have successfully added a server to this list, an Enable check box will available. Use this check box to prevent new connections without needing to remove a configured server.

Add Server Button

Opens the Add Server dialog, where you can type the connection parameters for a server to add to the list. This includes name or IP address, the port, and a check box to indicate whether the server uses Secure Socket Layer (SSL) encryption.
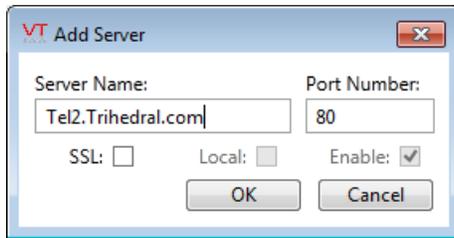
The local check box is configured automatically. If your network has a properly-configured, name-resolution system and if the server name can be resolved to an IP address owned by this computer, then Local will be checked.

If your network does not have a functioning name-resolution system, you may enable manual mode so that you are able to set the Local flag your-self. This is done by setting ManualVICServerLists to 1 in Setup.INI.

When adding a server, the Enable box is selected, and cannot be deselec-ted. After the server has been added, you may use the Modify Server dia-log (described following) to return and deselect the Enable option. By disabling a server, you can stop incoming connections without needing to remove the configuration.

Modify Server Button

Opens the Modify Server dialog, where you can change the port number of the selected server and you may choose to disable it.

The Local option can be changed only if ManualVICServerLists has been set to 1. The Enable option can be changed only if VTS is able to detect the server - this may not be the case if the name resolution system is unable to find the specified server.

To change a server's name or IP address, you must delete the old and cre-ate a new; An existing name or address cannot be changed.
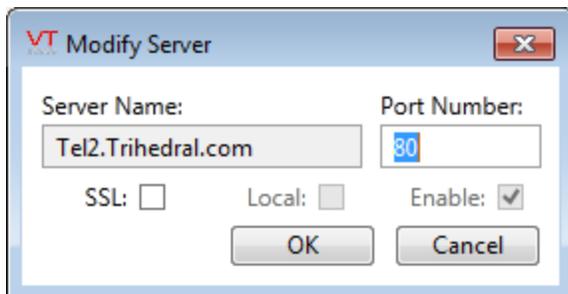
Settings for Port, SSL and Local

Port: Set the port number to use on the selected server. This will typically be port 80, but will depend on your network configuration and whether other software is already using that port. If you cannot connect to port 80, check with your systems administrator, or try other common HTTP port numbers.

SSL: Will be checked if this server uses a secure socket layer, as described in SSL Certificates.

Local: Checked if this server allows connections from itself in addition to connections from other workstations

Connection Parameters: Read Timeout, Session Timeout and Mode

These three parameters are used to control how VIC sessions will switch servers in the event that the connection is lost to the one they are connected to.

Detecting a Lost Connection – Read Timeout

Under normal operation, the VTScada Internet Client works by sending requests to a VTScada Internet Server. The server will respond with data if there is any to send, or with an empty response if there is none.

If no response is sent to the VIC by the end of the read timeout period, then the VIC detects that a server has not responded. A loss of connection is assumed.

Therefore:

- A shorter read timeout results in more rapid detection of loss of the server, at the expense of increased network traffic.
- Too short a timeout may result in false detection of server failure.

If the VIC detects that the server has not respond, it closes its connection to that server and makes one attempt at re-establishing the connection. If this fails, and the mode is not set to legacy, the VIC will attempt to connect to the next highest-ranked server available, according to the order of the server list.

A read timeout of 15 seconds is recommended as the maximum setting for a LAN, and the minimum setting for a WAN. If your network serves both LAN and WAN connections then 15 seconds is recommended. Otherwise, adjust the value up or down as needed.

Since there is always one attempt to reconnect after a timeout, the time required for the VIC detect irrecoverable loss of communication with its server is twice the read timeout value.

Orphaned Sessions – Session Timeout

Loss of a connection between the VIC and the server will result in an orphaned session taking up resources on the server. The session timeout exists to let the server know when it should recover those resources and clean up such orphaned sessions.

When the server detects that there has been no communication from the VIC for the period of time specified in the Session Timeout, it destroys all resources used by the orphaned session.

The minimum session timeout that you can specify is four times the read timeout value. An orphaned session will remain visible in the Internet Client Monitor display until the server destroys it at the end of the session timeout.

Mode of Operation

The Mode of Operation determines how the VIC will behave when:

- The VIC loses communication with its current server.
- A higher-ranked server than the one the VIC is connected to becomes available.

There are three choices for the VIC Mode; Priority, Sticky and Legacy.

Priority Mode: The VIC will always attempt to establish communication with the highest-ranked available server when loss of communication to its current server is detected.

When the VIC detects that a server of higher priority than the one it is connected to becomes available, it will connect to that server and terminate the older connection.

Sticky Mode: The VIC will always attempt to establish communication with the highest-ranked available server when loss of communication to its current server is detected.

The VIC will not switch connections if it detects that a server of higher priority than the one it is connected to becomes available.

Legacy Mode: The VIC will make no attempt to establish a connection to a different server when its current connection is lost. The VIC will not continue attempting to connect to a server if the primary is not available.

Note that the timeouts and VIC Mode are transmitted to the VIC at the start of each session; therefore the VIC will operate with whatever is configured on the server when it connects to it.

When the VIC makes its initial connection, it simultaneously attempts to connect to each server in the list and uses the server nearest the top of the list that it can successfully establish a session with. The VIC waits for an initial period of time, or until the highest-order server responds – whichever is shorter – then it selects and connects to a server. This initial timeout is left up to the TCP subsystem of the client computer and is normally around 20 seconds.

If the mode is set to Priority or Sticky, then if there are no servers available, the VIC will continue to try to connect to each server. In Legacy mode, the VIC will not make continuous attempts.

Load Balancing

This box is checked by default, but relevant only if you have more than one VTScada Internet Server for an application. When checked, new VIC connections will automatically go to whichever server in a cluster has the fewest connections.

Related Information:

...License Clusters – Maintaining more than one VTScada Internet Server for your application.

...See: "ManualVICServerLists" in the VTScada Admin Guide. Enables manual configuration of each server within a VTS/IS cluster.

Maintain Two VIC Connections

Given an installation with two VTScada Internet Servers, you may wish to be able to maintain connections to both servers.
This can be done as follows:

1. Set ManualVICServerLists = 1 in the Remote section of the file Setup.ini, located in the VTScada installation folder.

2. Set the VIC Server Mode to "Priority".

3. On each of the two servers, define the server lists such that the order of primary and backup is reversed on each server. (see example following). This is possible since ManualVICServerLists was set to TRUE in step 1.

4. Uncheck the Load Balancing flag in the server setup.

Example.
Given the following two VTScada Internet Servers:

     TelServer1
     TelServer2

On TelServer1, the server list is:

     TelServer1
     TelServer2

And, on TelServer2, the server list is:

     TelServer2
     TelServer1

The server lists are not shared between servers (ManualVICServerLists is set to 1), therefore they can be different. The Priority Mode setting of the VIC Server Mode then gives the functionality required.

MIME Types Supported

In addition to allowing clients to remotely monitor VTScada applications, the VTScada Internet Server will also send files of the following MIME types to a client:

- MIME type: text/html.
  Hyper Text Markup Language. Both .HTML and .HTM file extensions supported.
- MIME type: application/JavaScript
  JavaScript content – .js file extension
- MIME type: text/xml
  Extensible Markup Language – .XML file extension

For example, if you have a configured VTS/IS[1] and any of your application folders contain an HTML[2] file, the VTS/IS will serve that page to any client that requests it. Examples of URLs to access these files from a VTS/IS can be found in the topic: URL Interpretation.

Clients attempting to access files of any type other than those specified in the preceding list will see an "Access denied" message rather than the file. This means that the VTScada Internet Server will not distribute your application's .SRC, .INI or other files.

Warning: Files with one of the supported MIME type extensions, (.HTML, .HTM, .JS and .XML) are served without authentication. They are available to everyone who has Internet access to your server.

Do not store sensitive information in a file having one of the listed extensions.

Error Dialog (Add Server)

You may configure the same VTS/IS on multiple ports, and you may configure different VTS/IS on the same port; however, should you

---

[1]VTScada Internet Server

[2]Hyper Text Markup Language. Used for web pages.

inadvertently attempt to duplicate the name and port of a VTS/IS that has already been added to the VTS/IS list, the Add Server error dialog opens.
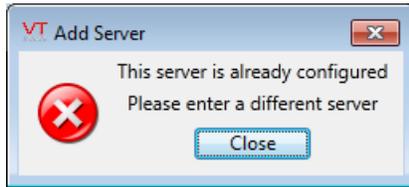
Click the Close button to return to the Add Server dialog, and add a different port number for the server.

Internet Realms

A realm in VTScada has two aspects:

- It is a name given to configuration options that include the connection protocol (HTTP) and port (usually 80). Without a realm there will be no port for Internet communications.
- It is a group of applications that are logically associated with one another for the purpose of authenticated access from a web browser. Without a group of applications, there will be nothing to connect to on the configured port.

Realms are required and used by:

- VTS/IS operations including VIC and Mobile Internet Client connections.
- ODBC Interface to VTScada History
- Web services via SOAP

Any number of realms can be created, and any application can be placed into one or more realms. When connecting to an application, the name of the realm is included as part of the connection URL[1].

Group Logons and Realms

If you are using security groups and realm-area filtering, then you must create a realm having the same name as each group. Operators who would normally logon using their group name, account name, and

---

[1]Uniform Resource Locator. The address of a web page.

password will instead open a URL having a realm that matches the group name and log on using just their account name and password. They will not be allowed to connect to any other realm.

Super users, who are not members of any group , will not be able to log on over the Internet unless the application property RootNamespace has been added, and its value set to the name of a realm created for the use of these accounts.

Related Information:

...Reference: Realm Setup Details – Description and instructions for the realm configuration tools.

...Test the Connection – Test the connection / Obtain and automatically-built URL.

...Realm Display Setup Tags – Set window configuration and default page on a per-realm basis.

...See also, RootNamespace in the VTScada Admin Guide

Related Tasks:

...Configure a Realm

Configure a Realm

Note: If you plan to allow programmers and developers to access diagnostic applications such as the Source Debugger or TraceViewer over an Internet connection, then you must add those applications to the realm.

These diagnostic applications do not store user accounts, therefore to be accessed over a VTScada Internet Connection, they must share a realm with at least one secured application that has a user account with the Internet Client privilege. That account can be used to connect to diagnostic applications in the same realm.

To provide access to an application, you must associate it with a Realm name.

1. Open the Internet Setup dialog, from the VAM[1].

2. Ensure that the Realms tab is selected.
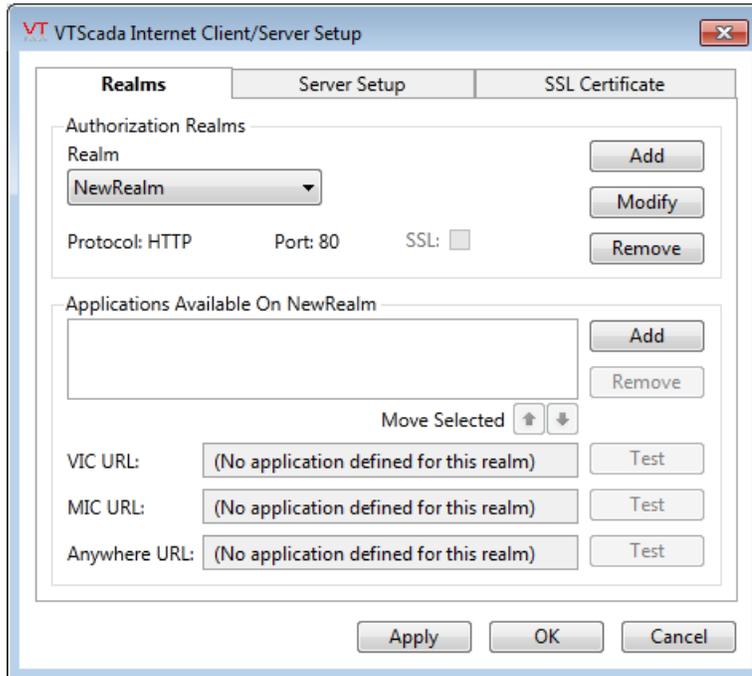
3. Click Add in the Authorization Realms section of the dialog.
   The Add Realm dialog opens.
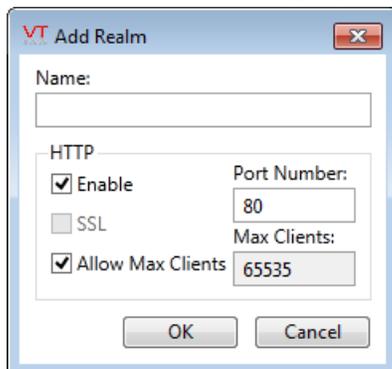
4. Enter a meaningful name for the realm in the Realm Name field.
   Realm names should not include spaces. Use a hyphen, underscore or mixed case to indicate word boundaries (e.g. "My-Realm", "My_Realm" or "MyRealm").

5. Enable the HTTP protocol.

6. Enter the port number you wish VTS/IS to use.

---

[1]VTScada Application Manager

The default port number is 80, the standard Hypertext Transfer Protocol (HTTP) port. If you are using SSL, you must first have obtained an SSL certificate (see SSL Certificates), and installed it. Supply the SSL standard port number of 443, in this case, and check the SSL check box. If this check box is disabled, you need to supply the host + domain name for the SSL certificate in VTScada's Setup.ini configuration file (located in the installation directory). Add the following line to the [SYSTEM] section:

SSLCertName = <host+domain>

where <host+domain> is the host and domain name you specified when obtaining an X.509-compliant SSL certificate. (Do not include the angle brackets.) This must exactly match the "CN=" field of your SSL certificate. After modifying the Setup.ini configuration file, you must stop and restart VTScada for your change to take effect.

If operating over a public network (e.g. the Internet), you will likely have to traverse firewalls and other security mechanisms. Configuring a realm or VTS/IS to operate on other than the standard ports (port 80 for plain text HTTP, or port 443 for SSL-secured HTTPS), will likely require special configuration of such interposing security mechanisms. It is therefore advisable to operate on the standard ports whenever possible.

7. Click OK.

The new realm is created, and you are returned to the VTScada Internet Client/Server Setup dialog where the new realm appears in the Realm drop-down list. This will look similar to the following example:
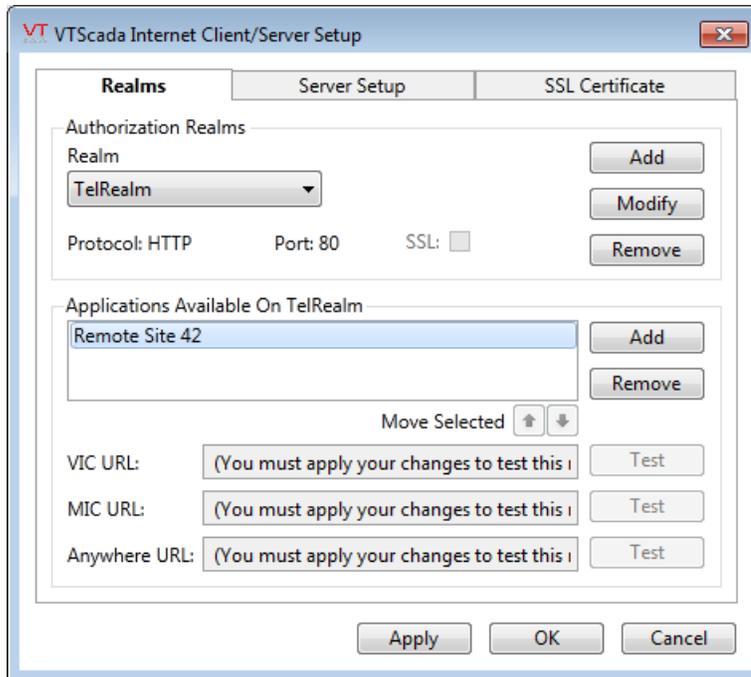
8. Click Add in the Applications Available On section of the VTScada Internet Client/Server Setup dialog.

   The Add Application dialog opens as shown.
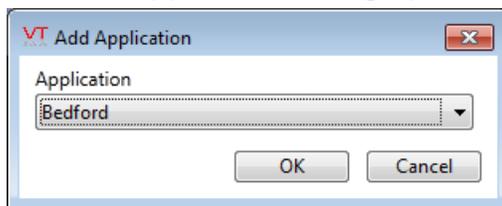
9. Select the VTScada application you wish to add to this realm from the Application drop-down list.

Note: The first VTScada application added to this realm is the default application for the realm. A connection is automatically attempted to the default application if a partially specified URL is provided to your web browser. For example:

   http://myserver.trihedral.com/myrealm

is a partially specified URL, containing only the protocol (http), host (myserver), domain name (trihedral.com), and realm (myrealm).

10. Click OK.

You are returned to the VTScada Internet Client/Server Setup dialog where the selected application has been added to the realm.

11. Repeat steps 8 through 10 to add as many VTScada applications (both standard and script applications) as you wish for this realm.
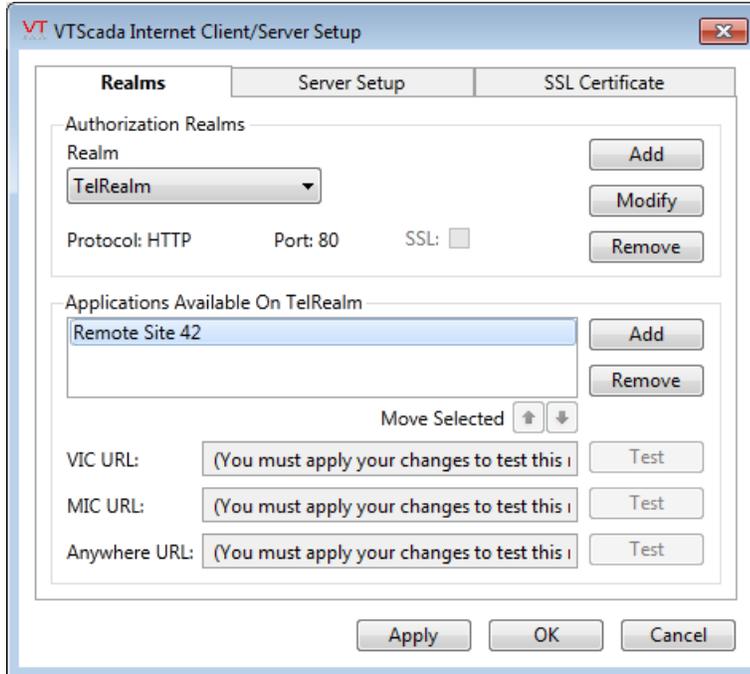
12. Click OK.

Your application is now available to VTScada Internet clients.

Troubleshooting:

- Unable to connect.
  Check that no other service is using the configured port.
  If trying to connect using the server computer, check that the Local option has been checked (done automatically). If not, it is likely that the domain is not being recognized.
  If trying to connect remotely, check that the server is visible on the network. Firewall or proxy server configuration may be required.
  Check that the VTS Internet Server configuration has been completed correctly.
  Check that security has been enabled in the application, and that your account has the Internet Client Access privilege.

Reference: Realm Setup Details

Details of, and instructions for, the components of the Realms tab of the VTScada Internet/Client Server Setup dialog.

Realm List

Displays all realms. You may select any realm, in order to edit its configuration properties. As each realm is selected, its configured protocol and port is displayed in the space below the list.
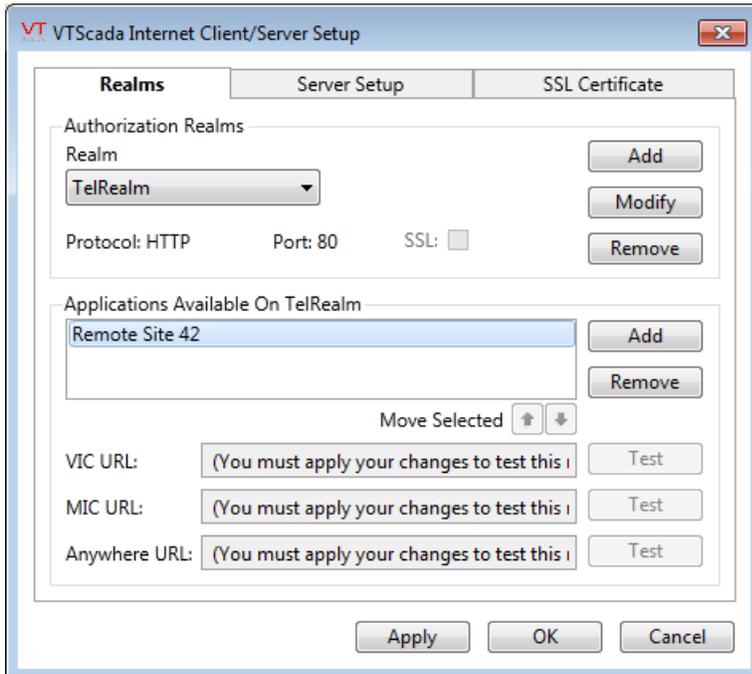
Add (realm)

Opens the Add Realm dialog where you can configure a new realm.
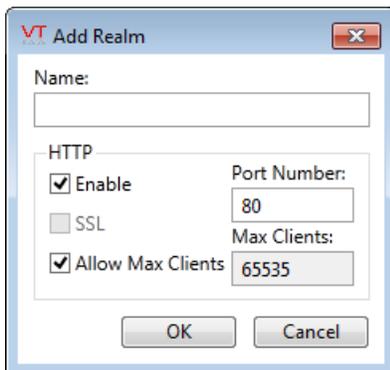
It is recommended that the name be a single word and not contain punctuation symbols. The name must conform to both VTScada and HTTP naming conventions, therefore the simpler, the better.

- **HTTP Enable**. This option might not be selected for every case. You might want to create a realm but not grant immediate access to it.
- **Port Number**. This must match one of the ports you configured in the Server Setup tab.
- **SSL** This check box will be enabled only if SSL security has been configured.
- **Max Clients**. Use this field to balance your client licenses between configured realms. For example, if you have 5 licenses, you might wish to ensure that no realm can have more than three clients, thus leaving at least 2 licenses available for the other realm. See VTS/IS License Clusters for more information on managing your licenses.

Modify (Realm)

Opens a window that is identical to the Add Realm dialog except that you cannot change the name of a realm. Rather than change a realm's name, you must create a new one and delete the old.

Remove (Realm)

Removes the selected realm (and the list of applications attached to it) from the list.
You will be prompted to confirm this action before the realm is removed.
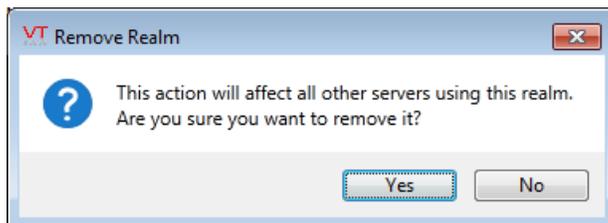
In the case that you have more than one internet server, the realm will be removed from all. Any users who are connected to the realm will be disconnected immediately.

Protocol, Port and SSL Check boxes

Shows the configured protocol (HTTP) and the port number (usually 80) for the selected realm.

If you have configured the Secure Sockets Layer protocol to encrypt data that is transferred over the SSL connection, then you can enable or disable that protection for this realm using the SSL check box.

Applications Available On

Shows the list of VTScada applications that have been can be accessed using the current realm name. Note that script applications can be accessed only if the list also contains at least one standard application.

Move Selected (Application in Realm List)

The Move Selected arrows enable you to change the position of the selected VTScada application in the Applications Available On list. The order of the applications is significant since the first application in the Applications Available On list is the default application and controls the account name and password that operators must use when connecting to a realm.

Add (Application to Realm)

Opens the Add Application dialog, which is used to select VTScada applications to be added to the selected realm.
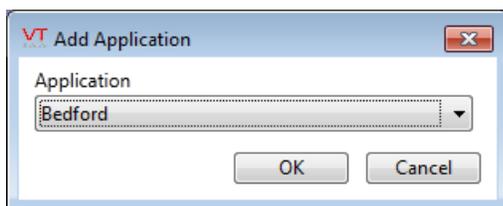
Remove (Application from Realm)

The Remove button enables you to remove the selected application from the current realm.

Note: in versions of VTScada prior to 10.1.06, removing the last application from a realm's list would delete the realm. Since that version, realms can be deleted only by using the Remove (realm) button.

Test (VIC and MIC)

Having successfully configured a VTScada Internet Server, and a Realm, you may test your work directly from the Realm configuration dialog. Note that, before you can rest a connection, you must save your work by using the Apply button (or by clicking OK, then returning to this dialog) Requirements before opening a test connection:

- It must be possible to create a local connection to the VTScada Internet Server.

If the "Local" box option is not selected in the Server Setup page of this dialog, you should review your Name Resolution service. If there is no network connection, this might be done by editing the HOSTS file in your C:\Windows\System32\Drv\ folder to configure the IP address for your computer name.

- Internet Explorer must be installed on this computer for the VIC URL test. The MIC URL test will open whichever browser is your default, and is capable of displaying Mobile connections.
- The application must be running, security must be enabled, and there must be at least one security account, having the VTScada Internet Client privilege.

The URL portion of these fields may be useful even if one of the preceding conditions is not met. This provides the complete URL to use when testing the connection locally.

Related Information:

...License Clusters

Test the Connection

Having successfully configured a VTScada Internet Server, and a Realm, you may test your work directly from the Realm configuration dialog. Note that, before you can rest a connection, you must save your work by using the Apply button (or by clicking OK, then returning to this dialog) Requirements before opening a test connection:

- It must be possible to create a local connection to the VTScada Internet Server.
  If the "Local" box option is not selected in the Server Setup page of this dialog, you should review your Name Resolution service. If there is no network connection, this might be done by editing the HOSTS file in your C:\Windows\System32\Drv\ folder to configure the IP address for your computer name.

- Internet Explorer must be installed on this computer for the VIC URL test. The MIC URL test will open whichever browser is your default, and is capable of displaying Mobile connections.

- The application must be running, security must be enabled, and there must be at least one security account, having the VTScada Internet Client privilege.

The URL portion of these fields may be useful even if one of the preceding conditions is not met. This provides the complete URL to use when testing the connection locally. The automatically-built URL will always be based on the first server configured in the Server Setup page of the dialog.

Anywhere Client

URL format, assuming that you have enabled SSL security:

   https://YourCompany.com/RealmName/anywhere

This is one of three connection types that are available. See also, MIC Mobile Device Clients and VIC Internet Client Configuration.

The VTScada Anywhere Client is a JavaScript-based technology that enables you to view and operate your application on any device, using any of the major browsers. The Anywhere Client has been tested on Windows, iOS, OS X, and Android operating systems and will work using Chrome, Internet Explorer version 11 or later, Edge, FireFox, Opera and Safari browsers.

(Not all combinations of operating systems and browsers have been tested. The Anywhere Client will not work on IE 10 or earlier.)

The client is zero-footprint, meaning that no code and no applications are installed on your computer. Everything runs within the browser.

The Anywhere Client enables you to store your logon credentials temporarily using the Remember Me check box on the logon screen. This will remain in your browser's cache for the time specified by the application property, RememberLoginDuration. This defaults to 20160 minutes (14 days).

The Anywhere Client logon screen, showing the Remember Me option.

The user interface is nearly identical to a full VTScada workstation install-
ation, allowing for minor differences in font and window details that may
occur from one browser to another. All the features of a run-time
VTScada installation are enabled with the exception of:

- The Select Color dialog has limited functionality. The color-picker eye-
  dropper tool will not work.

- No client-side reading or writing, therefore...

  - Not able to export either reports or HDV data to files.

  - Not able to create or apply a ChangeSet.

  - Not able to manage types.

  - Not able to export/sync tags.

  - No access to the Lexicon dialog.

- Internet Explorer version 11 does not support sound with the Anywhere Cli-
  ent.
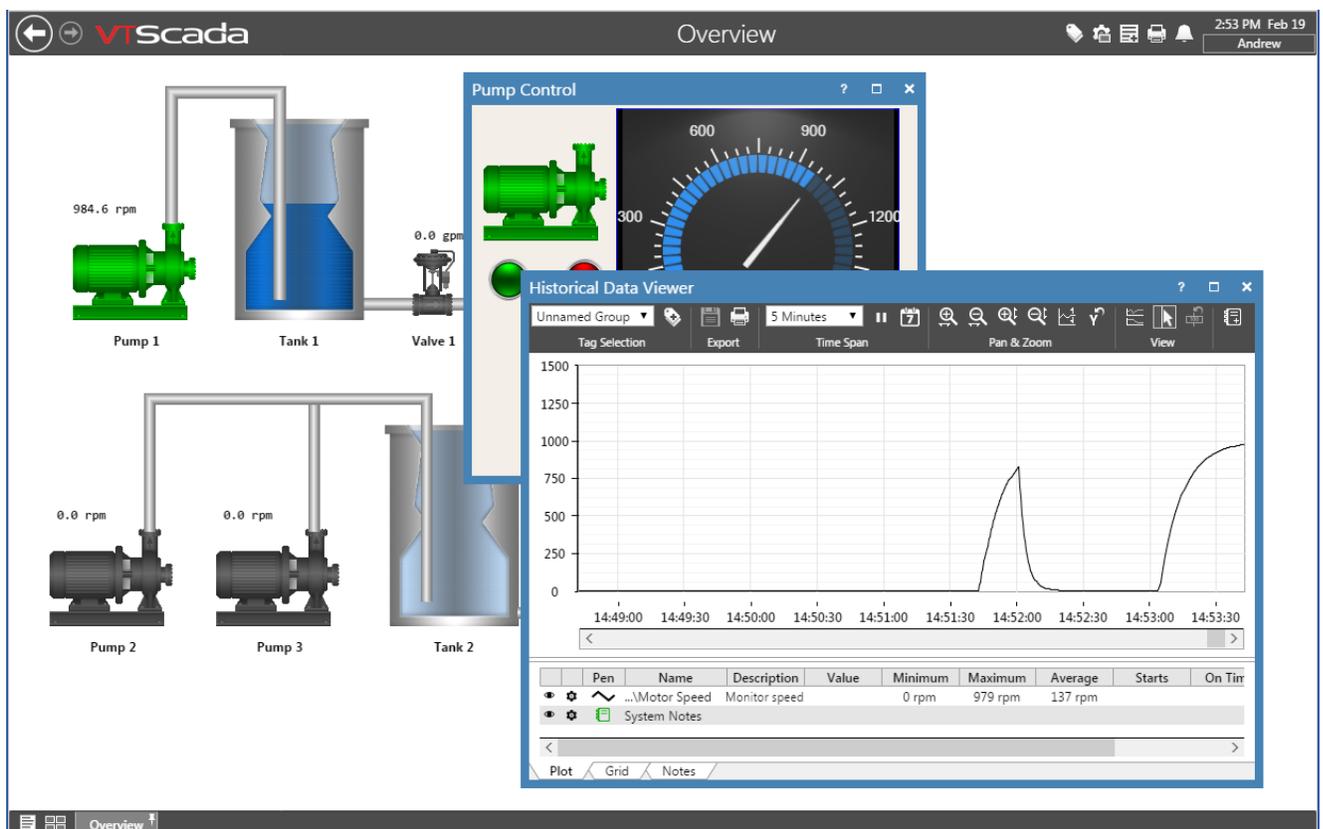
*The Completed Tutorial application, with a pop-up window and a trend, running in
Chrome using the Anywhere Client.*

Using domain aliases with the Anywhere Client:

An uncommon configuration option is to provide users with a single URL that, depending on which of the VTScada Internet Servers are presently running, will automatically refer to one or the another server. This configuration of a single URL with failover is sometimes done by hosted solution providers. It enables them to take one or the other server offline without needing to notify customers that they must use a backup URL pointing to the backup server.

As an example, given the following VTScada Internet Server list:

- primaryserver.com
- backupserver.com

If customers connect to the Anywhere Client using https://primaryserver.com or https://backupserver.com, then no additional configuration is necessary beyond what is normally done for configuring VIC servers.

However, if the customers use a connection URL such as, https://myservice.com (that is, one that is not in the VIC server list), then the following configuration is required:

On every VTScada Internet Server, add "myservice.com = 1" to the [Clients-AdditionalAllowedOrigins] section of Setup.INI.

Programmer's Notes for the Anywhere Client

Due to limitations of what can be done in a web browser, the following features will not work in the Anywhere Client. These notes are of interest primarily to VTScada programmers, and much of the following will have meaning only for advanced VTScada programmers.

- It is not possible to read or write files on the client using "session-aware" FileStreams. Calling FileStream with the Flags set so as to be session-aware causes FileStream to do nothing and return Invalid.
- The following functions do nothing in Anywhere client sessions:
    - FileDialogBox,
    - PrintDialogBox,
    - FontDialog,

- WindowSnapshot,
- ZEditField,
- Text,
- Beep,
- Table,
- Line,
- Pipe,
- Spawn,
- PrtScr,
- ActiveX

- The following WindowOptions that work on the VIC will not work on the Anywhere Client: 9, 10, 11, 14.

- WinToolTipCtrl only supports style bit 128 within the Anywhere Client. All others are ignored.

- GUIButtons with rotation, or sides not equal to 0 or 4, are not displayed.

- Changes to Application Window Behavior and Application Window Size settings do not take effect on Anywhere clients until they restart.

- "Enable Logged Off VIC Sessions" has no effect on Anywhere clients. They disconnect on logout.

- WinShiftKeys only reports the state of Shift, Ctrl, and Alt on Anywhere clients. Those keys plus the Caps, Num and Scroll lock keys work on VICs.

- Pages may contain regions that must respond independently to pick, touch and panning messages. For example, a Sites page containing a map. The Pick() function has options to suppress horizontal and vertical panning events over a region.

- All resizable Windows with scroll bars disabled (those with bits 2 and three set to zero and bit 8 set to one) will have unlimited maximum size when running on the Anywhere client. This will affect the virtual size reported by the VStatus tool.

Related Information:

...See: "IsVICSession" in the VTScada Admin Guide.

MIC Mobile Device Clients

URL format, assuming that you have enabled SSL security:

   https://YourCompany.com/RealmName/mobile

This is one of three connection types that are available. See also,VIC Internet Client Configuration and Anywhere Client.

VTScada provides a Mobile Internet Client (MIC) feature, enabling access to your application using a mobile device. It is the presence of the word "/Mobile" in the address that distinguishes the mobile connection from the Internet client connection. *No extra configuration is required, beyond that required to support the VIC*[1].

> Note: Note: If your pages include tabbed folders, then the Unique Key property of the folder must be set in order for tabs to function when viewed using a Mobile Internet Client.

Mobile platforms supported include the following (note that JavaScript must be enabled in all cases).

- iOS (iPhone and iPad) (iOS 4.3.5+)
- Android 2.1+
- Blackberry 6.0+

You can also access the "mobile view" from desktop browsers (*) such as:

- Firefox 3.6+
- Opera 11+
- Internet Explorer 9+
- Safari 5.1+

You can view and acknowledge alarms, view and operate I/O controls on a page-by-page basis, and view a trend graph (HDV) for any numeric tag that can be plotted. The HDV display supports user-interface features such swipe and pinch. If your application makes use of slippy maps, those will also be available on the mobile device.

---

[1]VTScada Internet Client. Allows you to connect to an application over the Internet with many of the features of a full VTScada workstation.

(*) Use of the IP address rather than the computer name may be required by some mobile devices when connecting, particularly if the server is on an internal network.

> Note: There is no "Logout" button in the mobile interface. Clients are automatically logged out and their session ended when they have been inactive for the time specified by IdleWebSessionTimeout.

> Note: Realm Display Setup tags do not apply to mobile device connections.

Related Information:

...Securing a VTScada Internet Server – Prevent unauthorized access.

...VIC Internet Client Configuration

...See: "IdleWebSessionTimeout" in the VTScada Admin Guide

Related Tasks:

...See: Connect Using a Mobile Device in the VTScada Operator's Guide.

VIC Internet Client Configuration

URL format, assuming that you have enabled SSL security:
   https://YourCompany.com/RealmName
This is one of three connection types that are available. See also, MIC Mobile Device Clients and Anywhere Client.
A VTScada Internet Client (VIC) is a computer that connects to an application on a VTScada Internet Server (VTS/IS). The connection is done over a TCP/IP network.
Trihedral provides an ActiveX control, with which operators can use the application as if at a workstation with a run-only license. Using this control, you may view applications that are running on the VTS/IS, operating controls and acknowledging alarms. It is assumed that security has been enabled in the application, that you have a user account with the Internet

Client Access privilege, and that the VTScada Internet Server has been configured.

VIC connections can be monitored and diagnostic information logged, using the VIC Monitor application. Diagnostic logging is described in the list of user interface elements.

Application development on a VIC is limited to adding new tags, copying existing tags, modifying the properties of a tag, deleting a tag, modifying options and pronunciation, and working with features of the Application Configuration dialog. The Idea Studio cannot be opened on an Internet or mobile client connection.

Set VIC Display Options

VIC[1] workstations have the ability to control their screen layout independently of the VTScada Server. The display options that will be available to the VIC are granted at the server by way of one or more Realm Display Setup tags added to the application. Using one or more tags embedded within the application provides you with a fine level of control over how the application may be displayed on the VIC.

Example:

In a given application, some VIC stations might be desktop workstations while others might be laptops with smaller screens. Operators at each type of station could connect to the application via a different configured realm and thereby have access to the right display settings for their screen.

If a Realm Display Setup tag is not added to an application, then the client will use the display settings from the VTScada Internet Server.

Related Information:

...Securing a VTScada Internet Server – Prevent unauthorized access.

---

[1]VTScada Internet Client. Allows you to connect to an application over the Internet with many of the features of a full VTScada workstation.

...Capabilities of Internet Clients – Reference for what tasks may and may not be done from a VIC.

...Realm Display Setup Tags – Set window configuration and default page on a per-realm basis.

...Client Requirements – Hardware and software.

...Reference: The VIC Status Dialog

Related Tasks:

...Connect Using a Browser

...Connect Without Internet Explorer

Client Requirements

> Note: The use of Internet Explorer as a client is valid only for Windows 8® and earlier. After version 8, use only the stand-alone VTSX client. See:
> Connect using the VTSX Program in the VTScada Operator's Guide.

The ActiveX component has a very small footprint and will run adequately on older computers. The recommended minimum operating system is Windows Vista. Windows XP can be used, but substitutions will be made for certain display features that are not native to that operating system. Macintosh and Linux operating systems are not supported.
The following requirements apply only if you intend to use Internet Explorer to download and run the ActiveX control. Your other option is to run the VTSX.exe program, distributed with VTScada.

- Microsoft Internet Explorer version 8.0 or higher;
- Logged in to Windows with an account that is permitted to download and install signed ActiveX components.
  Note: Use the 32-bit version of Internet Explorer rather than the 64-bit version for your VIC connections. Relevant technical reasons can be found by doing a web search for "Installing 32-bit IE 9 on 64-bit Windows".
- JavaScript enabled;
- It is advisable to enable HTTP 1.1 within Microsoft Internet Explorer.

Internet clients do not require the VTScada software to be installed.

The ActiveX component is equally able to run in both LAN and WAN environments, and is particularly suited to business-use computers on which the user has a desire to periodically monitor the state of an industrial process.

Related Information:

...Server Requirements and Licensing

Related tasks:

...Connect Using a Browser

...Connect Without Internet Explorer

Connect Using a Browser

These steps assume that the VTS/IS[1] has been configured. Only Internet Explorer provides the ActiveX support required. If clients attempt to connect using another browser, they will be given the option of downloading the installer for the VTSX program, or creating a mobile client connection.

1. Ensure that your copy of Internet Explorer has been configured to permit ActiveX components to run.
   You may have to define a security exception to permit this control to run.

2. Direct Internet Explorer to a URL[2] relative to the VTS/IS.
   Once the client has successfully connected to the VTS/IS, the ActiveX control is downloaded, and the client becomes a VIC.

If the server has a newer version of the ActiveX control than that installed on the VIC, the newer version will automatically be downloaded to the VIC and installed.

Enable HTTP 1.1 in Internet Explorer

It is advisable when using Internet Explorer for your VTScada Internet Client, that you have HTTP 1.1 enabled in your browser settings

---

[1]VTScada Internet Server

[2]Uniform Resource Locator. The address of a web page.

1. Run Microsoft Internet Explorer.
2. Select Internet Options from the Tools menu. The Internet Options dialog opens.
3. Click the Advanced tab.
4. Scroll through the list until you find "HTTP 1.1 Settings".
5. Select the "Use HTTP 1.1" check box.
6. If the web browser connection must traverse a firewall via proxies, performance may be severely compromised unless the proxies are HTTP 1.1 compliant. If such proxies are HTTP 1.1 compliant, select the "Use HTTP 1.1 through proxy connections" check box. If such communication via proxies is only performed using SSL, the setting of this check box is irrelevant.
7. Click Apply.
8. Click OK.

Troubleshooting:

- No connection
  Ensure that the server is running, and properly configured.
  Ensure that you are able to connect to (ping) the server. If connecting locally, ensure that the "local" box is checked on the VTS/IS configuration. If not, the domain name might not be properly registered on the server.
  Ensure that you have included "http://" or "https"//" as the beginning of the server address.
  Add the realm name to the connection parameters, if you have not already done so.

Related Information:

...Configure a VTScada Internet Server

...Connect Without Internet Explorer

...Reference: The VIC Status Dialog

Connect Without Internet Explorer

You can install and run the ActiveX control without using Internet Explorer. To do so, you will need to install the control on each workstation that is going to connect to the VTScada/IS.

> Note: If an operator attempts to connect to a VTScada internet server using any browser other than Internet Explorer, they will be given the opportunity to download and install the ActiveX control from the server.

1. Locate the file, VICSetup.EXE.
    This is distributed with the VTScada installation program. Clients attempting to connect to a realm using a browser other than Internet Explorer® will be given an opportunity to download it.
2. Run this program.
    The installation wizard will start.
3. Click on the "Next" button.
    The License Agreement page opens.
4. Select the option, I Agree, then click on Next
    The Destination Location dialog opens.
5. Change the location if you wish, or accept the default of C:\Trihedral.
6. Click, "Next".
    The Start Installation page appears.
7. Click on Next
    The installation proceeds.
8. Click, "Finish".

This will install the program file, VTSX.EXE in the directory you specified. (e.g. C:\Program Files\Trihedral\VTSX.EXE)

Note that you will need to find and run it from there on first use.

Users will find it easier to locate and run VTSX.EXE if you create a Windows™ shortcut to it, but this is best done from inside the running VTSX program. The "Save As.." option provided there, will allow you to save connection information with the shortcut, thereby making it easy for the operator to return to the application.

The example shows the VTSX program running with the Advanced option expanded. This initial screen is used to connect and login to the application.

Server Address

The full URL to the VTS Internet Server. Note that this must include the "http" or "https" prefix.

The realm name must not be included as part of the server address.

In the event that your server uses a port other than 80, you should specify that as part of the address. Example: http://vt–s.trihedral.com:81

Username

Your account name for the application. The account must have the Internet Client Access privilege.

Password

Your password for that account in the application.

Realm

The name of the realm that includes the application. If not specified, you will be given a choice of which configured realm to use.

Page Name

The name of the page that should be shown first upon connecting. Use this to select a page other than the configured default.

Parameters

If the page you are connecting to is parametrized, you should use this field to provide values for those parameters.

Troubleshooting:

- No connection
  Ensure that the server is running, and properly configured.
  Ensure that you are able to connect to (ping) the server. If connecting locally, ensure that the "local" box is checked on the VTS/IS configuration. If not, the domain name might not be properly registered on the server.
  Ensure that you have included "http://" or "https"//" as the beginning of the server address.
  Add the realm name to the connection parameters, if you have not already done so.

Related Information:

...Connect Using a Browser

Reference: The VIC Status Dialog

Whenever the VIC is not connected to a server, a status dialog is displayed, informing you what the VIC is attempting to do:
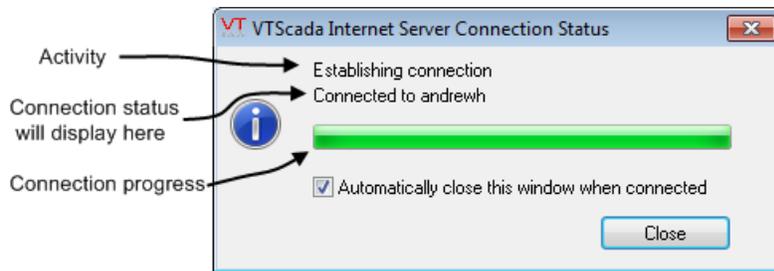
The activity text gives the reason that the dialog has been displayed. It can be one of the following:

- Establishing initial connection –the VIC is attempting to establish the initial connection to a VTScada Internet Server.
- Connection to server <servername> lost – the VIC has lost connection to its current server.
- Higher priority server available – the VIC was maintaining communication with its current server when a higher-ranked server became available. This is only displayed in priority mode.
- Application stopped on server <servername> – the operator stopped the application on the VTScada Internet Server, but VTScada continued to run.
- Authentication failure – During a VIC session, the credentials of the user were changed and are no longer valid. This is a terminal condition.
- Session logged out – During a VIC session, the user manually logged out or the SecurityManager inactivity timeout expired, causing the user to be forcibly logged out. This is a terminal condition.
- Session terminated by operator – During a VIC session, the operator forcibly terminated the session, via the ICM. This is a terminal condition.
- Change-over forced by operator – During a VIC session, the operator forced the VIC to changeover to its next best server.

The Connection Status Text tells you what the VIC is attempting to do. It can be one of the following:

- Blank – The connection process has recently begun and the message has not yet been updated
- Connecting to <servername> – the VIC is in the process of connecting to the named server.
- Connected to <servername> – the VIC is now connected to the named server.

- Waiting for an available server – none of the servers are contactable by the VIC. The VIC will continue to try to establish a connection.
- Unable to continue. Please exit program. – a terminal condition has arisen and the VIC cannot continue.

The Status Icon can be one of three pictures:

- An information icon – the status dialog is simply telling you something. The VIC is operating normally

- A warning icon – the status dialog is advising you of a non-fatal (yet) problem.

- An error icon – the status dialog is advising you of a serious problem, usually meaning that the VIC cannot operate.

- The Close Button will dismiss the dialog. It can have one of two texts:
    - Close – Clicking the button will cause the dialog to simply close.
    - Exit Program – Clicking the button will cause the VIC to terminate. Clicking the X button on the title bar of the status dialog has the same effect as clicking the Close Button.

- The Activity Indicator animates when the VIC is attempting to establish a session with a server.

- The Auto-Close check box, when checked, causes the dialog to dismiss automatically when the VIC is connected to a server. The dialog will always appear when the VIC is not connected to a server. The state of the check box is persisted in the registry and so is remembered even if a newer version of the VIC is downloaded from a VTScada Internet Server

Customize the VIC Installation Screen

While a VIC[1] is downloading the ActiveX component from a VTS/IS[2], a component-loading window is displayed.

---

[1]VTScada Internet Client. Allows you to connect to an application over the Internet with many of the features of a full VTScada workstation.
[2]VTScada Internet Server

VTScada Internet Client Installation

In order to run the VTScada Internet Client, you need to install our ActiveX viewer. If you are prompted "Do you want to install this software?" then you must permit the installation in order to view the requested page.
For your security our ActiveX control is fully signed.

The VTScada Internet Client ActiveX viewer is being downloaded.

For more information see: www.trihedral.com

This is an HTML[1] page, provided with VTScada. You may edit the file as required, provided you are familiar with HTML.

To create your own message:

1. Navigate to the VTScada installation directory.

2. Locate the "VTS.htm" file.

3. Copy the "VTS.htm" file.

4. Rename the copy "OEM.htm". VTScada is programmed to recognize a file by this name and loads it rather than the "VTS.htm" file if it is present in the VTScada installation directory.

5. Modify the "OEM.htm" file as you require, using text or HTML editor.

On the next occasion when a VIC downloads the ActiveX component, the contents of your new "OEM.htm" file will be displayed.

Troubleshooting:

- The custom file is not being displayed
  Ensure that the name is "OEM.htm" and that the file is in the VTScada install-ation folder of the computer hosting the VTScada Internet Server.

- Message is not being displayed as desired / Error messages.

---

[1]Hyper Text Markup Language. Used for web pages.

Refer to an HTML guide.

Sites & Maps

The Sites page (or Site List page), found in the menu of every new applic-
ation, will display a list of Site Tags beside an optional Site Map. Site tags
include the Polling Driver, DataFlow RTU, MultiSmart, MPE Duplexer, MPE
SC, and ScadaAce, as well as any context tags or user-defined types of
your creation that contain site parameters.

As of version 11.1.15, Analog Statistics and Digital Statistics tags can
also be included as folders in the sites list.

Items in the list may will include the tag description and basic com-
munication details, but may be displayed using a compact view when
there are otherwise too many to fit within one screen. In the compact
view, only the Site Draw icon and the name are shown. Use the compact
view button to switch modes. The list can be sorted by site name or
description. In the case of Polling driver sites, you can also sort the list
by data age or site number.

What will happen when an operator clicks on any site included in the list
will depend on how you have configured the site, and on whether the site
contains other tags that can be considered site tags.

> Note: There is no Update Site Location button on this page. To locate or
> move a site pin, open the Site Map page or Site Details page for one
> specific site.

List: Site Entries versus Folder Entries

You have control over whether a particular site tag will be included in the
list. Excepting Station Tag sites, you also have control over whether an
operator's click on the site-list entry will open the Site Details page, or
drill down into the site to display its child I/O and sites in the list.

All site tags will have an "Exclude from the site list" option in their con-
figuration folder on the Display tab.

Polling Drivers and Dataflow RTU tags will have the following con-figuration options in their Display tab. Context tags are similar but have an additional option, "Automatic".

- Display as Site: A click will open the Site Details page as a pop-up.
- Display as Folder: A click will leave focus on the Sites page, but the list will now show the child tags of this site.
- Exclude: This tag should not be shown in the Sites page list.
- Automatic: (Context tags only) If this tag contains child sites, folder display will be used. Otherwise, site display will be used.

As an example, in the following image of a Site page, Bedford Water 1 and Bedford Water 2 are context tags, identical in all respects except for their name, description and the fact that Bedford Water 1 was configured to be displayed as a site while Bedford Water 2 was configured to be dis-played as a folder. Both contain I/O tags and a child context, Tank 1.

Site: A click on Bedford Water 1 in this list will open its matching Site Details page:
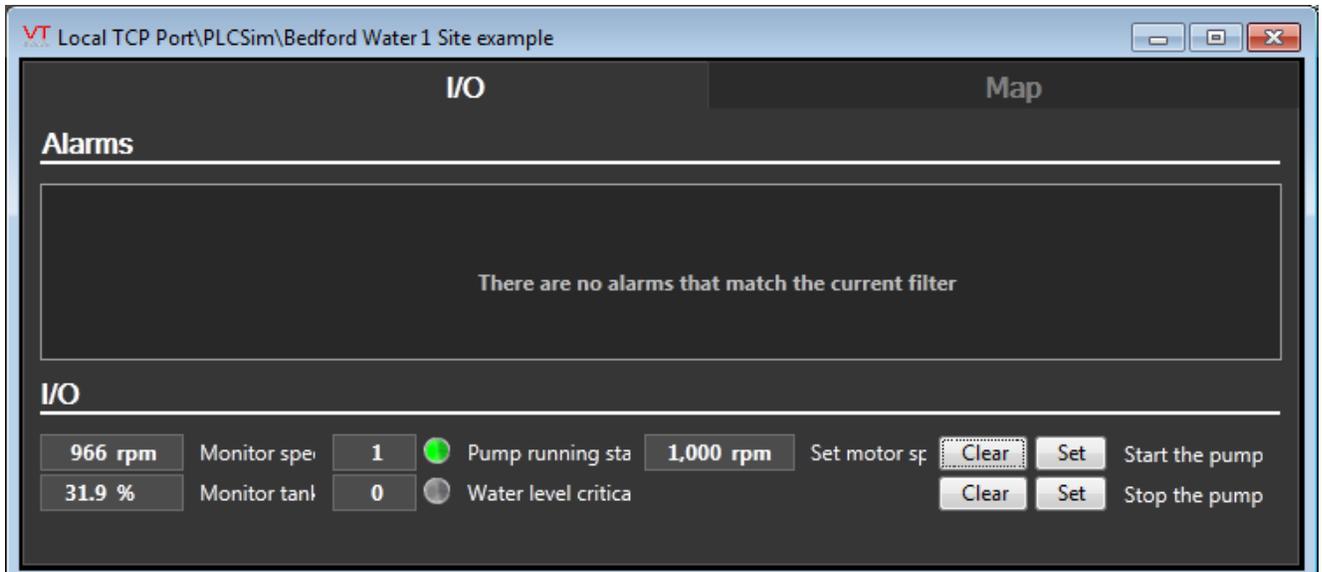
Folder: A click on Bedford Water 2 in the list will drill down into the site:
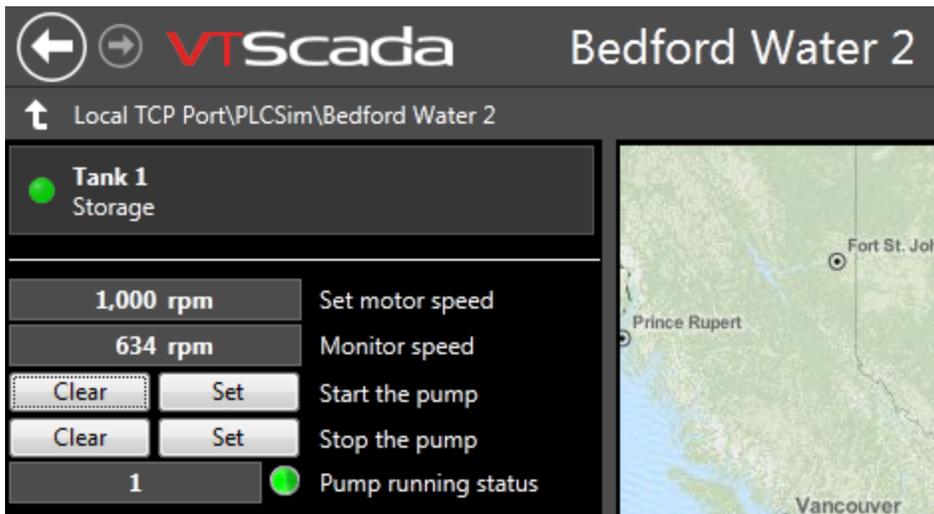
Note: If a tag is configured to be displayed as a folder, but has no child tags that could be displayed in the list, then it will be excluded from the list. The list cannot include an empty folder.

Note: To obtain this option in a Context tag, developers will typically click the Add Site Parameters button. This will add all available site parameters to the tag, some of which may not be needed. You can remove any that are not required for your purposes.

Sites Display Controls

The Sites page can be shown with or without the map, by using the map button ⊕. If the map is displayed, Site Draw icons will provide color-coded information about communications with, and alarms at each site. Hovering the pointer over any of the sites in the list will cause the corresponding icon to be indicated by an animated dot.

To locate a site icon for any site, click that site's entry in the list to open a Site-Details page. There, you will be able to click the Update Site Location button to select a location on the map for the site.

If your site contains sub-sites (for example, a county site that contains several station sites) then it is possible to navigate from parent site to child using the list. Parent sites will have an arrow that can be clicked to open the list of child sites. To return to the higher level, use the up arrow above the list.

Filter Child Types in the Sites Page

As an example, your custom type might be a city. Within that city (and as child tags of this type) there might be stations and other sites. When this tag is viewed in the sites page, you will be able to expand the list to view all such parent tags. But, you might want to restrict that list to only the stations or only the child sites.

A parent context, as shown in the sites list.

You can filter the list of child sites by adding the parameter CustomSiteListFilterType, and setting its value to the type name (or name of a group of types, such as "Station"), that should be shown in the list. The filter affects only the immediate children of this tag instance.

Related Information:

...Site Map

...Site Details Page

...Context Tags – Create a generic and customizable template that you can use to define any natural grouping in your application

...Polling Driver Tags

...Data Flow RTU Driver Tags

...MultiSmart Station Tags

...MPE Duplexer Station Tags

...MPE SC Series Station Tags

...ScadaAce Site Tags

Site Map

A Site Map is a dynamically-loaded map, primarily used to show the location of any of the site-related tags: Polling drivers, DataFlow RTU

drivers, MultiSmart™ stations, MPE™ Duplexer stations, or MPE™ SC stations. You can also link a Context tag or a user-created type derived from a Context tag to a map by adding the properties "Latitude" and "Longitude" to the parameter list. A Site Map is a standard part of the menu, and is built into every automatically-generated lift station details page and into the Site Details page . While you can add extra Site Maps to your application using the Site Tools palette or the Menu, it is usually not necessary or beneficial to do so.

Site:

Sites are displayed using markers, either the default pin or a shape of your own creation. Open the Site Map page for a specific tag in order to access the Update Location button to place or move the site.

Any of the following tags can be a site when configured to have Latitude and Longitude values, and to be shown on a map:

- Context tags and user-defined types derived from Context tags, if they have site properties.
- Polling drivers.
- DataFlow RTU drivers.
- MultiSmart™ stations.
- MPE™ Duplexer stations.
- MPE™ SC stations.

Connector:

Connectors are displayed as a line or pipe between two Sites. The color and style of the connector is configurable, but not the width. Connectors are displayed only when two sites with latitude and longitude values have been selected, and when the connector is configured to be shown. Only Context tags or user defined types derived from a Context tag can be connectors. A connector will remain visible if you zoom in to a map tile between two sites.

Since a connector is based on a Context tag, it can have user-defined properties and child I/O tags.

Slippy Map

The map is sometimes referred to as a "slippy map" because of the way it can be moved within its frame. Operators can pan and zoom to change the area displayed.

The map display is built of "tiles" – static images. At each higher magnification level, four times the number of tiles are used for a given area. As the operator moves the display or changes magnification level, new tiles must be downloaded. Once downloaded, they will be cached locally on your computer. These tiles are not distributed with VTScada, therefore your server must have an active Internet connection in order to be able to download tiles that have not been cached.

It is possible to switch from map view to satellite view, or to a different map provider. See the task list at the end of this topic.

Any instance of the tag types listed above can be located on a Site Map as a pin. Operators can click on the pin to open the associated Site Details Page. Instructions for create your own pin shapes are provided elsewhere. Refer to the list of tasks at the end of this topic.

The primary site associated with a map will be marked by an animated bubble. The head of the standard marker pin uses the same color-indication system as the Site Draw Drawing Method, and described by the Site Legend widget.

(c) OpenStreetMap Contributors http://www.open-streetmap.org/copyright
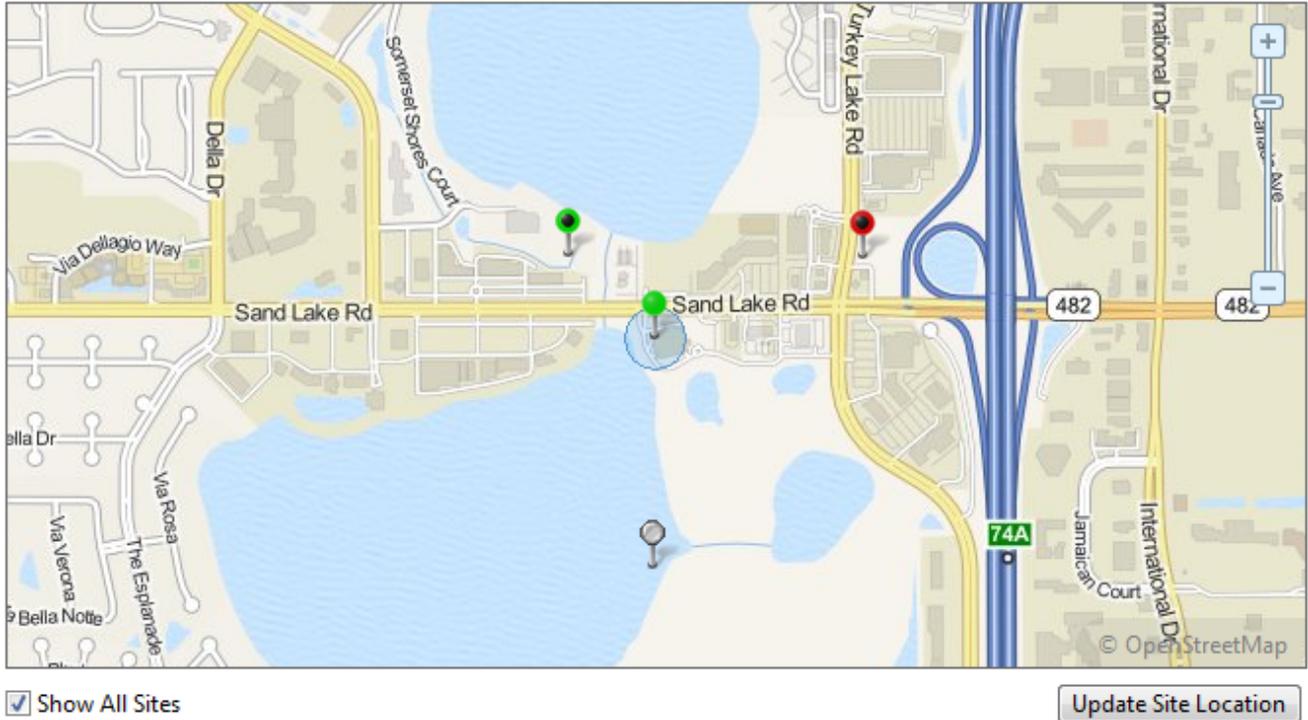
Map tiles will be downloaded only once. If you believe that newer images are available, delete the files from sub-folders of C:\VTScada\Data\SlippyMapTiles. New tiles will be downloaded the next time you view the map.

Available controls for the map include: (*)

Control	Action
Show All Sites	A check box to enable or disable pin display for sites other than the root.
Update Site Location	After clicking this button, the next click within the map area will re-locate the site.
Magnification Control	A slider between (+) and (–) buttons. Provides a way to increase or decrease the magnification (zoom level) one step at a time.
Click on any pin	Opens the details page for that tag, as appropriate for the type.
Click and drag	Relocate the focus of the map.

Mouse-wheel	Zoom in or out.
Double-click	Zoom in, one step, at the location clicked.

(*) Some controls will not be available, depending on how the map is viewed.

> Note: Rather than use the default cache, you may set an independent source for the map tiles with MobileSlippyMapTilesSource1 in your SETUP.INI file.

Related Information:

...Context Tags – Create a generic and customizable template that you can use to define any natural grouping in your application

> ...Add Site Properties. – Add and configure site properties

> ...Add Connector Properties – Add and configure connector properties

...MobileSlippyMapTilesSource1 – VTScada Admin Guide – Control the location where map tiles are stores.

...Site Details Page

...Sites & Maps

Related Tasks:

...Add a Site Map to a Page

...Add a Site Map Page to the Menu

...Change the Map Source

...Use Maps Without an Internet Connection

...Create Custom Map Icons

Add a Site Map to a Page

VTScada has built-in links to site maps, both within the menu, and within Sites Pages and Site Details pages. These are usually sufficient for most applications, but if you wish, you may add a Site Map to any page.

To add a site map to a page:

1. Open the Idea Studio and the page to which you intend to add a map.
2. Open the Site Tools palette (found within the Widgets palette section).
3. Locate the Site Map widget and drag it onto the page.
4. Provide parameters as appropriate. Parameters are described later in this topic.

Note that the map is drawn within its own window, which will have a higher z-order than the page it is being placed on. The result can be loss of part of the VTScada header or footer if the map is larger than the page.

The properties dialog options for a Site Map drawn this way, are as follows:

Parent

The station tag instance (and any child tags that also have valid latitude and longitude values) that is to be the primary target of this site map. All sites shown will be marked by a pin, but the root site will be further indicated by an animated dot.

Show All Sites

Choose whether this map instance should show only the root site, or if all sites visible in the map area should be shown. In either case, the pin marking the root site for the map is the only one highlighted by an animated bubble.

Use Theme

Choose whether the borders of the map should follow the VTScada color theme

Text Color

Specify the color to be used for the labels below the map.

Widget Folders:
Basic Components\Specialty\Mapping\
Tools\Site Tool Library\

Add a Site Map Page to the Menu

A Site Map page is included in the menu of every new application. You will need the information in this topic if you are upgrading an application that was created before version 10.1, or if you have removed the Site Map page from the menu and need to add it back. When doing so, it is important to configure the parameters correctly. To have the page open with no parameters, and without prompting for parameters, you must ensure that the three choices are cleared.
The default parameters are as shown:
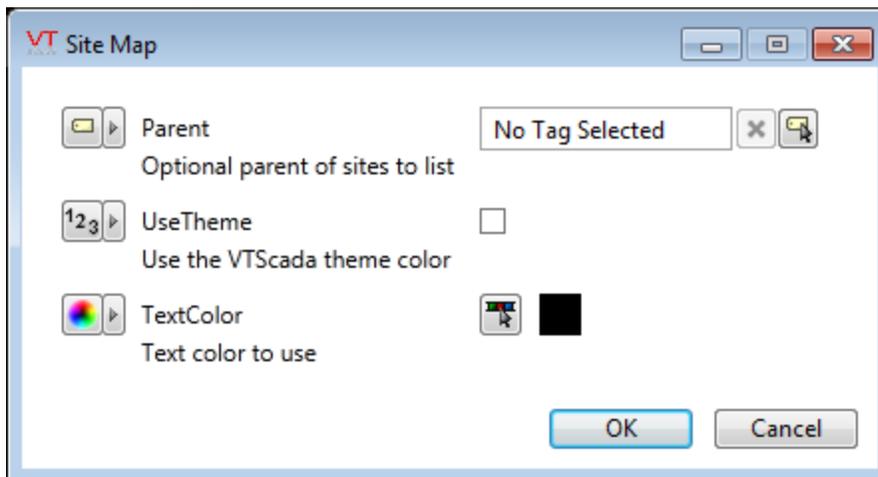
The default menu link does not specify any particular site. You may choose a specific site, or the parent of a set of sites.
The Use Theme option controls whether or not the application's color theme will apply to the overall look of the Site Map page. This is recommended in most cases.
The Text Color defaults to white (15 as defined in the VTScada [color] palette). You may choose any other number, or specify an RGB value.

It is expected that there will seldom be a reason to add additional Site Map pages, beyond the one already in the menu of every new application. The Site Map page is a Site Map that fills an application page. The scale of the map will adjust automatically each time that it opens in order to display all sites within the application. When the site map is opened as a page, you cannot specify a root tag.

Change the Map Source

The default configuration of the Site Map page is a basic map, showing roads and towns, and using tiles from Open Street Maps™. This is stored in the Setup.INI property, SlippyMapRemoteTileSource1.
You can change to another source, including:

- Any other third party provider of map tile images.
- Satellite image tiles rather than map tiles.
- Another workstation on your network, having a cache of tiles. This latter option avoids the need for each workstation to download its cache of tiles from an internet source.

In all cases, the images that make up the map (or satellite view) are square tiles, downloaded to the folder C:\VTScada\Data\SlippyMapTiles. A separate folder is used for each zoom level, with the numeric folder name corresponding to the zoom level. VTScada will expect the map tiles to use the .PNG file format.
A copyright attribution should be appended to the tile source.

Procedure:

1. Stop VTS
2. Using a text editor, open the file, Setup.INI
3. Change the property SlippyMapRemoteTileSource1 as follows:
   SlippyMapRemoteTileSource1 = http://Tile_Source /|/ copyright attribution
   Examples for Tile_Source follow.
4. Save the file Setup.INI
5. Restart VTScada and run your application.

Example values for the tile source:

Default value for SlippyMapRemoteTileSource1:

SlippyMapRemoteTileSource1 = http://c.tile.open-streetmap.org/ /|/ © [OpenStreetMap Contributors] (www.open-streetmap.org/copyright).

Value for another workstation:

SlippyMapRemoteTileSource1 = http://some_workstation_name/MapTiles/1/ /|/ © [OpenStreetMap Contributors] (www.openstreetmap.org/copyright).

Related Information:

...MobileSlippyMapTilesSource1 – VTScada Admin Guide – Control the location where map tiles are stores.

Troubleshooting:

- No map or satellite image is shown.
  Check for typos in the setting of SlippyMapRemoteTileSource1.
  Ensure that the computer can connect to the Internet.

Use Maps Without an Internet Connection

In many instances, VTScada will be running on a computer were internet access is forbidden or severely restricted. You can still use slippy maps by copying the map tiles that you need from a computer that has internet access to the computer that does not. The following method works because each time you view the same map again, VTScada will use the tiles that are in the cache, rather than pulling new copies from the Internet.

Note: Due to licensing restrictions, Trihedral cannot supply map tiles. The files must be downloaded from Open Street Maps or other source.

1. On a computer that has Internet access and a copy of VTScada (possibly a trial version), view all map areas that will be required for your application. Ensure that you view all areas at all zoom factors.

2. Copy the folder and sub-folders C:\VTScada\Data\SlippyMapTiles to the station that does not have Internet access.

The names of the folders and the map tiles match the area and zoom factor viewed, and are not unique to your computer or VTScada in any way.

VTScada will expect the map tiles to use the .PNG file format.

Troubleshooting:

- The steps were followed, but the map is not displayed.
  Ensure that the tile images are in the folder "Data\SlippyMapTiles" under your VTScada installation.
  Ensure that you are attempting to view an area that matches what was saved in the map tiles.

Create Custom Map Icons

You can create your own custom map icons (pins). For example, you may wish to provide visual clues to help operators identify sites when more than one is displayed on a map.
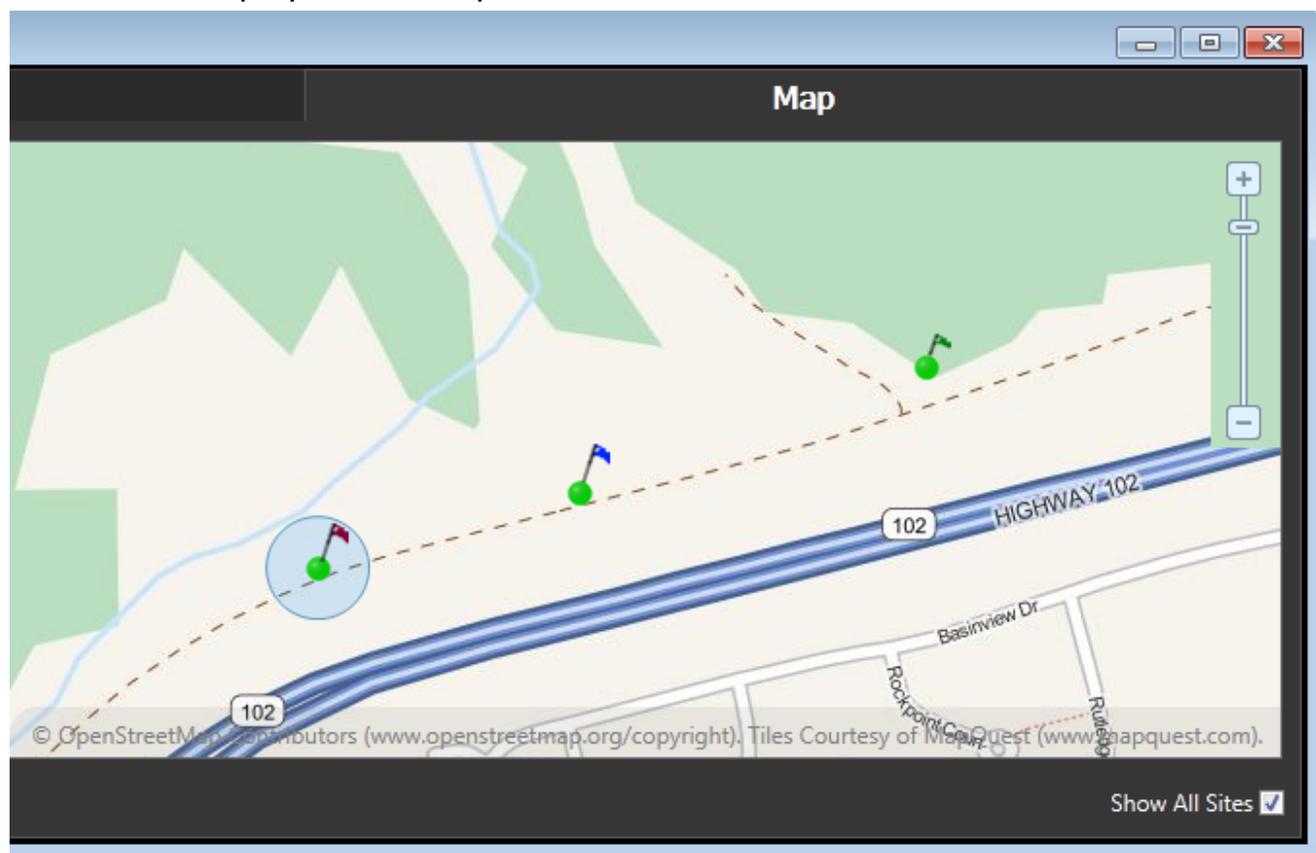
Each icon is a Tag Widget that you create. To function as a map icon, various parameters will need to be set and mapping-related VTScada widgets will need to be included. A pulse beacon will be added automatically for you. When an operator clicks your icon, it will open a site page (or your custom page). You do not need to add a hotbox or site icon.

General Steps to Create a Custom Map Icon:

1. Create a new Tag Widget.

2. When prompted for a name, choose one that will help you remember what the widget is for.

3. When prompted to select tag types, choose only the types that this icon will be used with.
   For example, Context, Polling Driver or your user-defined types.

4. Add the graphics that you want for your custom icon.
   See the following list of key details.

5. Instruct your site tag to use that widget instead of the built-in pin using one of the following methods.

Use Your Custom Map Icon

There are four ways to select the custom map icon:

Note: After changing the selected icon, you may need to refresh the map or update a site's location before the new icon will load in place of the old.

- To use the icon for a specific site:

   a. Open the tag's configuration folder.

   b. Within the Display tab, choose the custom map icon widget.

   Note: this assumes that, when creating the widget, you configured it to be a tag-widget for the type of site you are now trying to configure.

   If the site is a custom type, ensure that the type definition includes site parameters, including Custom Map Icon.
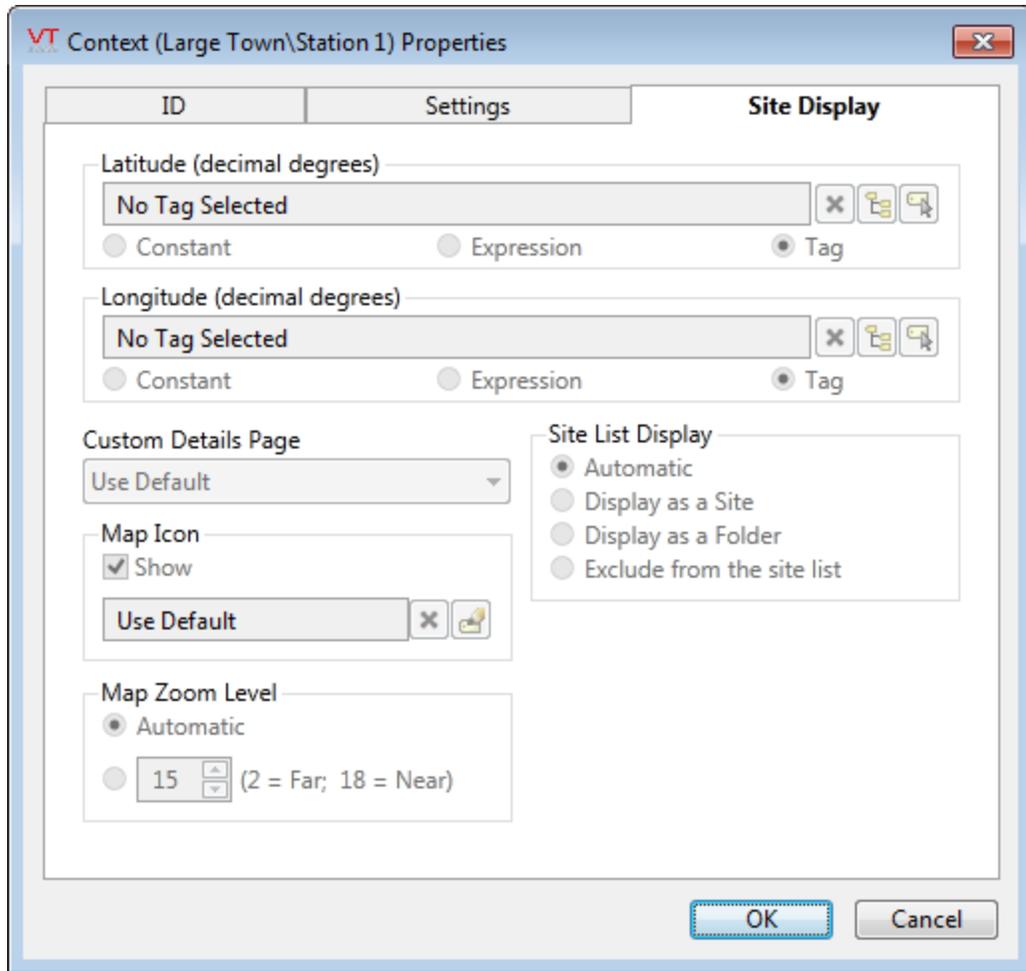
- To use the icon for a group of sites that are the children or grandchildren of a specific Context tag:
    a. Ensure that the parent tag has been assigned the site display parameters.
    b. Ensure that the icon widget can be linked to all the child tag types.
    c. Set the Custom Map icon property to the name of your widget.
- To use the icon for all instances of a given type of site:
    a. Add a property to the [System] section of your Settings.Dynamic file.
    b. Name the property after the *type* of tag you are configuring by putting the type name in front of the keyword, "MapIconName".
        For example, "PollingMapIconName = MyMapIconsModuleName".
- To use the icon for all sites of all types:
    a. The widget must have been named and therefore stored in a file name, "CustomMapIcon".

Note: this is the *name* of the widget, matching the file it is stored in. This is not its *title*.

Key Details:

- **Image or Shape:** You may wish to keep the following details of the default pin shape in mind while creating your new icon:
  The default pin is approximately 30x30 pixels in size.
  It uses a flag image that has a transparent background.

- **File Name:** The custom map icon is identified by the name of the file it is stored within, not the label displayed in the file menu.

  - To discover a widget's file name:
    1) Open the widget in the Idea Studio.
    2) Hover over the name tab.
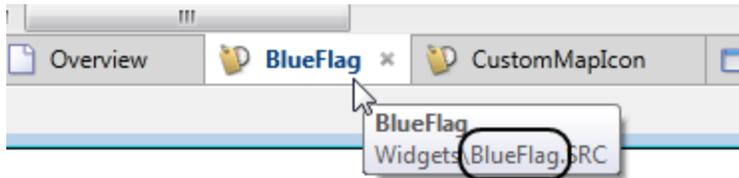
- **Pin Center:** The center of the new site pin's bounding box will be used as the pin location when shown on a map. The center's location is calculated using all graphics that make up the whole.

- **Parameters:** The tag widget should have the following parameters. Three of these are created automatically. ZoomLevel is optional (see next point).
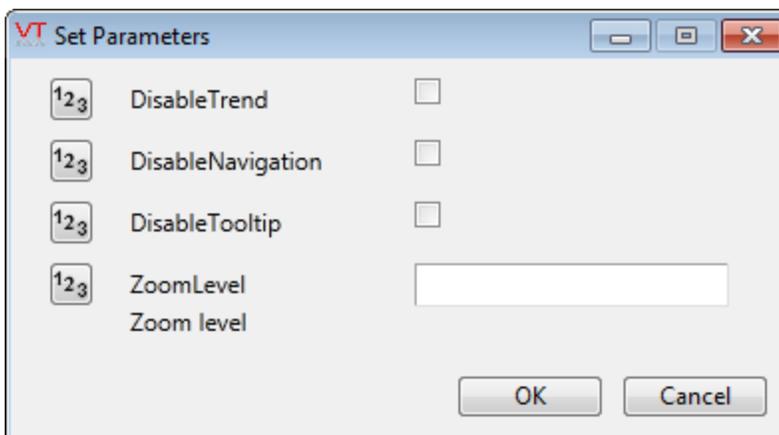
- **ZoomLevel:** A numeric value, ranging from two to eighteen. If you add this numeric parameter, your widget can know what the current zoom level of the map is, because VTScada will automatically set this parameter's value.

The value, "2", corresponds to the furthest out that an operator can zoom on any map and "18" is the closest in.

For example, you might set the opacity property of images within the widget to use this in an expression that will be become zero, hiding the icon, or portions of it, when the operator has zoomed in or out further than a given value.

Related Information:

...Site Details Page

...Create a Custom Site Details Page

...SiteDetailsWindowed in the VTScada Admin Guide

Site Details Page

The Site Details page shows communication, alarm and I/O information for site and station tags. These include the Polling Driver, DataFlow RTU, MultiSmart, MPE Duplexer, MPE SC, and ScadaAce tag types, as well as all custom types that you create, based on Context tags.

The Site Details page is automatically generated for you by VTScada. You may choose to use pages of your own creation instead, on a site-by-site basis.
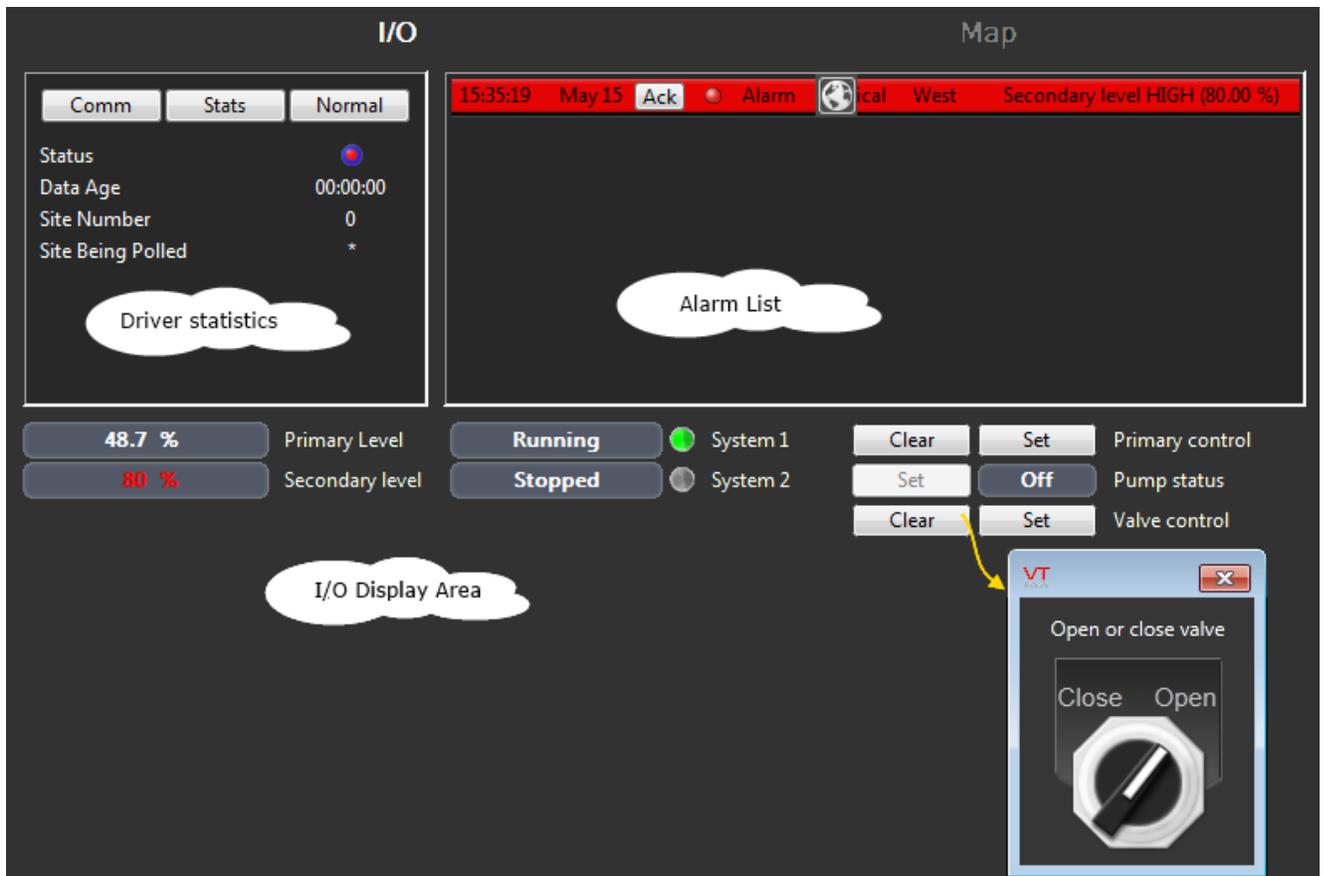
There are four methods that operators can use to open this page:

- Click on a site that is included in the list of the Sites page.
- If you use the Site Draw widget to draw a Polling tag or DataFlow RTU, then operators can click on the Site Draw to open this page.
- If you have drawn any of the available site types as a Site Icon or Site Summary, (this includes a pin, drawn on a Site Map), then operators can open the Site Details page by clicking on the icon or pin.
- You can draw the page within any other application page. In general, this method is discouraged in favor of the previous three. You are likely to encounter scaling problems when the Site Details is drawn inside a standard VTScada page.

Components of the Site Details page:

Tabs:

Every Site Details page has two tabs: I/O and Map. The map is simply a Site Map, showing the location of the station.

Driver Statistics Section:

Visible when the I/O tab is selected. When associated with a Polling driver or a DataFlow driver, a window on the left shows statistics associated with the driver. Also included are buttons that open the Comm Messages dialog or the Comm Statistics dialog for the driver.

Alarms Section:

The Alarms list will be populated with all current alarms associated with the driver. Operators may use the Ack button here to acknowledge any given alarm.

Note: Only alarms whose area matches the area of the site tag will be displayed. If two site tags share the same area, it is possible to see alarms from both. In general, each site tag should have its own area.

I/O Section:

All of the I/O tags associated with the site are shown at the bottom of the page. These are grouped into separate columns according to the type of I/O; analog or digital, input or output. Input tags that allow writing are still considered input tags. Selector switch tags are included with the digital outputs. Any unacknowledged alarm on an I/O tag will be indicated by the color red. Red text for analog tags and a red dot for digitals. Controls are provided for all output tags associated with the station.

> Note: The application property, SiteToolsConfirmOutput, controls whether a confirmation prompt will be shown before new output values are sent.

I/O Section Filtering:

If the site tag has a parameter named CustomSiteListFilterType, then the I/O list will be filtered to show only the type, or types in the group, specified for that parameter.

Related Information:

...Sites & Maps

...Site Map

...Site Details Configuration

...Create a Custom Site Details Page

...SiteToolsConfirmOutput in the VTScada Admin Guide

Site Details Configuration

The Site Details page normally opens in its own window. You can configure it to open in the main display window by setting the application property SiteDetailsWindowed to false (0).
You can control the colors used for digital values (Digital Input, Digital Status and Pump Status). The application properties are included in the following table, which also shows the default color for each state.

State	Color	Property
Invalid	black	DigitalIndicatorInvalidColor
0	light gray	DigitalIndicator0Color
1	green	DigitalIndicator1Color
2	blue	DigitalIndicator2Color
3	purple	DigitalIndicator3Color
alarm active	red (blinking if unac-	DigitalIndicatorAlarmColor

	knowledged)	
bad quality	orange	BadQualityColor

Related Information:

Refer to the VTScada Admin Guide for:

...BadQualityColor

...DigitalIndicatorInvalidColor

...DigitalIndicator0Color

...DigitalIndicator1Color

...DigitalIndicator2Color

...DigitalIndicator3Color

...DigitalIndicatorAlarmColor

...SiteDetailsWindowed

Create a Custom Site Details Page

You may prefer to create your own site details page in place of the built-in version. Any page can be used as a Site Details page, but in general these tend to be parameterized pages designed to show the I/O values, controls, and other information associated with a site.

Custom site pages should be provided with the following parameters *in order*. The names are suggestions; the order counts.
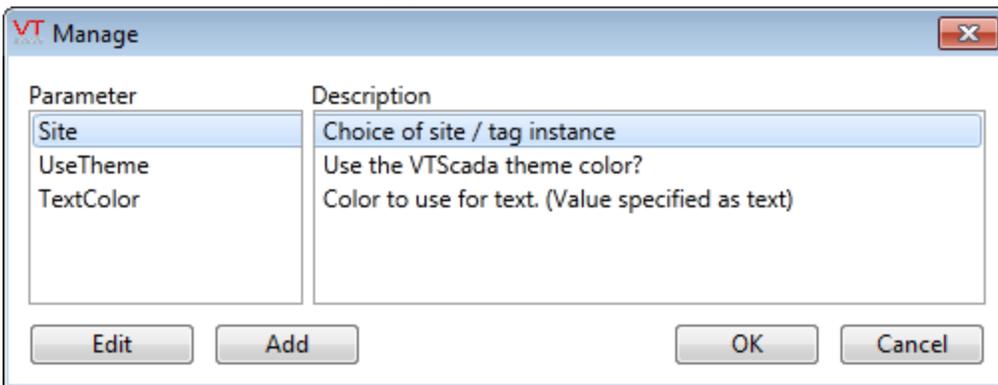
- Site: A tag parameter. Widgets in the page should be linked to the children of this parameter.

- UseTheme: A status (Boolean) parameter. Controls whether or not the VTScada color theme is applied to the page.
- TextColor: Must be specified as a text parameter. You may set values using numbers from the VTScada palette or RRGGBB strings. Controls the color used for labels within the site page.

Standard Site Details Page

A Custom Page to Use Instead
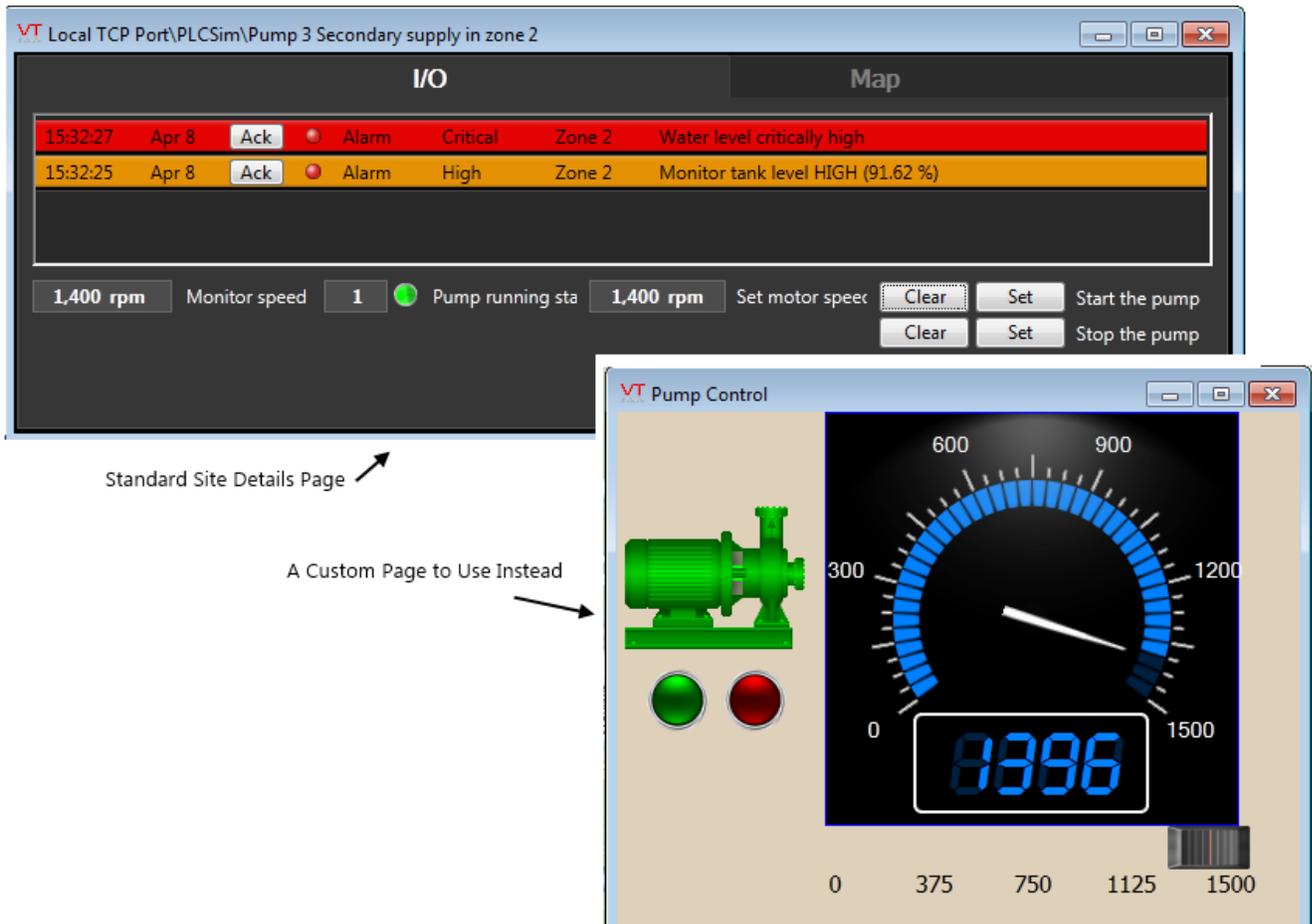
There are two methods to specify a custom site details page:

Method 1: Configure your site tag:

1. Configure your site tag so that it has the parameter, CustomDetailsPage. This parameter is part of every station tag, and can be added to your Context tags with the button Add Site Properties in the Display tab. If you have already turned your context tag into a custom type, use the same button in the Manage Types page of the Application Configuration dialog.

New Context Properties

	ID		Settings		Site Display

Property Name	Value	Comment
Latitude		Latitude (decimal degrees)
Longitude		Longitude (decimal degrees)
InitZoom		Map Zoom Level
CustomDetailsPage		Custom Details Page
CustomMapIconParm		Custom Map Icon
SiteListDisplay	0	Site List Display

Add	Copy	Delete

Add Site Properties	Add Connector Properties

OK   Cancel

2. For each instance of your site tag, select the page to use and configure the parameter values, if any.

Method 2: Configure the Site Summary or Site Icon widget that displays your site tag.

1. If your site is linked to a Site Summary widget or Site Icon widget, then configure that widget to use the custom site details page.
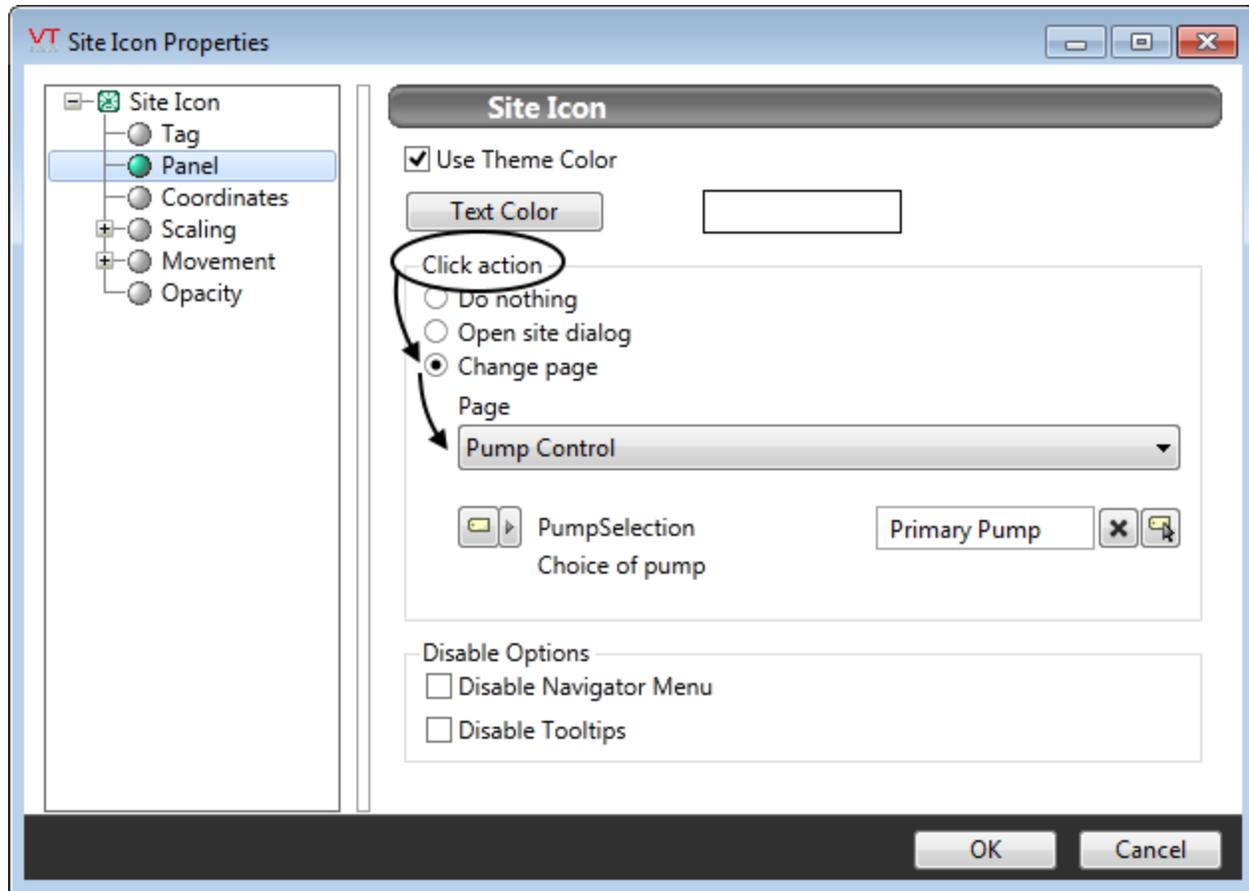
Related Information:

...Page Parameters & Widget Parameters

Version Control

Every time that you make a set of changes to your application, VTScada records the new configuration in a repository. You can return to an earlier stage of development at any time by using the commands in the Version Log.

> Note: Full access to the version control system is an optional feature that your company may or may not have purchased with your VTScada license. Check the list of features included with your VTScada license in the About VTScada dialog.

The version control system offers far greater flexibility than a simple Undo-Redo. Features include:

- The ability to return directly to any earlier stage of development.
- Since the act of switching to an earlier version is itself logged as a stage in the version control system, you have not "undone" and lost changes to intermediate stages of development, but have added one more change to the log.
- You can merge (or then reverse) specific changes from a selected version. For example, given an application with 20 versions recorded in the version control system, you decide to switch to version 10. You can then choose to merge just the changes made in versions 12, 15 and 18 into version 10 and carry forward from there. (Note: later versions often depend on work done in earlier versions. If you merge the work from a given version, be sure to include all related changes.)
- You can reverse or merge specific changes from a range of versions. You can select a range of versions, and then select specific changes done across those versions to be reversed, or merged.
- VTScada uses a distributed version control system. Version changes can be made by authorized users on any workstation, regardless of whether the primary configuration server is available.

In the context of the VTScada version control system, "reverse" means to undo a change. "Merge" means to blend a change into the current revision. That change may be taken from another workstation that has not

been configured to deploy its changes automatically, or it may a change that was reversed.

Related Information:

...Version Changes and Security – How version changes may affect account information.

...Commenting Changes – Maintain a record of why changes were made in each revision.

...Local Changes versus Deployed Changes – Choose when to share updates with all machines the application is running on.

...The Version Log – View and work with revisions.

Related Tasks:

...Show Version Details – View line-by-line what changed in what file.

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Version Changes and Security

When you add, modify or delete user accounts, these actions count as changes to your application and therefore become part of an application's version history.

If a switch to a different version of the application will affect user accounts, you will see a warning, asking if account information should be changed. You can say "Yes" in response to this dialog to apply all changes including those that will affect account information.
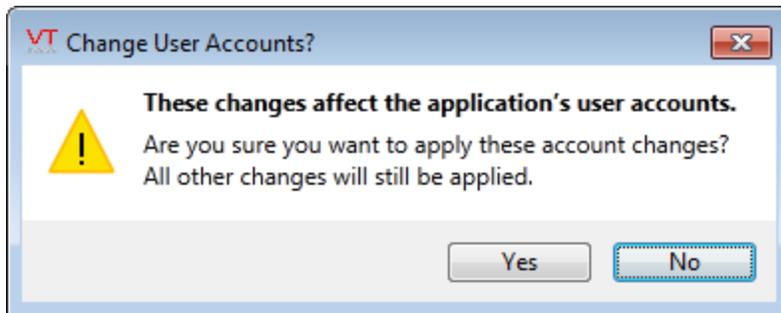
If you select "No" only the changes that do not affect user accounts will be applied.

Having selected "Yes," you can still recover those account changes. Do so by merging back in, those revisions in which the account changes were made.

When reversing or merging changes, you can choose to exclude the Accounts.Dynamic from the list of files included in the action.

Note: Security changes are always deployed immediately, regardless of the current setting of Auto-Deploy.

Related Information:

...The Version Log – View and work with revisions.

...Commenting Changes – Maintain a record of why changes were made in each revision.

...Local Changes versus Deployed Changes – Choose when to share updates with all machines the application is running on.

Related Tasks:

...Show Version Details – View line-by-line what changed in what file.

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Commenting Changes

When you apply or deploy a change, VTScada will prompt you for a comment. The title of the dialog will vary according to the type of change being saved.
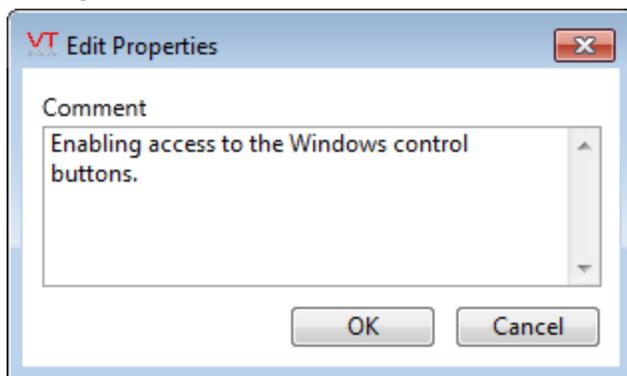
Comments are extremely helpful. The version history for any application is likely to become large over time. By making a note of why each change was made, you can create a clear history documenting your application's development.

Comments are not mandatory in the default configuration of VTScada. If you prefer to make comments mandatory, you can change the minimum length required. To do so, add the application property RepositoryCommentMinLen to the Layer section of your Settings.Dynamic file. After setting a value for the property RepositoryCommentMinLen, users will see a dialog similar to the following if they do not provide the required number of characters.
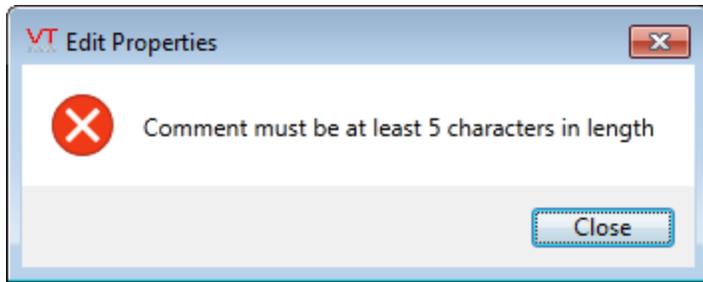
Related Information:

...The Version Log – View and work with revisions.
Refer to the VTScada Admin Guide for:

...RepositoryCommentMinLen – Enforce comments.

...RepositoryCommentDisable – Comments not requested.

Related Tasks:

...Add a Property – VTScada Admin Guide – Instructions for adding a property to your application.

Note: Version Control is an optional license feature and may not be available at your site.

Local Changes versus Deployed Changes

The most recent deployed change becomes the official version of the application. In a remote or networked application, deployed changes are distributed by VTScada to all the workstations running that application. Until a change is deployed, it is said to be "local". Local changes are stored in a separate branch of the application's version history. They do not become active on other workstations (although they can be viewed there), and they can be erased using the Revert Changes page. Local changes also have an effect on applications that run on a single computer: A ChangeSet is created from the most recent deployed version. Modifications that have not been deployed are not included.
While there are local changes on a workstation, the title bar will display a "Work-in-progress" image.

Application has changes that are not yet deployed
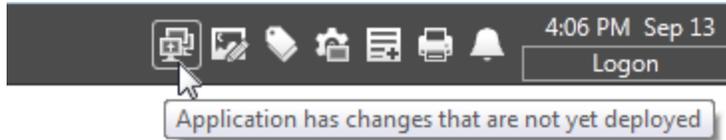
All changes to an application are automatically and immediately deployed unless you choose otherwise. In most cases, it is an advantage to have the Automatically Deploy Changes option in effect since it ensures that your edits are always saved as the official version of the application in the repository. A situation where you might not want this feature is if you are making changes to a networked application and do not want your edits to be installed on the other workstations until you have finished and tested your work.

When auto-deploy is off, you will need to take extra care to deploy changes that need to be in effect on all servers.

To toggle the Auto-Deploy Setting:

1. Open the Application Configuration dialog.
2. Click on "Information" from the menu.
3. Select or deselect the Automatically Deploy check box.
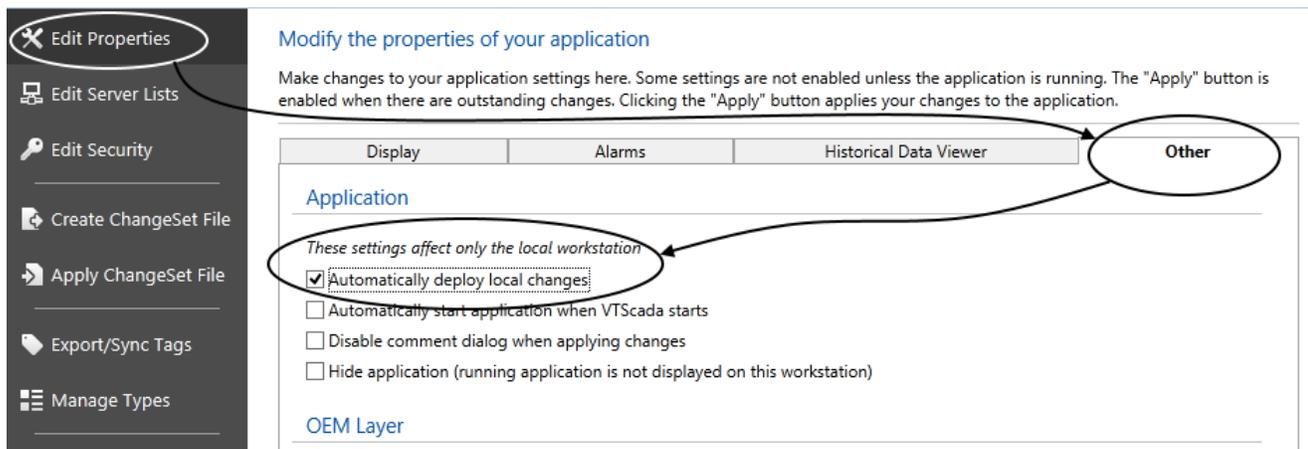
4. Click Apply.

When the option is deselected, all changes made to the application will be local to the current workstation until you either re-select this box or

use the Deploy Changes command of the Application Configuration dialog.

Note: Changes related to security accounts are always deployed automatically and immediately.
  * Auto-Deploy cannot be deactivated on a Run Time licensed workstation.
  * Changes to the Automatically Deploy option affect only the current workstation. You cannot remotely change this value on other workstations.
  * ChangeSet files are affected by Auto-Deploy. A ChangeSet file is an encapsulation of the version history in a file, therefore if you make a ChangeSet file while Auto-Deploy is switched off, your local changes to the application will also be marked local in the ChangeSet. If the ChangeSet is installed on another workstation, the running version will be the deployed version and will not include your local changes.

Troubleshooting:
- Critical updates are not being distributed to other servers.
  Use Deploy Changes, or turn Auto Deploy back on.

Related Information:

...Deploy Local Changes

...Reverting Local Changes

...The Version Log

Note: Version Control is an optional license feature and may not be available at your site.

Deploy Local Changes

(Except for the header text, the same screen is used when merging or reversing a range of changes.)

The Deploy Changes page is disabled while the Automatically Deploy Changes feature is in effect. The following information applies only while the Auto-Deploy feature is switched off.

Local changes can be deployed while you are working in the Idea Studio. Click the button provided in the title bar of the studio:

A more powerful tool for deploying changes can be found in the Application Configuration dialog:

You can open this page as follows:

1. Open the Application Configuration dialog.
2. Click on "Deploy Changes" from the menu.

Use the Deploy Changes screen to see exactly what was done to each source file and tag. For example, clicking on the Analog Control tag (shown in the previous image) will open a display showing what was done to configure this tag:

Changes made to source code such as a page, will show green lines for what is new, and yellow for the previous values. (Circled numbers link to descriptions in the following text.)

The following tools are available to help you work with this display:

(1) The number of changed files in each category is shown in parentheses.
(2) Before and after time stamp and file size.
(3) Columns can be sorted by type of change (addition + or removal –), line number, alphabetic.
(4) Move Next and Move Previous changes the focus box (5) from one change to the next.
(6) Changes throughout the file are shown in color bars on the side panel. The black rectangle shows how much of the file is visible and can be dragged to show the next change.

When you first open the Deploy Changes page, all changed items will be selected. By selecting individual items in the list and using the buttons at the bottom of the page, you can choose which changed items will be included when you click the Deploy Changes button.

Note: Warning: Development tasks often involve more than one file. Choosing to deploy only some of the code changes required by a VTScada object may cause serious errors in your application.

The Reload button will refresh the display. If you have made local changes while the Deploy Changes page was open, click this button to ensure that all changes are displayed before deploying.

Related Information:

...Local Changes versus Deployed Changes

...Deploy Local Changes

...The Version Log

Related Tasks:

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Reverting Local Changes

The Revert Changes page is disabled while the Automatically Deploy Changes feature is in effect. The following information applies only while the Auto-Deploy feature is switched off.
Local changes to an application (that is, those changes that have not yet been deployed) can be reverted or undone. If you need to undo local changes, use the Revert Changes page of the Application Configuration dialog.
You can open this page as follows:

1. Open the Application Configuration dialog.
2. Click on "Revert Changes" from the menu

You can revert all changes, returning to the last deployed stage of the application, or you can select particular changes to revert.

When you first open the Revert Changes page, all changed items will be selected. By selecting individual items in the list and using the buttons at the bottom of the page, you can choose which changed items will be included when you click the Revert button.

Note: Warning: Development tasks often involve more than one file. Choosing to revert only some of the code changes required by a VTScada object may cause serious errors in your application.

The Reload button will refresh the display. If you have made local changes while the Revert Changes page was open, you should click this button to ensure that all changes are displayed before reverting.

Deployed changes are managed (and can be undone) using the Version Log.

Related Information:

...Local Changes versus Deployed Changes –

...Deploy Local Changes

...The Version Log

Related Tasks:

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

The Version Log

You can review stages of development by opening the Version Log:

1. Open the Application Configuration dialog.
2. Click on "Show Version Log" in the menu.
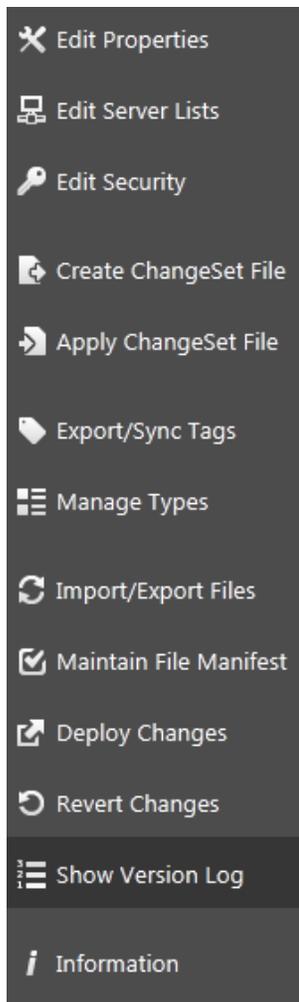
View the version status of workstations

The current application version on each workstation is shown below.

Local modifications are shown in blue. Deployed versions older than the latest are shown in red. The current deployed version is shown in black. Selecting a workstation from the Workstations list will display the Version Log for that workstation.

Right-clicking on a Version Log entry will allow you to display more detailed information about that version, switch to that version, revert to that version or adopt that version.

Workstations

Workstation	Last Update	Current Version
ANDREWH	Wed Aug 18, 2010, 15:19:22.283	ANDREWH-D20

Version Log for ANDREWH (Total of 21 records in log)

Version	Time Applied	User	Comment
ANDREWH-D20	Wed Aug 18, 2010 15:19:22.283	Manager1	Edit Properties: resizable screen
ANDREWH-D19	Wed Aug 18, 2010 14:49:28.597	Logged Off	Committing on Layer stop.
ANDREWH-D18	Wed Aug 18, 2010 14:26:27.255	Manager1	

Next 100   Show All   Reload

Version numbering uses the following system:

- First, the computer name where the change was made.
- A hyphen or dash.
- "D" indicates a deployed change.
- "L" indicates a change that was local-only at the time it was made, whether or not it was later deployed.
- The number following the "D" or "L" is a simple counter.

For example, version "ANDREWH-D20" is the 20th deployed change made on the workstation, ANDREWH.

To prevent the possibility of a faulty system clock creating out-of-sequence entries, the version counter takes priority over the "time applied" value.

The various features and tools in this page are shown in more detail in the following image. Numbers refer to notes in the following text.

1) List of workstations running the application

If this is a remote application, all the workstations that the application runs on will be listed in this table. Note that, if a machine is isolated from the network, its version list will almost certainly be out of date. Color-coding is used so that you can see at a glance whether each workstation is running the most recently deployed version of the application.

Black Text: The workstation is running the most recently deployed version

Blue Text: The workstation has local changes that have not yet been deployed.

Red Text: The workstation is running an out-of-date version. That is, the version on the workstation is older than the most recently deployed version.

2) List of version changes for the selected workstation

Displays the list of version changes for the computer selected in the Workstations table. Information displayed includes:

Version number – The version numbering system is explained earlier in this topic (example: ANDREWH-D20).

Time applied – All version changes are logged using UTC time, however the display will show local time at the current workstation.

User – The user logged in to the workstation when the change was made. If security has not yet been enabled, the change will be attributed to the Logged Off user account.

Comment – When manually deploying a change, you will be prompted for a comment for the version. Automatically deployed changes will often, but not always, be assigned a comment from a list of standard change descriptions.

3) Right-click on a version to open the options menu

By right-clicking on an entry in the version table, you can open a menu that provides several options for learning more about what went into a version, or to select changes to make current.

The buttons, Next 100 and Show All are used to step through the list of version changes, which may become lengthy over time. The Reload button will refresh the display to show any version changes that may have happened since the page was opened.

Related information that you may need:

...Show Version Details - View line-by-line what changed in what file.

...Commenting Changes – Record the reasons for the changes in each revision.

Related Tasks:

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Show Version Details

Detailed information about the changes that went into any version can be seen by right-clicking on an entry in the Version Log and selecting Show Changes.
This is an extremely powerful feature of VTScada that shows you exactly what changed in every altered file from one version to the next. Note that each folder type (left window) displays a count of the number of files of that type that were changed in this version (if any).
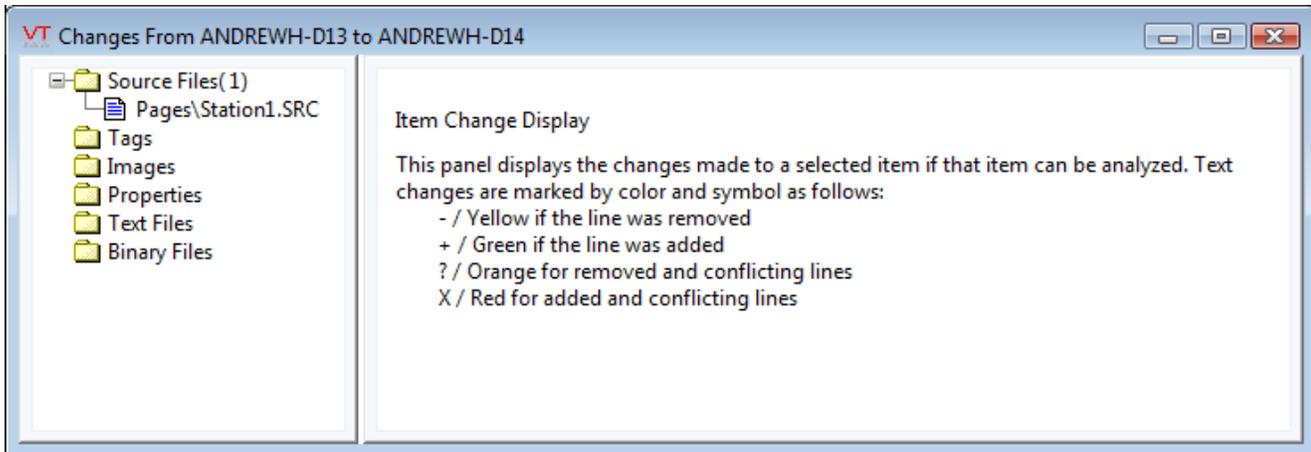
The details display is divided into two screens. On the left is a list of all files that were changed in this version. The right side screen initially displays a guide to interpreting version changes, until you select a file from the left screen, as in the following image. Each type of change is indicated by both a symbol and a color.
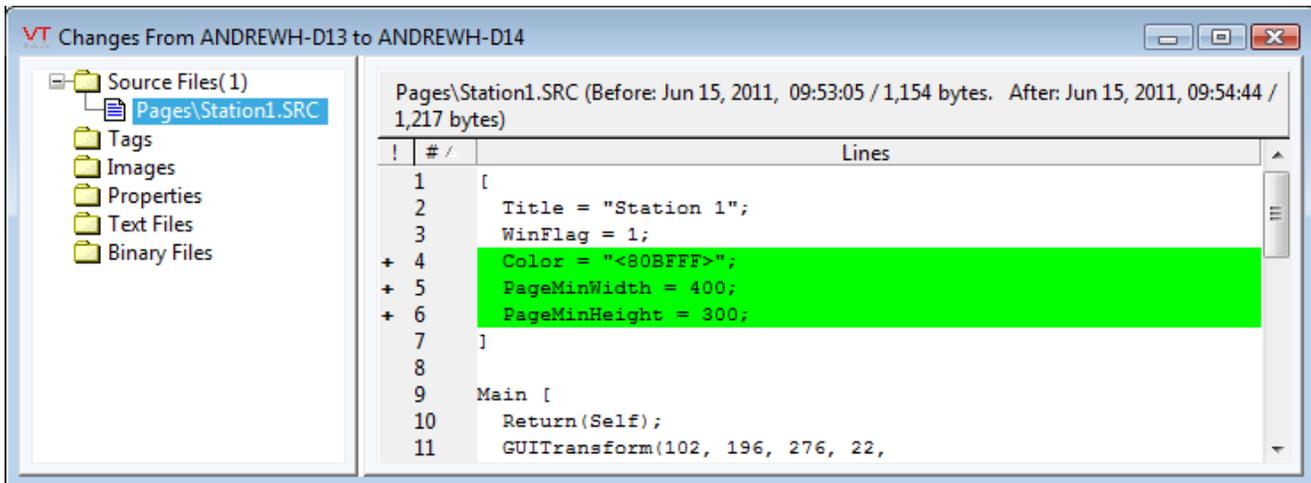
Using the color and symbol codes, as listed in the first image, one can see that in version D14 (note the dialog box title), the background color of the page Station1 was changed to color "<80BFFF>" (pale blue) and a minimum page height and width were set.

Related Information:

...Version Changes and Security – How revision changes may affect security configuration.

Next Steps:

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Switch to This Version

Use this option to make any earlier version current. Changes made in versions since the one you are switching to are not lost. The act of switching versions becomes a version change itself. For example, your current version number is D12 and you switch to version D8. What will happen is that a new version will be added – D13 – which will be an identical copy of D8. Changes made in versions D9 though D12 will no longer be in the current version, but if required, you could easily switch to version D12 (thereby creating version D14, which is an identical copy of D12).

Version Log for ANDREWH (Total of 13 records in log)

Version	Time Applied	User	Comment
ANDREWH-D12	Thu Aug 19, 2010 13:31:52.242	Logged Off	Automated commit
ANDREWH-D11	Thu Aug 19, 2010 13:31:36.791	Logged Off	
ANDREWH-D10	Thu Aug 19, 2010 13:31:19.899	Logged Off	
ANDREWH-D9	Thu Aug 19, 2010 13:30:32.407	Logged Off	Automated commit
ANDREWH-D8	Show Version Details 13:30:08.864	Logged Off	Automated commit
ANDREWH-D7	Switch to this Version 13:29:38.422	Logged Off	
ANDREWH-D6	Reverse Version Changes 13:28:34.262	Logged Off	Completing page addition.
	Merge Version Changes		

Switching from D12 to D8 creates D13 - a copy of D8

Version Log for ANDREWH (Total of 14 records in log)

Version	Time Applied	User	Comment
ANDREWH-D13	Thu Aug 19, 2010 13:38:36.759	Logged Off	Switch to this Version (ANDREWH-D8):
ANDREWH-D12	Thu Aug 19, 2010 13:31:52.242	Logged Off	Automated commit
ANDREWH-D11	Thu Aug 19, 2010 13:31:36.791	Logged Off	
ANDREWH-D10	Thu Aug 19, 2010 13:31:19.899	Logged Off	
ANDREWH-D9	Thu Aug 19, 2010 13:30:32.407	Logged Off	Automated commit
ANDREWH-D8	Thu Aug 19, 2010 13:30:08.864	Logged Off	Automated commit

Switching back to D12 creates D14. Changes are always added, never lost.

Version Log for ANDREWH (Total of 15 records in log)

Version	Time Applied	User	Comment
ANDREWH-D14	Thu Aug 19, 2010 13:40:28.423	Logged Off	Switch to this Version (ANDREWH-D12)
ANDREWH-D13	Thu Aug 19, 2010 13:38:36.759	Logged Off	Switch to this Version (ANDREWH-D8):
ANDREWH-D12	Thu Aug 19, 2010 13:31:52.242	Logged Off	Automated commit
ANDREWH-D11	Thu Aug 19, 2010 13:31:36.791	Logged Off	

Troubleshooting:

- An error message indicates a conflict.
  The change you are making affects an unsaved change, or possibly work being done by another developer. Examine the error message for details and resolve the conflict before re-attempting this action.
- After this change, part of the application unexpectedly stops working.
  The changes include a detail that may be critical to the operation, such as a changed I/O address. Restore the previous version, examine the version log in detail, then repeat your action taking care to avoid changes that will damage the application.

Related Information:

...Show Version Details – View line-by-line what changed in what file.

...Reverse Version Changes – Remove selected revisions from the running application.

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Reverse Version Range – Select a set of version changes to remove.

...Merge Version Range – Select a set of previously removed changes to restore.

...Version Changes and Security – How revision changes may affect security configuration.

Note: Version Control is an optional license feature and may not be available at your site.

Reverse Version Changes

By reversing the changes made within a selected version, you can remove just the edits made to the application in that version.

For example, you have made the following changes to your application:

    a. Version D10 Added a page

    b. Version D11 Deleted a tag

    c. Version D12 Created a new tag

You then decide that it was a mistake to have deleted the tag in version D11. You can reverse just that one change as follows:

    1. Select the row D11 in the Version Log.

    2. Right-click to open the context menu.

    3. Choose "Reverse Version Changes".

All of the edits that separate version D11 from D10 are removed, but the new tag created in version D12 remains.

> **Note:** Use extreme caution when reversing a version change. If later versions depend upon work that was done in the version that you are reversing, your application will be damaged.

The act of reversing a version change does not remove those changes from the history. Instead, it adds a new version in which the change is the undoing of whatever was done in the reversed version. You could later decide to re-incorporate the changes from the reversed version by using the Merge Version Changes command.

Troubleshooting:

- An error message indicates a conflict.
  The change you are making affects an unsaved change, or possibly work being done by another developer. Examine the error message for details and resolve the conflict before re-attempting this action.
- After this change, part of the application unexpectedly stops working.
  The changes include a detail that may be critical to the operation, such as a changed I/O address. Restore the previous version, examine the version log in detail, then repeat your action taking care to avoid changes that will damage the application.

Related Information:

...Show Version Details – View line-by-line what changed in what file.

...Switch to This Version – Set the running application to an earlier (or later) version.

...Reverse Version Range – Select a set of version changes to remove.

...Version Changes and Security – How revision changes may affect security configuration.

Next Steps:

...Merge Version Changes – Restore selected, previously removed revisions to the running application.

...Merge Version Range – Select a set of previously removed changes to restore.

Merge Version Changes

If you have used either the Switch Version or the Reverse Version Changes command, then you can use the Merge Version Changes command to re-introduce the changes from a specific version. You can also use this tool to merge other machine's local changes into a new deployed version.

For example, your version log includes the following changes:

a. Version 10 Added a page

b. Version 11 Added a tag

c. Version 12 Drew the tag on the page.

You decide that the creation of the page was a mistake, but that the creation of the tag was not. You can proceed as follows:

1. Select version 9 in the table.

2. Right-click to open the context menu.

3. Click Switch to This Version
    Changes made in versions 10, 11 and 12 vanish from the current version.

4. Select version 11 in the table and right-click to open the menu.

5. Click Merge Version Changes.
    The tag is again part of the current version. The page does not exist and the tag is not drawn on the non-existent page.

Note: Depending on the change you may need to re-start your application.

For this example, the same effect could have been achieved by reversing the changes in versions 10 and 12.

Troubleshooting:

- An error message indicates a conflict.

The change you are making affects an unsaved change, or possibly work being done by another developer. Examine the error message for details and resolve the conflict before re-attempting this action.

- After this change, part of the application unexpectedly stops working.
  The changes include a detail that may be critical to the operation, such as a changed I/O address. Restore the previous version, examine the version log in detail, then repeat your action taking care to avoid changes that will damage the application.

Related Information:

...Show Version Details – View line-by-line what changed in what file.

...Switch to This Version – Set the running application to an earlier (or later) version.

...Merge Version Range – Select a set of previously removed changes to restore.

...Version Changes and Security – How revision changes may affect security configuration.

> Note: Version Control is an optional license feature and may not be available at your site.

Reverse Version Range

Rather than reverse one version change at a time, you can select all the versions within a range. This feature includes the ability to select file-by-file, which changes within that range you want to reverse, leaving some changes intact. For example, in versions 19 and 20 you may have created and configured tag A. In steps 21 and 22, you created and configured tag B. In step 23 you made further changes to tag A.
You can reverse all the actions related to tag A without damaging tag B, by selecting items to reverse in a range that spans steps 19 to 23.

To reverse a range:

1. Right-click on the version either beginning or ending the range.
2. Select Reverse-Version Range from the pop-up menu.

Version	Time Applied
ANDREWH-D23	**Show Version Details** .223
ANDREWH-D22	Switch to This Version .265
ANDREWH-D21	Reverse Version Changes .663
ANDREWH-D20	Merge Version Changes .702
ANDREWH-D19	Reverse Version Range .299
ANDREWH-D18	Merge Version Range .244
ANDREWH-D17	Thu Jul 28, 2011 12:26:22.849

The selected version will be highlighted to help you keep track. Note that this version is also highlighted in the workstation list – you can select a range across workstations as well as across the versions on your workstation.

3. Right-click on the version marking the end of the range you wish to select.

Version Log for ANDREWH (Total of 32 records in log)

Version at the start (or end) of the range is highlighted.

Version	Time Applied
ANDREWH-D23	Thu Jul 28, 2011 13:04:26.223
ANDREWH-D22	Thu Jul 28, 2011 13:03:23.265
ANDREWH-D21	Thu Jul 28, 2011 12:32:02.663
ANDREWH-D20	Thu Jul 28, 2011 12:31:17.702
ANDREWH-D19	Thu Jul 28, 2011 12:29:55.299
ANDREWH-D18	**Reverse Version Range End**
ANDREWH-D17	Cancel Reverse Version Range
ANDREWH-D16	Thu Jul 28, 2011 12:21:27.906

Right-click on the version to mark the other end of the range to be reversed.

4. Optionally, remove items from the list of changes to be reversed.

Note: Some development items will depend on others – use care in your selection.

Select the changes to apply

Select the items you wish to change from the tree of changed items. Click the "Apply" button to apply the displayed changes to those items.

- ⊟ 📁 Source Files (2)
  - ☑ Pages\Overview.SRC
  - ☐ Pages\Station1.SRC
- ⊟ 📁 Tags (2)
  - ☑ Tags\DigitalStatus_256\241.tag
  - ☑ Tags\DigitalStatus_256\62.tag
- 📁 Images
- 📁 Properties
- 📁 Text Files
- 📁 Binary Files

Item Change Display

This panel displays the changes made to a selected item if that item can be analyzed. Text changes are marked by color and symbol as follows:

- / Yellow if the line was removed
+ / Green if the line was added
? / Orange for removed and conflicting lines
X / Red for added and conflicting lines

5. Click Apply and provide a comment in the dialog that appears.

Troubleshooting:

- An error message indicates a conflict.
  The change you are making affects an unsaved change, or possibly work being done by another developer. Examine the error message for details and resolve the conflict before re-attempting this action.

- After this change, part of the application unexpectedly stops working.
  The changes include a detail that may be critical to the operation, such as a changed I/O address. Restore the previous version, examine the version log in detail, then repeat your action taking care to avoid changes that will damage the application.

Related Information:

...Show Version Details - View line-by-line what changed in what file.

...Switch to This Version - Set the running application to an earlier (or later) version.

...Reverse Version Changes - Remove selected revisions from the running application.

...Version Changes and Security - How revision changes may affect security configuration.

Next Steps:

...Merge Version Changes - Restore selected, previously removed revisions to the running application.

...Merge Version Range - Select a set of previously removed changes to restore.

Note: Version Control is an optional license feature and may not be available at your site.

Merge Version Range

If you have made an earlier version current, or have reversed version changes, you can merge selected changes from a range of versions into

your current version.

The steps are identical to those when reversing a version range. The difference is only that selected changes are merged into the current version rather than reversed from it.

To merge a version range:

1. Right-click on a version marking the start of the range and select "Merge version range" from the menu.

2. Right-click on the version marking the end of the range and select "Merge version range end".

3. Optionally remove files from the list of changes that will be merged.

4. Click Apply and provide a comment in the dialog that appears.

Troubleshooting:

- An error message indicates a conflict.
  The change you are making affects an unsaved change, or possibly work being done by another developer. Examine the error message for details and resolve the conflict before re-attempting this action.

- After this change, part of the application unexpectedly stops working.
  The changes include a detail that may be critical to the operation, such as a changed I/O address. Restore the previous version, examine the version log in detail, then repeat your action taking care to avoid changes that will damage the application.

Related Information:

...Show Version Details - View line-by-line what changed in what file.

...Switch to This Version - Set the running application to an earlier (or later) version.

...Merge Version Changes - Restore selected, previously removed revisions to the running application.

...Version Changes and Security - How revision changes may affect security configuration.

Note: Version Control is an optional license feature and may not be available at your site.

ScadaAce Applications

ScadaAce is a VTScada OEM layer, designed for use with Motorola hardware. For those building Motorola Remote Terminal Unit (RTU) systems, the ScadaAce layer simplifies the configuration process.

Upon first starting a new application that is based on the ScadaAce layer, you will be prompted to configure a Motorola Gateway IP tag. Every ScadaAce application must have one (and only one) Gateway IP tag. By convention, its name will always be "GW".

After providing the IP address of the gateway unit, and optionally choosing to enable the system time synchronization feature, click OK. Immediately, you will be prompted to begin entering site information. Click Add.

New sites may be added at any time using the Add New Site button at the bottom of the Sites page. This method will always begin by asking you to place a pin on the map to show where the site is located.

Note also, the Manage Sites button at the bottom of the Sites page. This re-opens the Manage Sites dialog, allowing you to add, edit or delete sites. This method does not prompt for a site location.

The Area field is automatically made the same as the name (RTU_1 in this example). This is very useful when viewing the alarm list since it is the area value that will distinguish alarms in one site from another.
The RTU ID value should match the ID value of the hardware.
Following the initial configuration of the new site, as shown, you will be able to configure the inputs and outputs of this instance.

Related Information:

...ScadaAce Site Tags

New ScadaAce Applications

For those building Motorola Remote Terminal Unit (RTU) systems, the ScadaAce layer simplifies the configuration process.
To create a ScadaAce application:

1. Open the VAM window.
2. Click the Add New button.
3. Select Create New and click OK.
   The New Application Properties dialog opens.
4. In the New Application Properties dialog, enter the following:

   - Name
     The name should describe the application's purpose.

   - Path
     The path will fill automatically.

   - Type
     You must choose ScadaAce, as shown in the example.
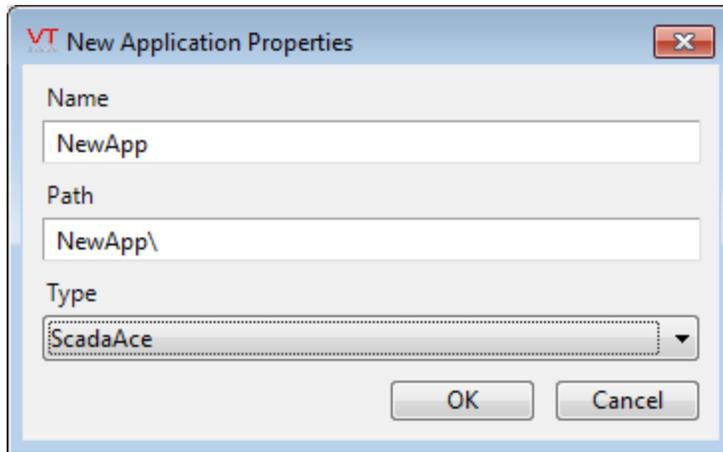
4. Click "OK".

Upon first starting a new application that is based on the ScadaAce layer, you will be prompted to configure a Motorola Gateway IP tag. Every ScadaAce application must have one (and only one) Gateway IP tag. By convention, its name will always be "GW".

After providing the IP address of the gateway unit, and optionally choosing to enable the system time synchronization feature, click OK. Immediately, you will be prompted to begin entering site information. Click Add.

New sites may be added at any time using the Add New Site button at the bottom of the Sites page. This method will always begin by asking you to place a pin on the map to show where the site is located.

Note also, the Manage Sites button at the bottom of the Sites page. This re-opens the Manage Sites dialog, allowing you to add, edit or delete sites. *This method does not prompt for a site location.*

View ScadaAce Sites

An overview of your ScadaAce sites is provided using a Sites page. This provides a list of configured sites (left side of screen) and a map showing the location and current status of each.
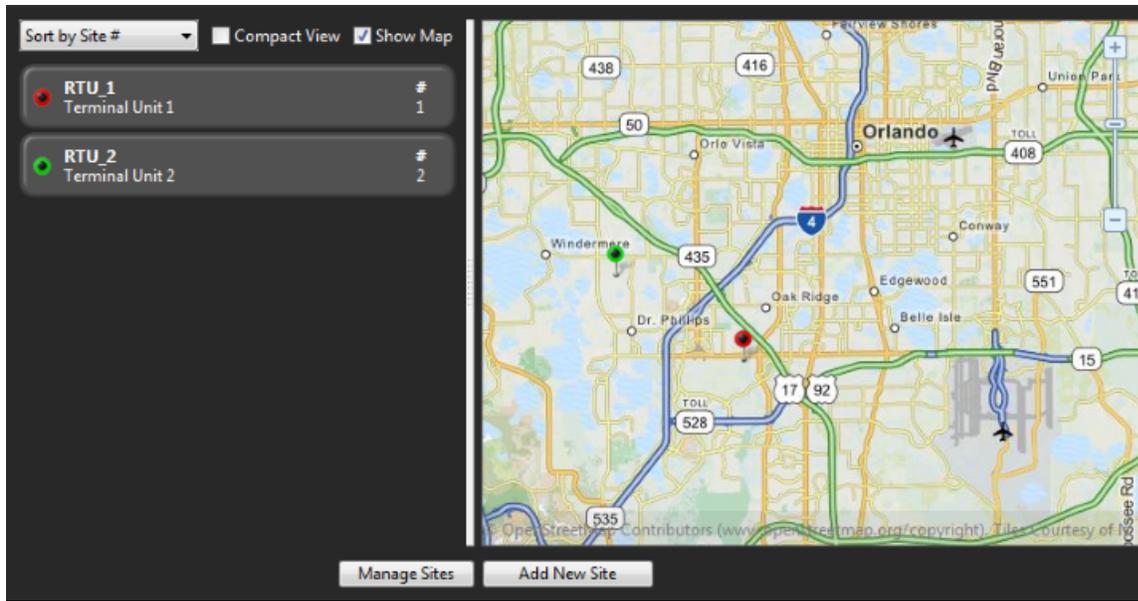
Clicking on any RTU in the left column, or on its matching pin within the map (if a location has been configured) will cause the site details page to open. These are created automatically for you, with each new site that you add.

ScadaAce Site Monitoring Pages - RTU Tab
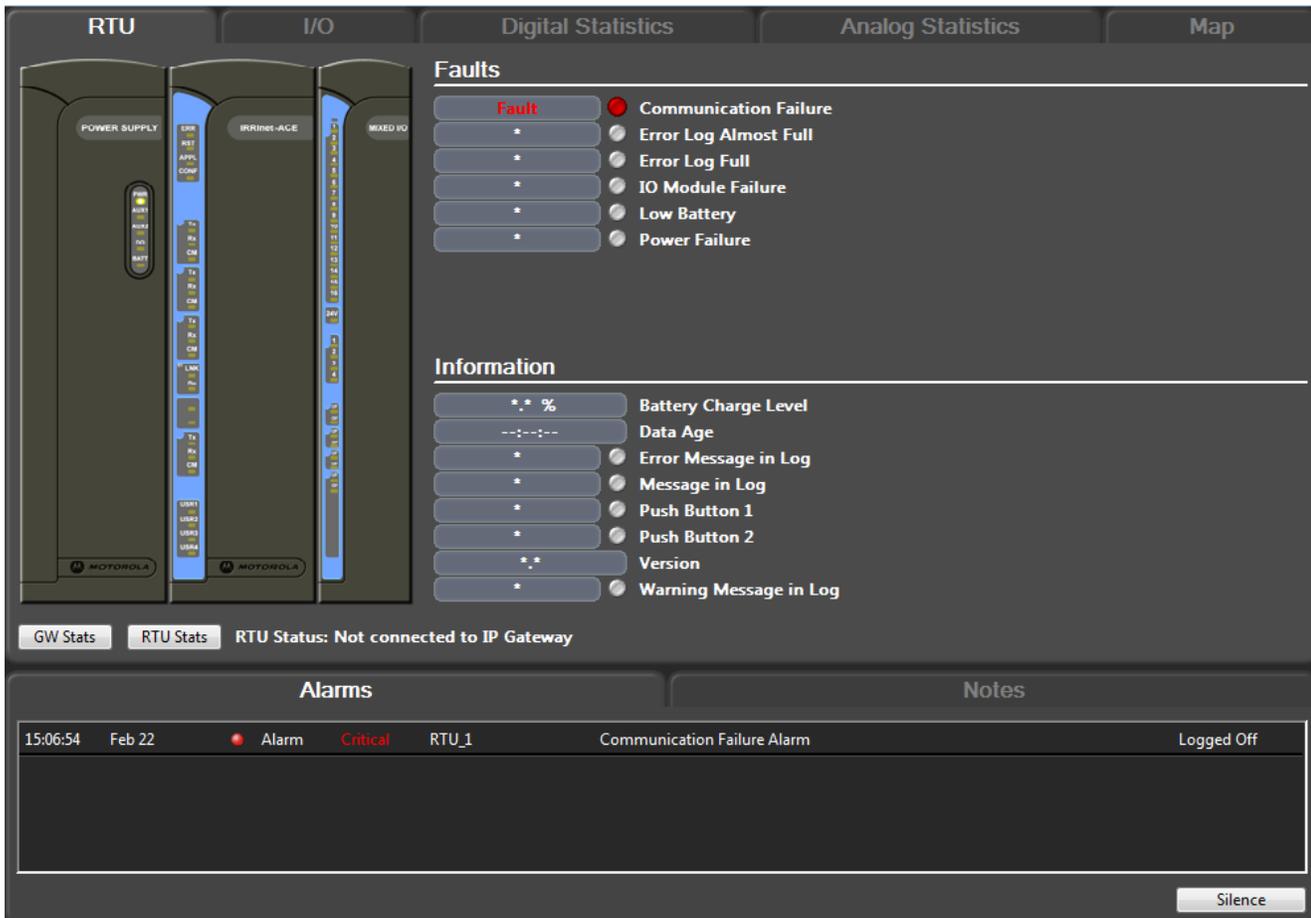
The RTU tab provides general information about the RTU at the site, including the list of faults, device statistics (available via two buttons), and other information.

Along the base of this page (regardless of tab selected in the upper portion) are two tabs: one will show the list of alarms at the site, the other may be used for adding operator notes, specific to this site.

ScadaAce Site Monitoring Pages – IO Tab

All analog and digital I/O tags that belong to the site will be listed on this tab. You can click the Manage I/O button, which will be found at the bottom-center of the list, to re-open the site configuration dialog, where you can add, edit and remove any of the I/O tags.

ScadaAce Site Monitoring Pages – Digital Statistics Tab and Analog Statistics Tab

Statistics are available only if configured when creating the input tags within the site.
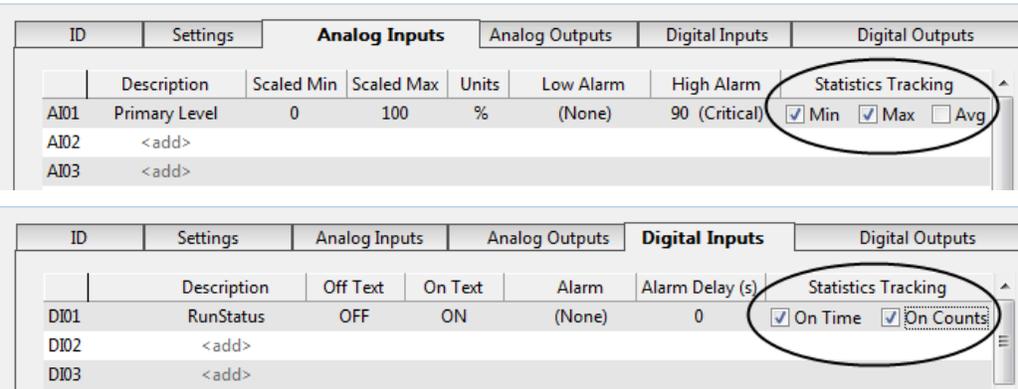
For each statistic tracked, values will be calculated for both "yesterday" and "last 24 hours". Other time ranges may configured by adding History Statistics tags from the Tag Browser. See: History Statistics Tags.

When viewing the statistics, you may choose to view a table showing the values for all collected statistics over a time period, or a table showing the specified statistic, over both time periods.
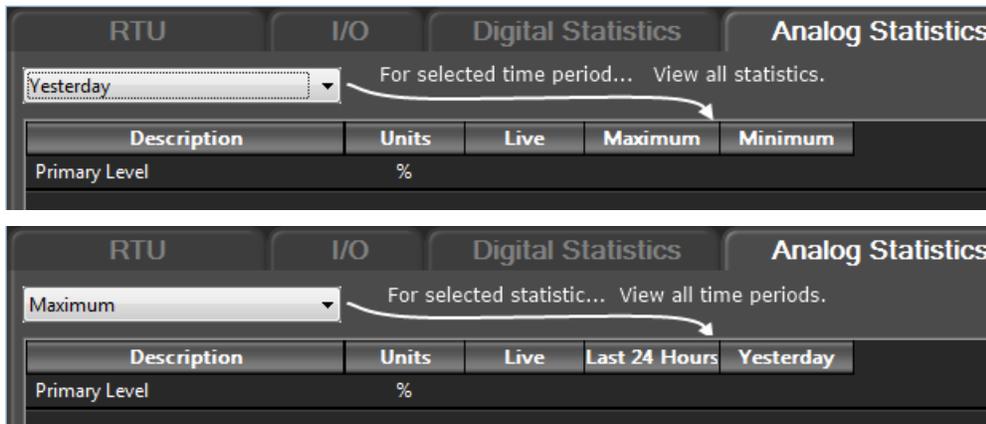
Specialty Pages

Pages described in this section are not included in the default VTScada menu configuration, but may be relevant to your installation.

Related Information:

...Communications Data Page

...Lift Station Page

...Station Page (Legacy)

Communications Data Page

The purpose of the Communications Data page is to display information transmitted and received between the VTScada application and the DataFlow RTUs for your entire system.

```
Continue

 Reply: 12:45:55.15 -  C91xl■
 Reply: 12:45:55.29 -  C92xm■
  Send: 12:45:57.37 -  ■5BRD5C714■
 Reply: 12:45:57.77 -  5C71g■
 Reply: 12:45:58.30 -  5C7w0000000000000000000000000000000000m■
 Reply: 12:45:58.82 -  5BRs5C7w0000000000000000000000000000000000i■
  Send: 12:45:58.83 -  ■CA003^■
 Reply: 12:45:59.21 -  CARxU■
 Reply: 12:45:59.40 -  CA1x4■
  Send: 12:46:01.57 -  ■5BRD5C8I10n■
 Reply: 12:46:01.99 -  5C8I10a■
 Reply: 12:46:02.53 -  5C8p0000000000000000000000000000000005\■
 Reply: 12:46:03.06 -  5BRs5C8p0000000000000000000000000000000005X■
  Send: 12:46:03.08 -  ■02003<■
 Reply: 12:46:03.45 -  02Rx3■
 Reply: 12:46:03.60 -  021xR■
  Send: 12:46:04.32 -  ■03003=■
 Reply: 12:46:04.67 -  03Rx4■
 Reply: 12:46:04.83 -  031xS■
  Send: 12:46:06.07 -  ■ZZ3C04001f■
 Reply: 12:46:06.93 -  04Rx2g■
 Reply: 12:46:07.16 -  05Rx2h■
 Reply: 12:46:07.39 -  06Rx2i■
 Reply: 12:46:07.60 -  07Rx2j■
  Send: 12:46:10.99 -  ■0B11[■
 Reply: 12:46:11.72 -  0B1w00000000000001D0000001001000001000009■
  Send: 12:46:11.77 -  ■0D105Q■
 Reply: 12:46:12.14 -  0D1xd■
  Send: 12:46:12.81 -  ■0D35c■
 Reply: 12:46:13.36 -  0D3k0003400000000000000000001a■
  Send: 12:46:13.37 -  ■0E003O■
 Reply: 12:46:13.74 -  0ERxF■
 Reply: 12:46:13.89 -  0E1xe■
  Send: 12:46:18.09 -  ■10003;■
 Reply: 12:46:18.45 -  10Rx2■
 Reply: 12:46:18.59 -  101xQ■
  Send: 12:46:18.93 -  ■0D5J00Z■
 Reply: 12:46:19.64 -  0D5n100100000000000000000000000001000023■
  Send: 12:46:21.13 -  ■0D5J30]■
 Reply: 12:46:21.86 -  0D5n00000000000000000000000000000003067■
  Send: 12:46:21.91 -  ■11003<■
```

As shown, the Communications Data page displays packets of message
code sent from the VTScada application to the DFS RTUs (preceded by
"Send" and colored blue), and from the DFS RTUs to the VTScada applic-
ation (preceded by "Reply" and colored green), along with the date and
time at which each signal was transmitted or received.
Depending upon your application's configuration, the display of data on
the Communications Data page may scroll quite rapidly. The
Freeze/Continue toggle button at the top of the Communications Data
page enables you to freeze the display so that you can have a closer look

at the communications data. When you are finished reviewing the frozen display, you can hit the button again to unfreeze the display.

Information in the Communications Data Display

The information in the Data Display follows the DFS protocol. For every blue Send message, there should be one or more green Reply messages. A Send message without a matching Reply is a clear indication of a problem preventing the station from answering.

The following figure shows an example of one Send/Reply transaction:

As shown, each Send message will be bracketed by two non-printing characters (the square boxes). Each Reply ends with a single box symbol. You should disregard these non-printing characters in the following guide to the codes.

The first two characters in each message are the address of the DFS RTU. (Again, disregard the boxes.) Looking at the examples, you should see that for every Reply following a Send, these two leading characters match.

One exception to the above is the case of a "ZZ" command in the Send message. This indicates that a group poll is being requested, starting at the address appearing in the 5th and 6th character places of the Send field. The following Replies can be seen to include a series of RTU addresses.


```
Send:  12:46:06.07  -  ■ZZ3C04001f■
Reply: 12:46:06.93  -  04Rx2g■
Reply: 12:46:07.16  -  05Rx2h■
Reply: 12:46:07.39  -  06Rx2i■
Reply: 12:46:07.60  -  07Rx2j■
```

A second exception to the above is the case of a station with a Digipeat path. In this case, the Send will start with the address of the station it is repeating through, followed by 'RD' and then the station address of the RTU being polled.

The third character in the Send message is the module number. Here, "1" will mean module A, "2" means module B and so on, except for "R" which indicates a Radio module.

There is one exception to this pattern: A "0" in the third character position of the Send indicates that a Full Station Read is being requested. All modules in the station should reply with their status, where an "x" will indicate "No Change" and a "y" will indicate "Change".

Normally if a reply indicates that there has been a change, a Send request will go to that module to request its new value.

The final character in either a Send or Reply is a check-bit. This may be any printing character, including punctuation. This is the only character in either the Send or Reply that should ever have punctuation: a character other than a letter or number in any other position is a sign of communication errors.

Freezing the Communications Data Display

To temporarily freeze the communications data displayed on the Communications Data page, simply click the Freeze button. A hold will be placed on the data being displayed. This enables you to review the data closely. The Freeze button will now be labeled, "Continue". Clicking the Continue button will remove the hold from the data and cause it to revert to its regular display.

Note: Clicking the Freeze button does not stop data from being communicated; it merely freezes the display for closer scrutiny. Data continues to be transmitted and received while the Freeze button is depressed.

Lift Station Page

This page will use the information from a Station tag to create complete monitoring and control displays for your pumping stations. The lift station page is normally opened by clicking a Site Draw widget that is linked

to a station tag. The contents of this page will vary according to the type of station tag linked to it. This page may also be referred to as the Station Details Page.

If you choose to add a menu item for a Lift Station page, you must configure three parameters:

- The station tag that will be linked to the component widgets.
- Whether the application's color theme will be applied to the page.
- The color to use for labels. This may be an RGB value in the form, <RRGGBB>, or it can be a numeric value as specified in the VTScada color palette.
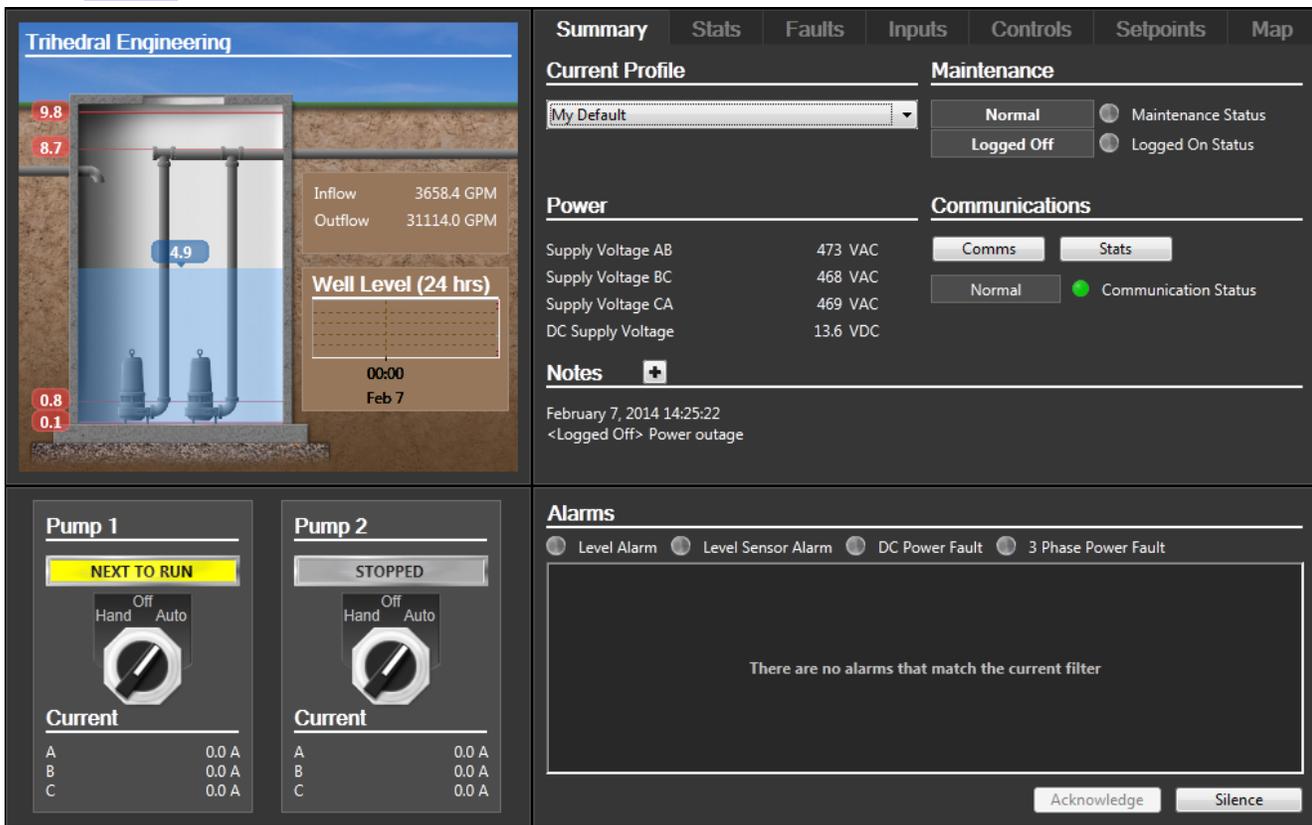
The Station page is an assembly of various monitoring and control drawing methods for the associated station tag. Each component is described in the following sub-topics. The specifics will vary according to whether the station is a MultiSmart, MPE SC series or MPE Duplexer, but most components of the pages will be found for all types of station tag.

Well Details

Includes the Well Flow and Well Plot drawing methods.
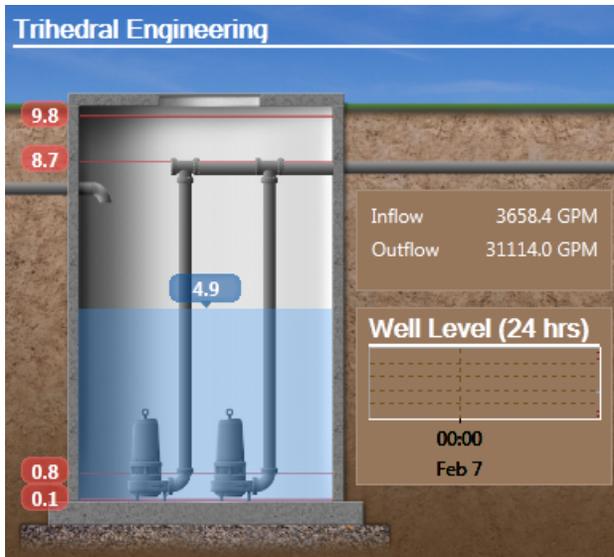
The number of pumps displayed will match the configuration of the associated Station tag. When a pump is running, it will be shown in green. The current water level is represented by the shaded background and the numeric display. High and low alarm setpoints for the pumps are shown by the red lines with matching numeric values.

The current flow, both in and out, is displayed as well as a trend of the past 24 hours, showing well level and alarm levels.

Pump Control

Provides a Hand-Off-Auto control for each pump in the station.

The label indicates whether the pump is running (displayed in green), stopped (gray) or if it will be the next to run (yellow).

Station Summary

The Station Summary varies according to the type of station tag.

MultiSmart Station Summary:

For the MultiSmart Station, six operational profiles may be defined in the Setpoints tab. Profiles are used to store lead and lag setpoints to be used under varying operating conditions such as storms, drought, etc. The profile selection in the summary page shows which is active and select which to use.

If your station uses the Modbus protocol, only the profile number will be shown. If using the DNP3 protocol, you will see the profile description displayed.

Operating notes may be added using the + button. This opens the standard VTScada notebook dialog. Use these to create a permanent record of operational and maintenance tasks.

For comparison, the station summary pages for the MPE Duplexer Station and the MPE SC Station are shown in the following images. Note the lack of a Setpoints tab for the MPE Duplexer Station. The MPE SC Station has a Setpoints tab, but only a single profile.

MPE Duplexer Station Summary:
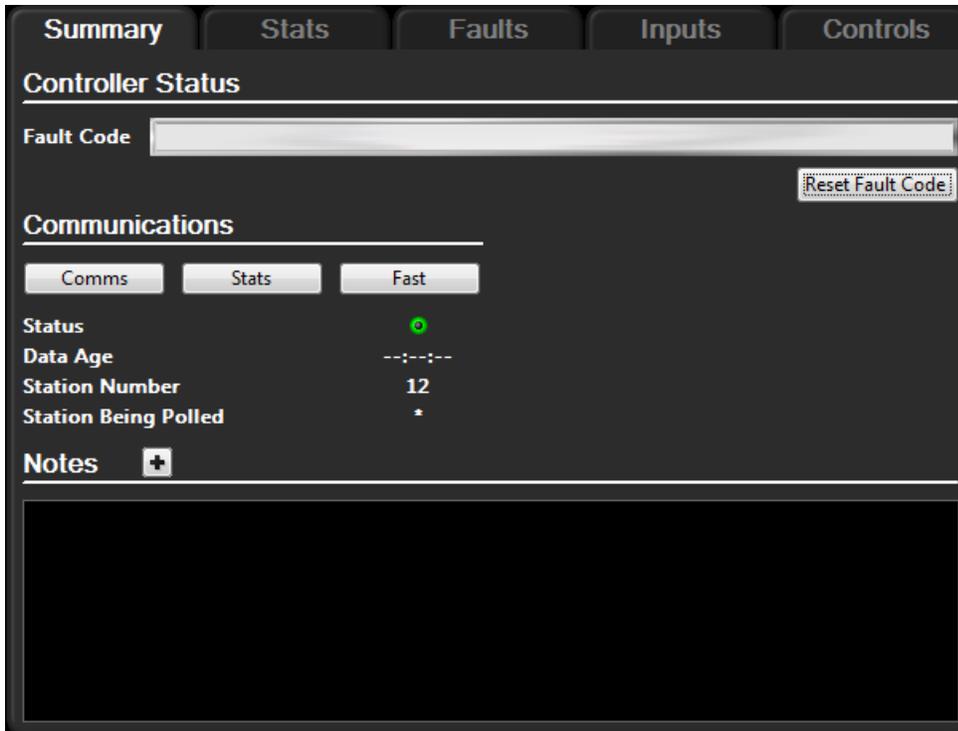
MPE SC Station Summary:
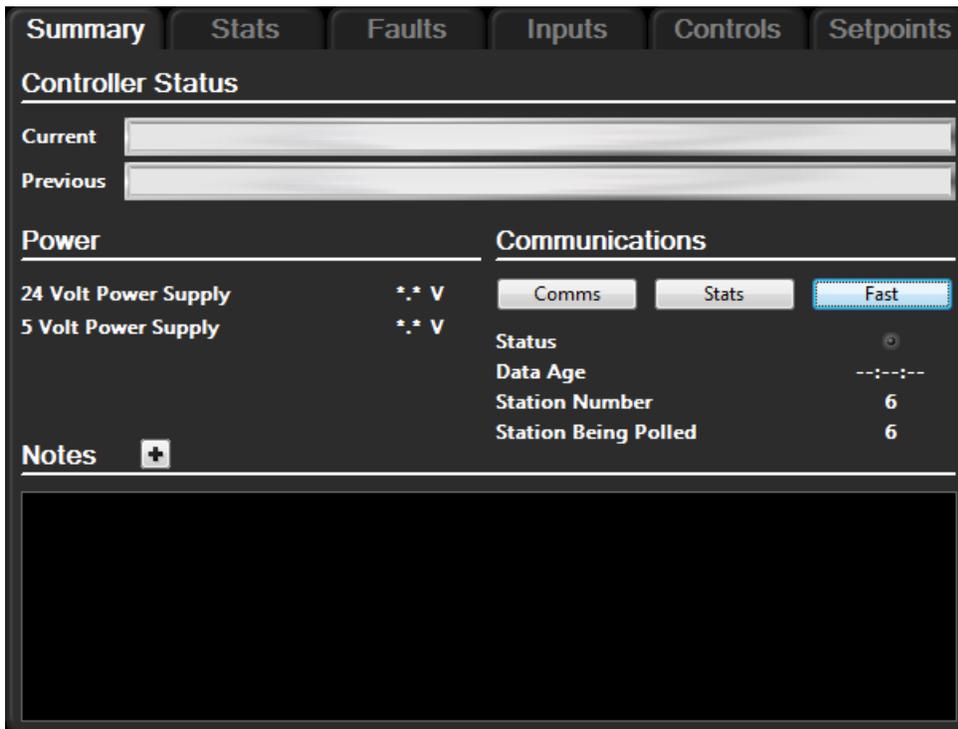
Station Alarm List

Shows all current faults in the station. You may acknowledge or silence alarms with the buttons provided. A blinking indicator shows that the

alarm has not received a reset, sent from the Faults tab.
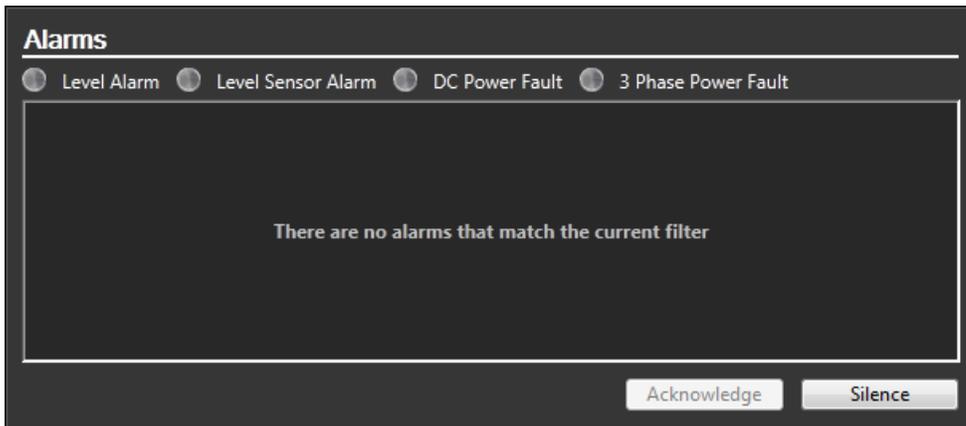
Station Statistics

An extensive collection of statistics is collected and displayed for each individual pump, the station as a whole, and the well.
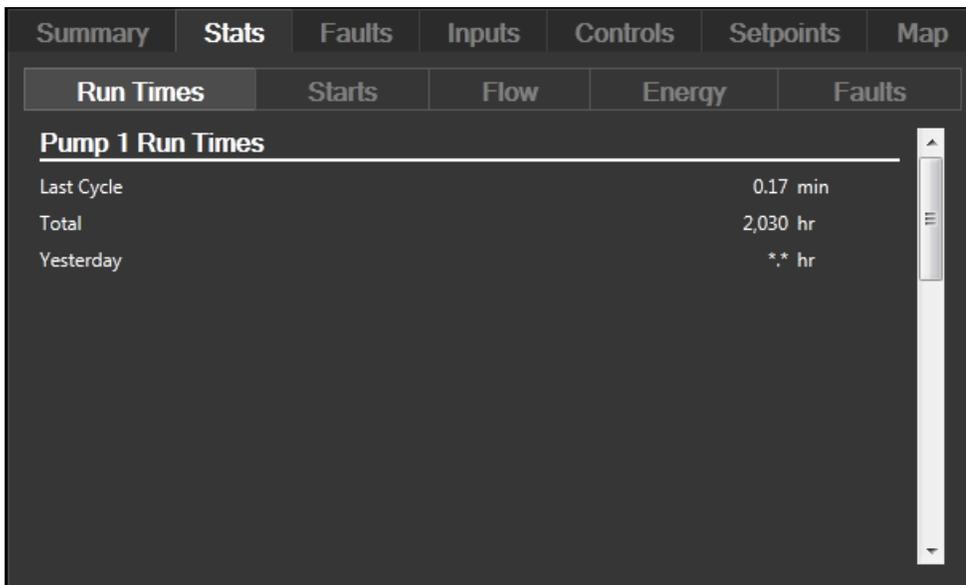
Tabs across the top of the display provide access to different sets of statistics. The number of tabs and the statistics gathered will vary with the type of station controller.

Within any tab, statistics are grouped by item of equipment (pumps, well, station as a whole). The number of equipment items shown will vary depending on your screen size. If there is not enough room to show all items, a scroll bar will be provided so that you can select which set of statistics to view.

Station Faults and Fault Reset

There are two pages within the faults tab: a list of faults, grouped by component and a set of fault reset controls.

Indicators will show red when a fault is present and green otherwise.

After a fault has been reset, the corresponding alarm must still be acknowledged in VTS.
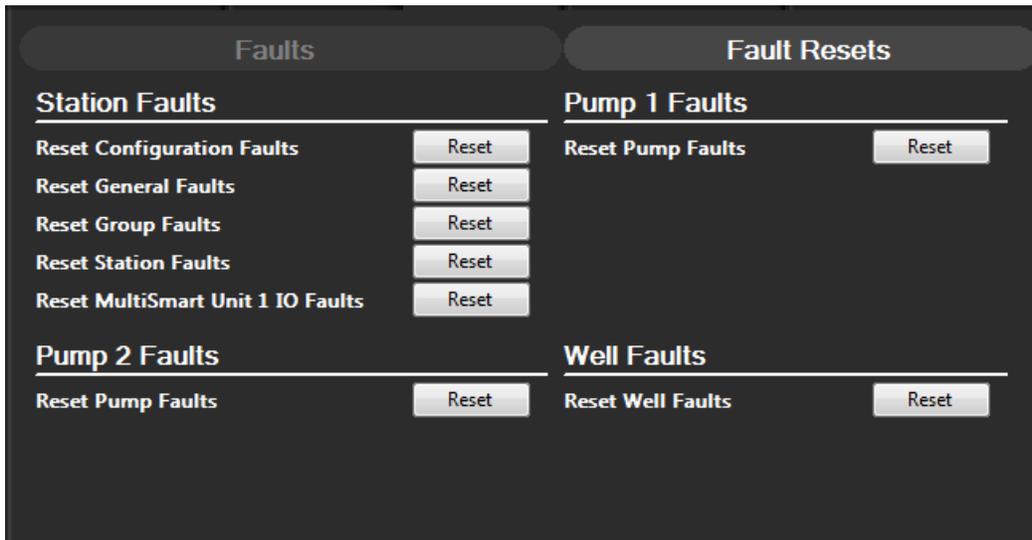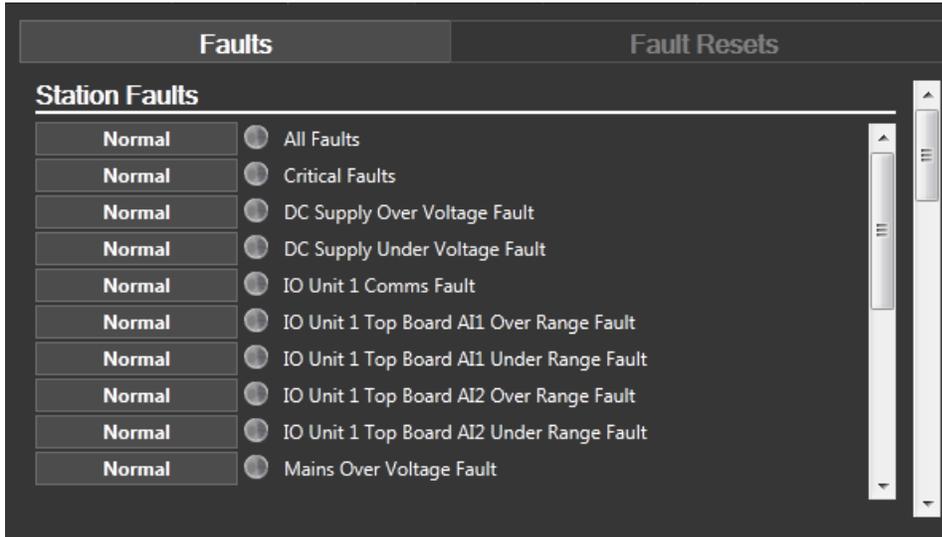
Station Inputs

Provides a list of all input tags associated with the station. The list is divided into two pages – digitals and analogs (digitals shown below analogs). For the digitals, a color dot indicates the tag's value: green is used for zero, red for one and gray indicates 'no data'. The analog tags are shown with their current value.
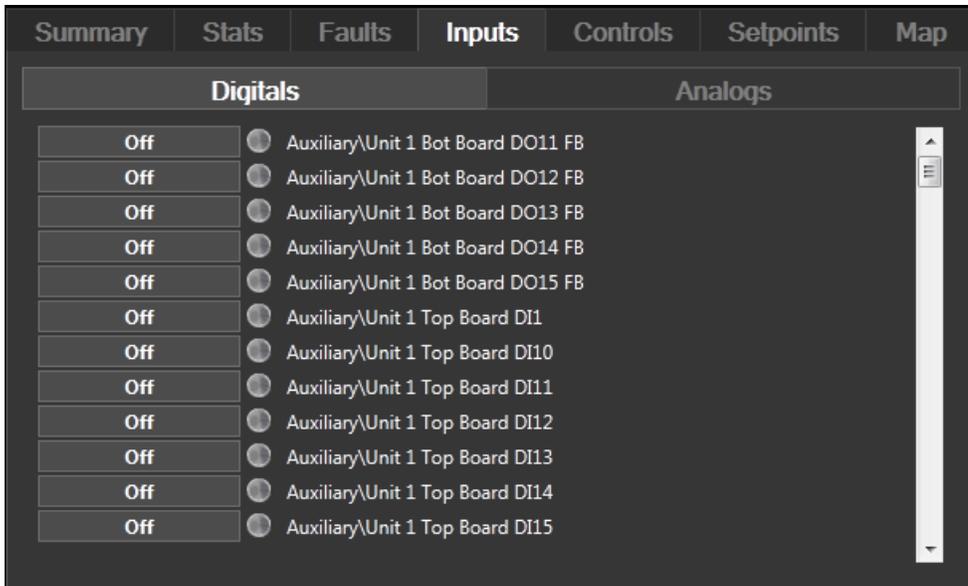
Station Controls

The station controls page provides access to all output tags in the station, grouped by digitals and analogs.

To operate a digital control, click the corresponding button. To operate an analog control, click in the appropriate field as shown, then type the new value that should be written to the station.
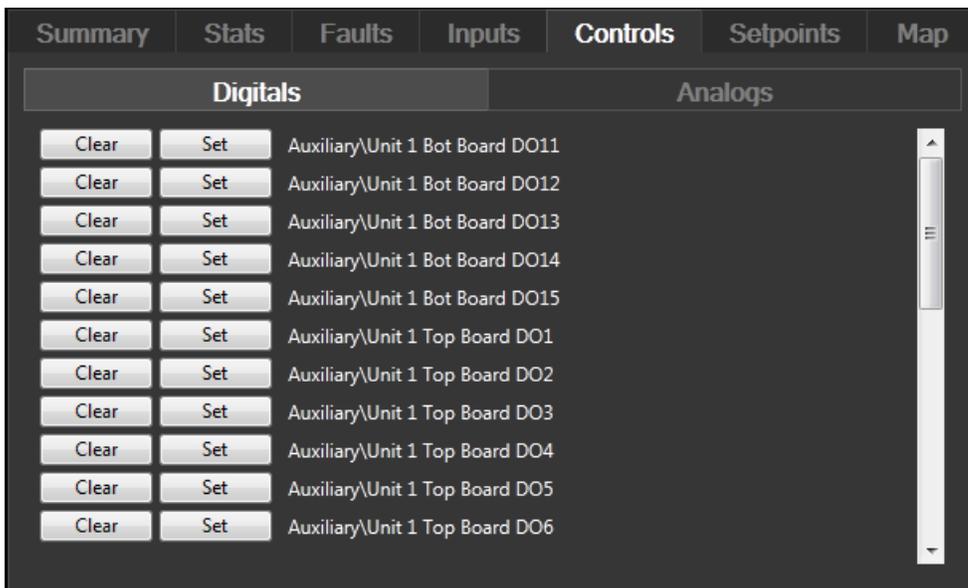
Station Setpoints

For MPE SC Stations and MultiSmart Stations. Sample image shows the MultiSmart's Station Setpoints display.

Set the lead and lag setpoints for the pumps. One profile can be set for the MPE SC Station, while six profiles may be created for the MultiSmart, each with a different configuration of setpoints. These are provided to allow operators to quickly switch operating parameters in response to changing conditions such as an approaching rainstorm, etc. The various profiles are used only for changing the pump lead and lag setpoints. Alarm setpoints are accessed only through the Current tab of the display. For reference, the display also shows the current depth of water in the well as a blue background.

You can adjust the setpoints using the sliders, or for greater accuracy, you may type values into the fields above each setpoint slider.
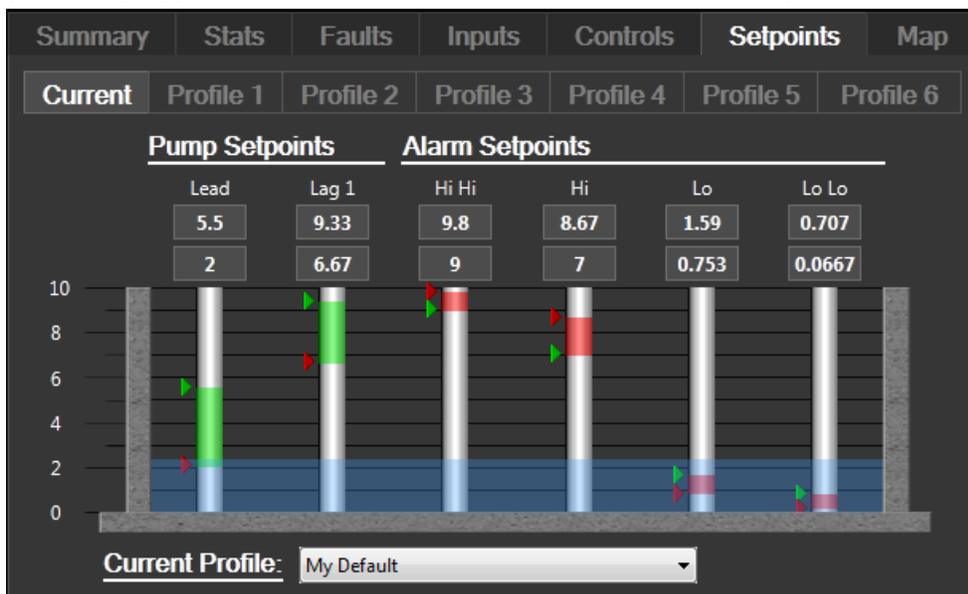
Related Information:

...Site Draw Widget

Station Page (Legacy)

The station page is considered obsolete, having been replaced by the newer Site Details Page.

If you would prefer to continue using station pages, add the property UseOldSiteDialog to the System section of your application's Settings.Dynamic file, and set the value of that property to "1".

Each time you create a new Polling Driver or a new Data Flow RTU driver tag for your application, VTScada automatically creates a station page for the new station.

Station pages are launched when the station symbol for the Polling Driver or Data Flow RTU driver tag is clicked. You may change the page that is launched when a station symbol is clicked (i.e. you may open a custom page you have created for the station) by changing the station symbol's configuration.

The following images are from VTS version 10.2. Support for legacy station pages under VTScada 11 and later is not guaranteed.
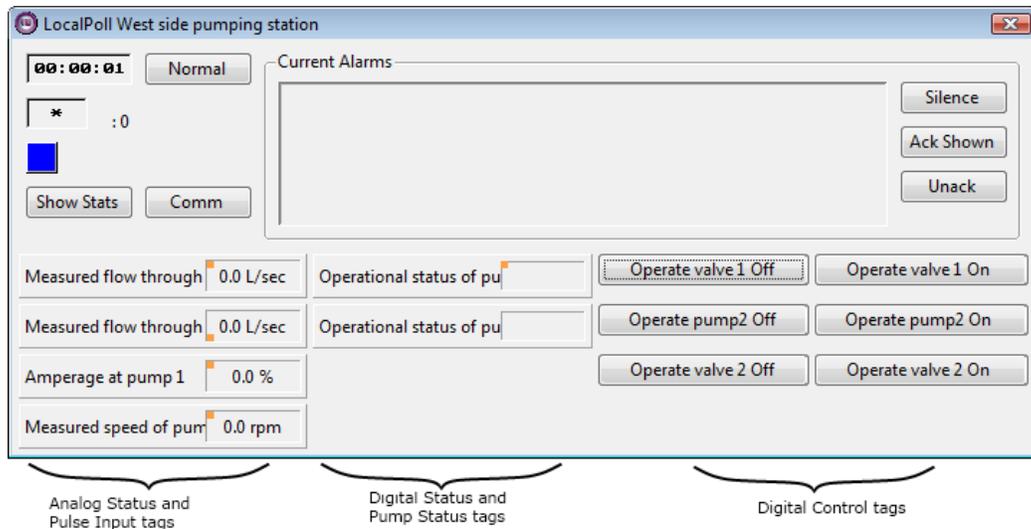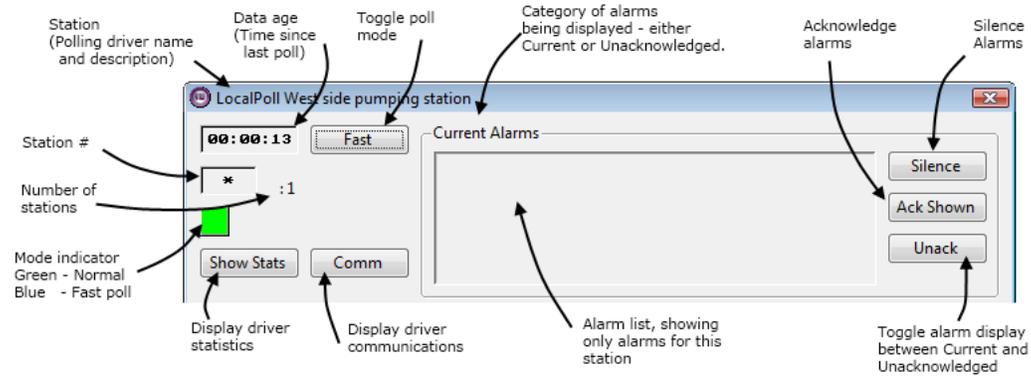
General Reference

Technical notes for:

Colors, Fonts and Graphics

...VTScada Color Palette

...Constants for System Colors

...Color Theme Definition

...Fill Patterns

... Font Character Sets

...GUI Object Return Codes

...Line Types

Time and Date

...predefined Date Codes

...Date Formatting Strings

...predefined Time Formats

...Time Formatting Codes

...VTScada and Time Synchronization

Value Types and Language Support

... SQL Data Types

...Database Type Codes used in the ODBC Manager

...VTScada Value Types – Numeric Reference

...Value and Type Conversions

...Language Support

...Using a Non-English Character Set

Help Files, Function Support, Other...

ASCII Constants

The following constants are defined at the \System layer. In code that parses user-input strings, you should use these constants rather than creating your own buffers for common key values or key combinations.

Name of Constant	Contents	Description
CR	MakeBuff(1, 13)	Carriage return
ESC	MakeBuff(1, 27)	Escape key
LF	MakeBuff(1, 10)	Line feed
NULL	MakeBuff(1, 0)	Null byte
TAB	MakeBuff(1, 9)	Tab key
CRLF	Concat(MakeBuff(1, 13), MakeBuff(1, 10))	CR/LF pair
UpArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x48))	Up arrow key
DownArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x50))	Down arrow key
LeftArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x4B))	Left arrow key
RightArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x4D))	Right arrow key
AltLeftArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x9B))	Alt and left arrow
AltRightArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0x9D))	Alt and right arrow
SUpArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0xB8))	Shift and up arrow
SDownArrow	Concat(MakeBuff(1, 253), MakeBuff(1, 0xC0))	Shift and down arrow

PageUp	Concat(MakeBuff(1, 253), MakeBuff(1, 0x49))	Page up key
PageDown	Concat(MakeBuff(1, 253), MakeBuff(1, 0x51))	Page down key
HomeKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0x47))	Home key
EndKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0x4F))	End key
DeleteKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0x53))	Delete key
SPageUp	Concat(MakeBuff(1, 253), MakeBuff(1, 0xB9))	Shift and page up
SPageDown	Concat(MakeBuff(1, 253), MakeBuff(1, 0xC1))	Shift and page down
SHomeKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0xB7))	Shift and home
SEndKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0xBF))	Shift and end
CtrlPageUp	Concat(MakeBuff(1, 253), MakeBuff(1, 0x84))	Ctrl and page up
CtrlPageDown	Concat(MakeBuff(1, 253), MakeBuff(1, 0x76))	Ctrl and page down
CtrlHome	Concat(MakeBuff(1, 253), MakeBuff(1, 0x77))	Ctrl and home
CtrlEnd	Concat(MakeBuff(1, 253), MakeBuff(1, 0x75))	Ctrl and end
CtrlBKey	MakeBuff(1, 2)	Ctrl and B
CtrlCKey	MakeBuff(1, 3)	Ctrl and C
CtrlDKey	MakeBuff(1, 4)	Ctrl and D
CtrlIKey	Concat(MakeBuff(1, 253), MakeBuff(1, 0xB5))	Ctrl and I
CtrlLKey	MakeBuff(1, 12)	Ctrl and L
CtrlNKey	MakeBuff(1, 14)	Ctrl and N
CtrlOKey	MakeBuff(1, 15)	Ctrl and O
CtrlUKey	MakeBuff(1, 21)	Ctrl and U
CtrlVKey	MakeBuff(1, 22)	Ctrl and V
CtrlXKey	MakeBuff(1, 24)	Ctrl and X
CtrlYKey	MakeBuff(1, 25)	Ctrl and Y
CtrlZKey	MakeBuff(1, 26)	Ctrl and Z

VTScada Color Palette

Colors in VTScada have been specified using RGB values since the release of version 10.2. Prior to that version, the color palette, described in this topic, provided the full range of colors available for use.

All VTScada functions that require a color value will still recognize values from the color palette. Use the following chart as a guide when selecting which number to use.

Color Theme Definition

You can add your own color themes to VTScada. These are stored in the Setup.INI file, within the [Themes] section.

Themes are defined by numeric values for Hue, Saturation, Brightness and Contrast in that order, following the theme name. The scale for those values is based on the tools that were available at the time that themes were introduced, and do not use plain HSL values. For reference, note that the Plum theme with values of 0,1,1,1 is the base color.

For example:

```
[Themes]
Theme = Grey,0,0,1.1,1
Theme = Navy,-15,2,0.7,1
Theme = Burgundy,110,2,0.5,1
```

Hue ranges from –180 to +180 and represents the departure from the color <FF343468> (a shade of purple).
Color saturation, brightness and contrast all range from zero to four.

The following example describes the best available method for defining a new theme.

1. Open the Idea Studio and add a grayscale image to the page.
   The choice must be an image, not a widget or shape.

2. Open the properties dialog for the image.

3. Click the Adjust button to open the Adjust Image dialog.

4. Use the Colorize control to set the base color of the image to RGB value FF343468. (Red 53, Green 52, Blue 104, Opacity 255)

5. Drag the Hue and Saturation sliders until the color of the sample image changes to approximately what you want.

6. Drag the Brightness and Contrast sliders to make fine adjustments to the sample image.

7. Note the values for Hue, Saturation, Brightness and Contrast in order.

8. Edit your Setup.INI file to add a new entry.

Remember that you will need to stop all your applications and restart VTScada itself before changes to Setup.INI take effect.

Image

| Select |
| Adjust |

Color Replace

☐ Tile Image and S

VT Adjust Image

🎨 ▶	Colorize		Base color: #FF000040
123 ▶	Mirror Image	☐	
123 ▶	Antialias	☑	
123 ▶	Rotation Angle	⭘————	0
123 ▶	Opacity	————⭘	1
123 ▶	Brightness	——⭘——	1.58
123 ▶	Contrast	——⭘——	2.18
123 ▶	Hue	———⭘—	151.49
123 ▶	Saturation	———⭘—	3.72
123 ▶	Add Tint	⭘————	0
123 ▶	Tint Hue	——⭘——	0

| Reset to Defaults | | OK |
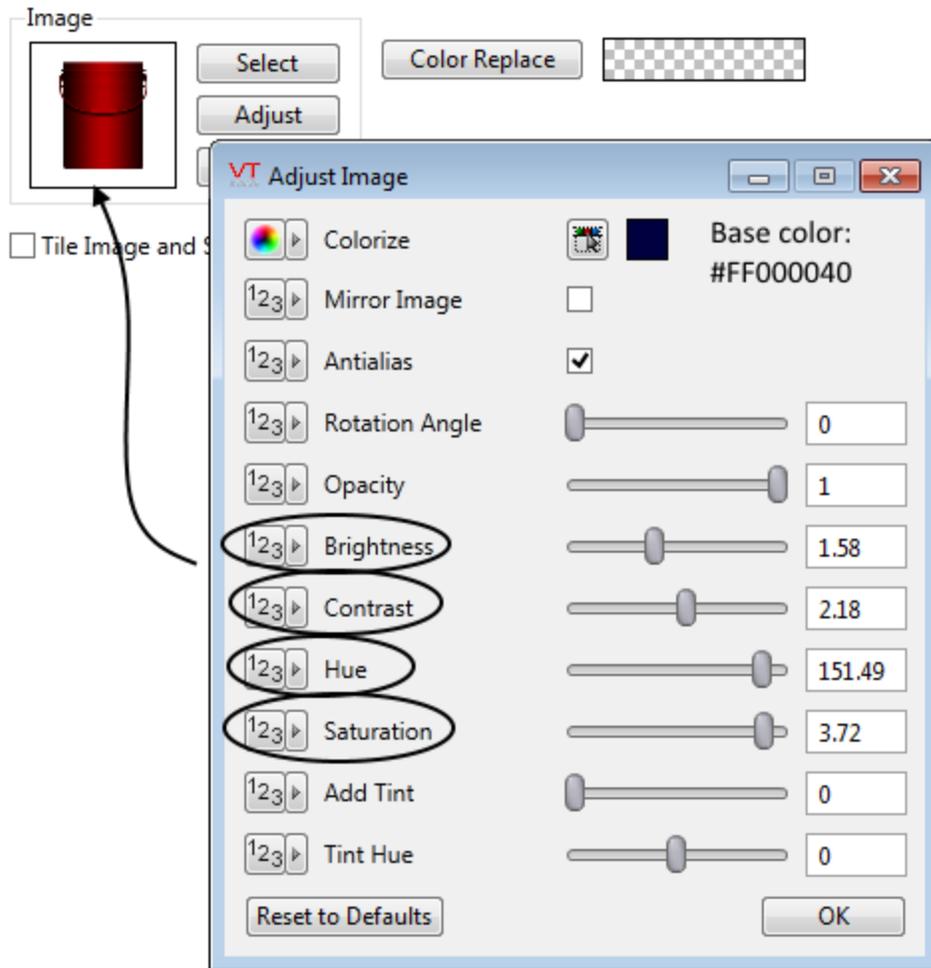
Setting the values.

The entry in Setup.INI:

```
Theme = AutumnGlow,151.5,3.72,1.58,2.18
```

 The resulting theme. The high contrast setting is most clear in the controls along the bottom.

Constants for System Colors

The following constants should be used in place of their numeric values for any function that will accept a system color designation.

Constant	Value
#SYSCOLOR_SCROLLBAR	−2
#SYSCOLOR_DESKTOP	−3
#SYSCOLOR_ACTIVEBAR	−4
#SYSCOLOR_INACTIVEBAR	−5
#SYSCOLOR_MENUBACK	−6
#SYSCOLOR_WINBACK	−7
#SYSCOLOR_WINFRAME	−8

#SYSCOLOR_MENUTEXT	-9
#SYSCOLOR_WINTEXT	-10
#SYSCOLOR_ACTIVETEXT	-11
#SYSCOLOR_ACTIVEBORDER	-12
#SYSCOLOR_INACTIVEBORDER	-13
#SYSCOLOR_MDIBACK	-14
#SYSCOLOR_SELECTEDBACK	-15
#SYSCOLOR_SELECTEDTEXT	-16
#SYSCOLOR_BUTTONFACE	-17
#SYSCOLOR_BUTTONSHADOW	-18
#SYSCOLOR_GRAYEDTEXT	-19
#SYSCOLOR_BUTTONTEXT	-20
#SYSCOLOR_INACTIVETEXT	-21
#SYSCOLOR_BUTTONHLITE	-22

Integrating Custom Help Files into VTS

VTScada provides you with the power to integrate custom help files into your VTScada applications. With your own help file, you might choose to:

- Link individual tag instances to topics in a custom help file.
- Link pages to topics in a custom help file.
- Link pages, widgets or user-defined tags to any of the 100 topic files that have been set aside in the VTScada help system for customer use. You may change the text in these files as required.
- Override the default VTScada help files so that your custom help file opens when your OEM-based application is selected in the VAM and the VAM's Help button is clicked.

VTScada provides support for custom help files in a variety of formats, but not for all possible formats. The Microsoft® .HLP and .CHM formats are fully supported. HTML-format help files are produced by many help authoring tools, but each uses its own code for context sensitive help. It is not possible for VTScada to work with every format. Support is provided for both the Doc-To-Help® DotNet format and for the Flare® HTML5 help format. (Both DocToHelp and Flare are products of MadCap Software Ltd.)

Add your topics to the built-in VTScada help system.

- One hundred topics have been set aside for you to edit within the VTScada help system. Each is named for the mapping ID value assigned according to the pattern, UserTopic100, UserTopic101, etc. Instructions for finding and editing these topic files are provided in User-Topics in the VTScada Help Folder.

Create your custom help file or system and assign topic mapping ID values:

- You will need to obtain a third-party help authoring tool. For .HLP and .CHM format help files, there are many programs to choose from and prices vary considerably. If you intend to create a DotNet or HTML5 format help system, only DocToHelp® and Flare® are supported by VTScada.
- Topic mapping ID values are typically created within the help authoring tool and assigned to topics there. VTScada's numbering system begins at 10,000. In addition, values 100 through 200 are reserved for user-defined topics that you may add to the VTScada help files. To avoid any conflict with the VTScada ID values, create mapping IDs that are less than 10000, excluding those in the 100 to 200 block.

Install your custom help file:

- If using the .HLP format, your help compiler will generate several files. Send both the .HLP file and the .CNT file with your application. Save both files to the VTScada installation folder, not to the application folder.

- If using the .CHM format, send only the .CHM file. Save this to the VTScada installation folder, not to the application folder.
- If using the DotNet format or the HTML5 format, copy the folder structure to a new sub-folder that you create within your VTScada installation folder (ex: C:\VTScada\MyHelpFolder)

Link tag instances to topics in your help file

Note: If using a UserTopic that you have edited in the VTScada help system, skip steps 1 and 2.

1. Use the advanced mode of the Edit Properties dialog to create a local copy of the property, HelpFile.
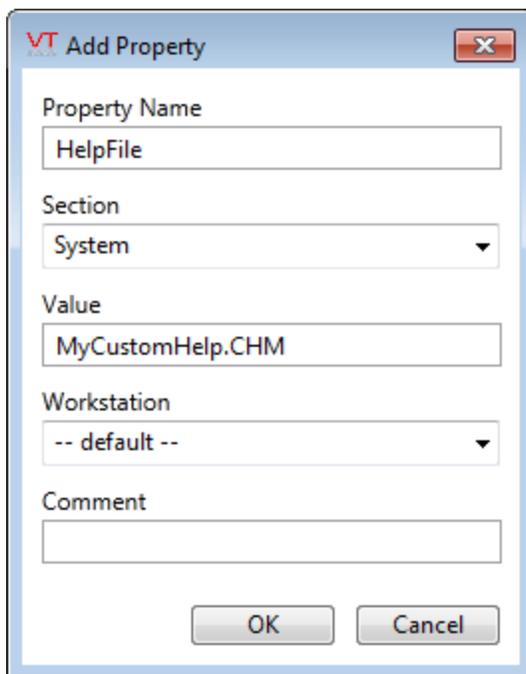
2. Set the value of that property as follows:
   - If using the .HLP or .CHM format, set the property to the name of the file, including the extension. It is assumed that the file will be stored in the VTScada installation folder, not in the application folder.
   - If using the NetHelp (DocToHelp) format, set the value of HelpFile to "MyHelpFolder\NetHelp", replacing "MyHelpFolder" with the name of the folder you created to store the help files within.

- If using the HTML5 (Flare) format, set the value of HelpFile to "MyHelp-folder\MadCapWebHelp", again replacing "MyHelpFolder" with the name of the folder you created to store the help files within.

3. Open the properties dialog of a tag instance.
4. Set the Help Key property to the value of topic mapping id withing your help file, matching a topic to the tag.

Operators can now right-click on widgets linked to that tag, then click Help in the menu that appears, in order to view your topic.
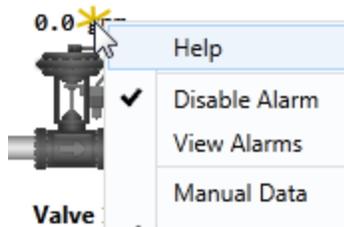
Link pages to topics in a custom help file

To create a link between your page and a topic in a custom help file, edit the page's source code, then import file changes. You can define the help file and a specific topic for any page by adding the following line within the module's Main state:

```
SetHelp(Self(), Help File Name, Mapping ID value);
```

Replace the parameter, "Mapping ID value" with the ID value in your help file that matches the topic to be opened when an operator presses F1. The help file name parameter should be replaced using the same rules as for the HelpTopic application property:

- If using the .HLP or .CHM format, set the parameter to the name of the file, including the extension. It is assumed that the file will be stored in the VTScada installation folder, not in the application folder.
- If using the NetHelp (DocToHelp) format, set the parameter to "MyHelpFolder-\NetHelp", replacing "MyHelpFolder" with the name of the folder you created to store the help files within. Include the quotation marks.
- If using the HTML5 (Flare) format, set the parameter to "MyHelp-folder\MadCapWebHelp", again replacing "MyHelpFolder" with the name of the folder you created to store the help files within. Include the quotation marks.

- If using a UserTopic that you have edited in the VTScada help system, set the parameter to \DevHelpFile. Do not add quotation marks.

Override the default help file for the VAM's help button

A legacy feature of VTScada is the ability to have the Help button in the VAM open your help file instead of the default. Your help file will open only if the operator has first selected (but not necessarily started) an application that is based on an OEM layer rather than the VTScada library layer. Given that requirement, and the fact that it is rare for an operator to have access to the VAM, let alone be selecting applications, then clicking the help button, the following information is provided mostly for interest's sake.

1. Stop all applications and stop VTScada.
2. Open the file, Setup.INI, in the VTScada installation folder.
3. Edit the OEM section property, OEMHelp, setting its value according to the rules described twice already in this topic. Do not use quotation marks, regardless of whether you are setting the value to MyHelpFile.CHM, MyHelpFolder\NetHelp or MyHelpFolder\MadCapWebHelp.
4. Restart VTScada.
5. Select an application that was built on an OEM layer.
6. Click the VAM's help button. Your help file should open to the default welcome page.

User-Topics in the VTScada Help Folder

One hundred mapping ID values have been set aside in the VTScada help system, which you can use to link to topics you create. These ID values can be used with your pages or with individual tag instances. When an operator presses F1 while viewing your page, or right-clicks on a tag-linked widget and opens the Help option, your topic will be displayed. You must create the topic files, using the provided template as a guide. The topic files must be saved in the VTSHelp\Content\UserTopics folder. Each topic must be named according to the pattern, "UserTopic100.htm" where the numeric portion of the name (100 in this example) matches

the ID value set aside for the topic. You will use that ID for your page or tag. One hundred ID values have been set aside, therefore your topics will be named, "UserTopic100.htm" through "UserTopic199.htm".

> Note: These user topics cannot be displayed in the menu of the VTScada help system, nor will your content be found in response to a search in the help system. They can be displayed only in response to an F1 request.

Edit your topic files:

1. Assuming that you have installed VTScada in the folder, C:\VTScada, navigate to the folder, C:\VTScada\VTSHelp\Content\UserTopics.

2. Copy the file, UserTopicTemplate.htm to UserTopicN.htm where N is a number from 100 to 199.
   Start with UserTopic100.htm and work through the set in order.

3. Use a text editor, or a web page editor to open the new file, UserTopicN.htm. Do not use MS Word, or any other word processing program that will add its own formatting characters.

4. Search for the keyword, "+++Start"
   The first 300 or so lines in the topic are required to load the CSS files, JavaScript, menu and other parts of the VTScada help system. Do not edit anything before "+++Start".

5. Replace the text between the <H1> header tags. Do not use headers other than <H1>.

6. The content that you are replacing includes HTML tags and CSS style tags that show styles used in the VTScada help system. You should use these as needed for your own text and images.
   All the text from "+++Start" through to "end---", should be replaced.
   "+++Start" and "end---" should be removed.

7. Save your file.

To link a tag to a user topic you created:

1. Select the topic file and note the numeric portion of the name.
2. Open the tag properties dialog and enter the number from the topic into the Help Search Key field of the ID tab.
3. Close the tag properties dialog.
4. To test, right-click on any widget linked to the tag, then click on "Help" in the menu that opens. The matching topic should open in your default browser.

To link a page to a user topic:

1. Select the topic file and note the numeric portion of the name.
2. Open the source code of the page to be linked to this topic.
3. In the Main state, add the following line of code, replacing the number with the one from the file.

```
SetHelp(Self(), \DevHelpFile, 101);
```

4. Click the Import File Changes button for this application in the VAM.

When updating your copy of VTScada, only the template file will be replaced. Your own topic files will remain but, within these the top menu of VTScada help topics may be out of date. You can bring your topics up to date by copying everything before the "+++Start" marker and after the "End---" marker from the template file, replacing the same within your UserTopicN.htm files.

Database Type Codes used in the ODBC Manager

The following numeric values are used by various functions in the library to select formatting characteristics appropriate to each database type.

Value	Meaning
0	MS SQL

1	MS Access
2	Oracle
3	MySQL
4	SyBase

predefined Date Codes

Use any of the following numeric date codes to format a data as shown. If you require a custom format, you can build one using the text codes shown in Date Formatting Strings.

Dates using these codes will always be displayed in English, regardless of system configuration.

All examples showing Monday, August 13, 2012.

Note: Use only the number in the first column – the second two describe the result of the code in the first column. They are not codes that you can use in the function.

Date Code	Example	Description
0		no date
1	120813	yyMMdd
2	08/13/12	MM/dd/yy
3	08-13-12	MM-dd-yy
4	Aug 13, 2012	MMM d, yyyy
5	August 13, 2012	MMMM d, yyyy
6	13 Aug 12	dd MMM yy
7	13 Aug 2012	dd MMM yyyy
8	13/08/12	dd/MM/yy
9	13-08-12	dd-MM-yy
10	13Aug12	ddMMMyy

11	13Aug2012	ddMMMyyyy
12	Aug 13/12	MMM d/yy
13	20120813	yyyyMMdd
14	08/13	MM/dd
15	08-13	MM-dd
16	08/12	MM/yy
17	08-12	MM-yy
18	08/2012	MM/yyyy
19	08-2012	MM-yyyy
20	Aug 13	MMM d
21	August 13	MMMM d
22	Aug 2012	MMM yyyy
23	August 2012	MMMM yyyy
24	13/08	dd/MM
25	13-08	dd-MM
26	Aug	MMM
27	August	MMMM
28	12-08-13	yy-MM-dd
29	12/08/13	yy/MM/dd
30	2012-08-13	yyyy-MM-dd
31	2012/08/13	yyyy/MM/dd
32	12-W35-01 ([1])	yy-WeekOfYear-DayOfWeek ([2])
33	12/W35/01	yy/WeekOfYear/DayOfWeek
34	2012-W35-01	yyyy-WeekOfYear-DayOfWeek
35	2012/W35/01	yyyy/WeekOfYear/DayOfWeek
36	12226	yyDayOfYear
37	2012226	yyyyDayOfYear

38	12–226	yy–DayOfYear
39	12/226	yy/DayOfYear
40	2012–226	yyyy–DayOfYear
41	2012/226	yyyy/DayOfYear
42	Mon, 13 Aug 2012	ddd, d MMM yyyy

([1]) Week Of Year is preceded by the character W.

([2]) "DayOfWeek" and "WeekOfYear" are descriptions rather than format codes that you could use.

Related Information:

predefined Time Formats – Use for displaying time.

Date – The function that uses the codes listed above.

Date Formatting Strings

To build a custom data format, assemble the following format codes into text strings. Format codes are case-sensitive. Text strings are always enclosed in double quotation marks.

Dates using these formatting strings will be displayed in the language of the user's locale.

Before building a date format with these strings, review the list of pre-defined Date Codes.

Format Code	Description
"d"	Day of month as digits with no leading zero for single-digit days.
"dd"	Day of month as digits with leading zero for single-digit days.
"ddd"	Day of week as a three-letter abbreviation.
"dddd"	Day of week as its full name.
"g"	B.C. or A.D.
"M"	Month as digits with no leading zero for single-digit months.
"MM"	Month as digits with leading zero for single-digit months.

"MMM"	Month as a three-letter abbreviation.
"MMMM"	Month as its full name.
"y"	Year as last two digits, but with no leading zero for years less than 10.
"yy"	Year as last two digits, but with leading zero for years less than 10.
"yyyy"	Year represented by full four digits.

Example:

```
Date(Today(), "dddd MMM dd, yyyy")
```

... yields, "Thursday Aug 28, 2008"

Related Information:

Time Formatting Codes – Use for displaying time.

Date – The function that uses the codes listed above.

Fill Patterns

The following numbers may be used in functions requiring a fill pattern value.

Index	Pattern	Example
1	Solid	
2	Even color mix	
3	Dominant background color mix	
4	NW-SE diagonals	
5	NE-SW diagonals	
6	Vertical lines	
7	Horizontal lines	

8	Vertical squiggles	
9	Horizontal squiggles	
10	Vertical jagged lines	
11	Horizontal jagged lines	
12	Diagonal jagged lines	
13	Vertical crosshatch	
14	Vertical and angled crosshatch	
15	Angled crosshatch	
16	Uneven crosshatch	
17	Hollow connected squares	
18	Checkerboard	
19	Vertical arrows	
20	Horizontal arrows	
21	Angled checkerboard	
22	Large dots	
23	Thick connected squares	
24	Squares with tiny dots	
25	Bricks	
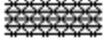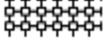
Font Character Sets

Returns the VTScada character set as an actual charset code. You may disregard these legacy presets and use the actual codes directly (see wingdi.h for list).

0	ANSI_CHARSET
1	DEFAULT_CHARSET (Current system charset. Typically, the same as ANSI_CHARSET)
2	SYMBOL_CHARSET
3	SHIFTJIS_CHARSET
4	OEM_CHARSET (System specific)
5	RUSSIAN_CHARSET
6	BALTIC_CHARSET
7	CHINESEBIG5_CHARSET
8	EASTEUROPE_CHARSET
9	GB2312_CHARSET
10	GREEK_CHARSET
11	HANGUL_CHARSET
12	MAC_CHARSET
13	TURKISH_CHARSET
14	VIETNAMESE_CHARSET

GUI Object Return Codes

Objects created using the GUIx commands will return a value when selected by the mouse or the enter button, according to the following table:

Return Value	Mouse Button(s)/Key	No. of Clicks
0	Invalid response	–
1	Right button	Single
2	Middle button	Single
3	Right and middle button	Single
4	Left button	Single

5	Left and right button	Single
6	Left and middle	Single
7	All three buttons	Single
8	Invalid response	–
9	Right button	Double
10	Middle button	Double
11	Right and middle button	Double
12	Left button	Double
13	Left and right button	Double
14	Left and middle	Double
15	All three buttons	Double
16	<ENTER> key	–

These are built from the following bit-wise values:

Bit	Meaning
0	TRUE: Right button clicked
1	TRUE: Middle button clicked
2	TRUE: Left button clicked
4	TRUE: Double-click, FALSE: Single-click
5	TRUE: Enter key presses (all other bits will be zero)

Known Path Aliases for File-Related Functions

Known Path Alias	Location
:{CommonProgramFiles}	Common program files (varies for 32-bit and 64-bit operating systems)
:{Fonts}	Fonts folder
:{ProgramFiles}	program files directory (varies for 32-bit and 64-bit

	operating systems)
:{System}	Windows System folder (varies for 32-bit and 64-bit operating systems)
:{SystemX86}	Windows System folder (32bit)
:{Windows}	Windows folder
:{ResourceDir}	Windows Resources folder

User Specific	
:{UserAdminTools}	Admin tools (start menu)
:{UserRoamingAppData}	Roaming app data
:{UserCDBurnArea}	Local pending CD to burn
:{UserCookies}	IE cookies
:{UserDesktop}	Desktop folder
:{UserFavorites}	IE favorites
:{UserHistory}	IE history
:{UserInternetCache}	Temporary internet files
:{UserLocalAppData}	Localized app data
:{UserDocuments}	Documents folder
:{UserMusic}	Music folder
:{UserPictures}	Photos folder
:{UserVideos}	Videos folder
:{UserNetHood}	Network Places shortcut folder
:{UserPrintHood}	Printer shortcut folder
:{UserProfile}	Users profile folder (root of users folders)
:{UserPrograms}	Programs (start menu)
:{UserRecent}	Shortcuts to recently viewed documents
:{UserSendTo}	Items in the 'Send To' context menu

:{UserStartMenu}	Start menu root
:{UserStartup}	Start menu startup folder
:{UserTemplates}	Document templates

All Users share the fol-lowing:	
:{CommonAdminTools}	admin tools (start menu)
:{CommonAppData}	app data storage
:{CommonFavorites}	IE favorites folder
:{CommonOEMLinks}	OEM Links
:{CommonPrograms}	start menu program list
:{CommonStartMenu}	start menu root
:{CommonStartup}	start menu startup folder
:{CommonTemplates}	templates folder
:{PublicDesktop}	desktop folder (shared icons, etc...)
:{PublicDocuments}	documents folder
:{PublicMusic}	music folder
:{PublicPictures}	photos folder
:{PublicVideos}	video folder

Line Types

The following numbers may be used in functions requiring a line style value.

Index	Style	Example
0	Invisible	
1	Solid	————

2	Dashed	— — —
3	Dotted	----------
4	Dot-dashed	- — - — - -
5	Dot-dot-dashed	- - — - - — - -

ParameterEdit Snap-ins

These are modules that do not stand alone, but can be used for parameter editing in the user interface.

ParmEditColor

Description | Used for choosing a color as your parameter value.

Parameters | **Color** – Parameter value
ParmPtr – Parameter pointer
Enable – Show the graphics for editing the parameter value
Left – Left coordinate
Bottom – Bottom coordinate
Right – Right coordinate
Top – Top coordinate
LabelWidth – Label width
LabelHeight – Label height
PtrWaitClose – TRUE to tell caller to wait to close

ParmEditExprMovement

Description | Used for choosing an expression as your parameter value for a movement parameter

Parameters | **Expr** – Parameter value
ParmPtr – Parameter pointer
Enable – Show the graphics for editing the parameter

value

Left – Left coordinate

Bottom – Bottom coordinate

Right – Right coordinate

Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label height

PtrWaitClose – TRUE to tell caller to wait to close

DlgRoot – Root of the edit dialog (UNUSED)

DropListLabel0 – Label 0 for the Movement direction drop-list

DropListLabel1 – Label 1 for the Movement direction drop-list

ParmEditExprNoNormalize

Description	Used for choosing an expression as your parameter. Note that, the result is not wrapped with a Normalize.
Parameters	**Expr** – Parameter value

ParmPtr – Parameter pointer

Enable – Show the graphics for editing the parameter value

Left – Left coordinate

Bottom – Bottom coordinate

Right – Right coordinate

Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label height

PtrWaitClose – TRUE to tell caller to wait to close

DlgRoot – Root of the edit dialog (UNUSED)

MenuEnables – Bits to enable options for expression editor

ParmEditExprNormalize

Description	ParameterEdit module for choosing an expression as your

parameter. The result is wrapped in a Normalize.

Parameters	**Expr** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label height
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	MenuEnables – Bits to enable options for expression editor

ParmEditFont

Description	Used by the ParameterEdit to choosing a Font as your parameter value
Parameters	**FontVal** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditHorizAlign

Description	ParameterEdit module for choosing horizontal alignment

as your parameter value.

Parameters	**HAlign** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditLineStyle

Description	ParameterEdit module for choosing a line style as your parameter value
Parameters	**Style** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditLineWidth

Description	ParameterEdit module for choosing a line width as your parameter value

Parameters	**Width** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditNum

Description	ParameterEdit module for choosing a number as your parameter value
Parameters	**NumVal** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	TypeOfValue – Optional, VTScada value type required
	MinVal – Optional, Min value allowed (Max also required)
	MaxVal – Optional, Max value allowed (Min also required)

ParmEditParmMovement

Description	ParameterEdit module for choosing a container's para–

meter as a value. Used for displaying the Movement para-
meter. Has all the additional information needed for the
Movement GUI call

Parameters	**ParmVal** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	ContainerInfo – Info struct for the container and its parameters
	DropListLabel0 – Droplist label 0
	DropListLabel1 – Droplist label 1

ParmEditParmValue

Description	ParameterEdit module for choosing a container's parameter as a value. NOTE. Due to the introduction of PickValids for default parameters, this selector must appear in the modules array before any of the expressions.
Parameters	**ParmVal** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label weight

PtrWaitClose – TRUE to tell caller to wait to close

DlgRoot – Root of the edit dialog (for sizing)

ContainerInfo – Info struct for container and its parameters

ValueTypeLo – Min value type allowed for this parameter. Note – If ValueTypeHi is invalid, then this is the only value type allowed.

ValueTypeHi – Max value type allowed for this parameter

SubTypeList – Optional list of subtypes for object type

Scaled – Optional Boolean to indicate whether the data be in a "Scale" expression. The default is FALSE

DefaultValue – Optional. The default value if the parameter's value is invalid

DefaultNoParm – Optional. The default value if no parameter is selected

ParmEditPattern

Description ParameterEdit module for choosing a fill pattern as your parameter value

Parameters **Brush** – Parameter value

ParmPtr – Parameter pointer

Enable – Show the graphics for editing the parameter value

Left – Left coordinate

Bottom – Bottom coordinate

Right – Right coordinate

Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label weight

PtrWaitClose – TRUE to tell caller to wait to close

DlgRoot – Root of the edit dialog (UNUSED)

ParmEditPipeColor

Description	ParameterEdit module for choosing a pipe color as your parameter value
Parameters	**HighColor** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditPipeWidth

Description	ParameterEdit module for choosing a pipe width as your parameter value
Parameters	**PipeWidth** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)

ParmEditTag

Description	Parameter Editing module for choosing a tag as your para–

meter value

Parameters	**TagParm** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the Parameter Value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label weight
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	PointType – Point Type allowed for this parameter

ParmEditTagMovement

Description	Parameter Editing module for choosing a tag value as your parameter value for a Movement parameter
Parameters	**TagMoveVal** – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label height
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	DropListLabel0 – Label 0 for the movement direction drop-list
	DropListLabel1 – Label 1 for the movement direction drop-list

PointType – Point Type allowed for this parameter

ParmEditTagProperty

Description

ParameterEdit module for choosing a property of the drawn tag.
NOTE. Due to the introduction of PickValids for default parameters, this selector must appear in the Modules array BEFORE any of the Expressions.

Parameters

ParmVal – Parameter value
ParmPtr – Parameter pointer
Enable – Show the graphics for editing the parameter value
Left – Left coordinate
Bottom – Bottom coordinate
Right – Right coordinate
Top – Top coordinate
LabelWidth – Label width
LabelHeight – Label height
PtrWaitClose – TRUE to tell caller to wait to close
DlgRoot – Root of the edit dialog (for sizing)
ContainerInfo – Info struct for container and its parameters
TargetTypeLo – Min value type allowed for this parameter. If TargetTypeHi is Invalid, then this is the only value type allowed.
TargetTypeHi – Max value type allowed for this parameter
TargetSubTypes – Optional list of subtypes for object type
Scaled – Should this value be scaled?
DefaultValue – Optional, default value if the tag's value is invalid
DefaultNoTag – Optional, default value if user selects 'No Property Selected'

ParmEditTPMovement

Description ParameterEdit module for choosing a container's para-
meter as a value

Parameters **ParmVal** – Parameter value

ParmPtr – Parameter pointer

Enable – Show the graphics for editing the parameter
value

Left – Left coordinate

Bottom – Bottom coordinate

Right – Right coordinate

Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label height

PtrWaitClose – TRUE to tell caller to wait to close

DlgRoot – Root of the edit dialog (for sizing)

DropListLabel0 – Label 0 for the movement direction
drop-list

DropListLabel1 – Label 1 for the movement direction
drop-list

ContainerInfo – Info struct for container and its para-
meters

ParmEditTagValue

Description ParameterEdit module for choosing a tag value as your
parameter value

Parameters **TagParm** – Parameter value

ParmPtr – Parameter pointer

Enable – Show the graphics for editing the parameter
value

Left – Left coordinate

Bottom – Bottom coordinate

Right – Right coordinate

Top – Top coordinate

LabelWidth – Label width

LabelHeight – Label height
PtrWaitClose – TRUE to tell caller to wait to close
DlgRoot – Root of the edit dialog (UNUSED)
PointType – Point type allowed for this Parameter
Scaled – Optional Boolean value indicating whether the data should be in a Scale expression. Default = true
DefaultValue – Optional, default value if the tag's value is invalid
DefaultNoTag – Optional, default value if no tag is set

ParmEditText

Description	ParameterEdit module for choosing a text message as your parameter value
Parameters	TextMsg – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label height
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	StringLiteral – TRUE to treat input/output as a string literal intended for placement in SRC code or similar. (Handles quotation marks.) Defaults to TRUE.

ParmEditTextExpression

Description	ParameterEdit module for choosing an expression that returns text as the parameter Value. Related to ParameterPanel, which calls the above.

Parameters	Expr – Parameter value
	ParmPtr – Parameter pointer
	Enable – Show the graphics for editing the parameter value
	Left – Left coordinate
	Bottom – Bottom coordinate
	Right – Right coordinate
	Top – Top coordinate
	LabelWidth – Label width
	LabelHeight – Label height
	PtrWaitClose – TRUE to tell caller to wait to close
	DlgRoot – Root of the edit dialog (UNUSED)
	MenuEnables – Bits used to enable options for the expression editor

ParameterPanel

Description	This is a Generic Panel which will handle the setting of Parameters for an object that does not have its own panel. The object can be a page or a widget.
	If available, hints are taken from the typing information of the object parameters and suitable Parameter Value choosers are offered.
	If the immediate container has parameters then, these too are considered as actual value candidates. However, if there is type information on the object or the container, then these hints are used to filter the set of options, possibly resulting in an empty set.
	This panel is designed to be callable from several sources, such as the VGE and the Display Manager.
Parameters	**ObjModule** – Object module that is to have its parameters modified
	pObjParams – Pointer to an array of the object's parameter values
	PtrWaitClose – TRUE when window closed or cancel

pressed

CodePtr – CodePtr so that the parameter value can be read

Left – Left position of the window

Bottom – Height of window

Right – Width of the window

Top – Top position of the window

HandleScrollBar – Flag – TRUE for this module to use a scrollbar if required

ContainerInfo – Information about container and its parameters

SelectedParms – Boolean array of selected parameters to make editable

DialogRoot – The calling dialog window

Related Functions:

ParameterEdit

SlippyMapRemoteTileSource1

Sets the URL, from which Site Map tiles are loaded. Defaults to: SlippyMapRemoteTileSource1 = http://c.tile.openstreetmap.org/ /|/ © [OpenStreetMap Contributors] (www.openstreetmap.org/copyright).

SQL Data Types

Type Indicator	SQL Data Type
-7	SQL_BIT
-6	SQL_TINYINT
-5	SQL_BIGINT
-4	SQL_LONGVARBINARY

−3	SQL_VARBINARY
−2	SQL_BINARY
−1	SQL_LONGVARCHAR
0	SQL_UNKNOWN_TYPE
1	SQL_CHAR
2	SQL_NUMERIC
3	SQL_DECIMAL
4	SQL_INTEGER
5	SQL_SMALLINT
6	SQL_FLOAT
7	SQL_REAL
8	SQL_DOUBLE
9	SQL_DATE
10	SQL_TIME
11	SQL_TIMESTAMP
12	SQL_VARCHAR

predefined Time Formats

Use any of the following numeric time codes to format a time as shown. If you require a custom format, you can build one using the text codes shown in Time Formatting Codes, as described in the VTScada Programmer's Guide.

Time Code	Example	Description
0		no time
1	103211	HourMinuteSecond

2	10:32:11	hour:minute:seconds
3	10321100	HourMinuteSecondHundredth
4	10:32:11:00	hour:minute:seconds:hundredths
5	10:32	hour:minute
6	10:32:11 AM	hour:minute:seconds AM or PM
7	10:32 AM	hour:minute AM or PM
8	103211000	hour minute second thousandths
9	10:32:11.000	hour:minute:second.thousandths

Related Information:

predefined Date Codes – Used to display a date value.

Time– Function that uses the codes listed above.

Time Formatting Codes

All examples display 9:07:12 p.m.

String	Example	Description
h	9	Hours with no leading zero for single-digit hours; 12-hour clock.
hh	09	Hours with leading zero for single-digit hours; 12-hour clock.
H	21	Hours with no leading zero for single-digit hours; 24-hour clock.
HH	21	Hours with leading zero for single-digit hours; 24-hour clock.

m	7	Minutes with no leading zero for single-digit minutes.
mm	07	Minutes with leading zero for single-digit minutes.
s	12	Seconds with no leading zero for single-digit seconds.
ss	12	Seconds with leading zero for single-digit seconds.
t	P	One character time-marker string, such as A or P.
tt	PM	Multi-character time-marker string, such as AM or PM.

Example:

```
Time(now(1), "hh.mm.ss tt")
```

... displays: 09.07.12 PM

Related Information:

Date Formatting Strings – Used to display a date value.

Time– Function that uses the codes listed above.

VTScada and Time Synchronization

Note: The following information is of concern only if you are not already maintaining time synchronization between servers and if it is important to your operation to maintain synchronization of clocks between servers.

VTScada includes code that will maintain time synchronization between networked servers. Due to security settings in Windows Vista and later versions, the Time Synchronization Service will work only if VTScada is started with an account that has the SE_SYSTEMTIME_NAME security flag set. The Windows Administrator account will generally have this privilege, but you can also set it for other user accounts.

To do so, you must set the user privilege, "Change the System Time" to true using the Windows Group Policy Management Editor. Please refer to Microsoft's documentation for instructions on using this system tool.

VTScada Value Types – Numeric Reference

The following table lists the value types used in VTScada. When referring to these in code, you should use the predefined constants rather than the type numbers. The general usage is:

Cast(Val, \#VtypeText)

Type	Constant Name	Name	Description
0	#VTypeStatus	Boolean	Logical data type, stores two states: "true" (0) or "false" (non-zero).
1	#VTypeShort	Short, 16-bit signed	Integer data type storing values from -32768 to 32767
2	#VTypeLong	Long, 32-bit signed	Integer data type storing values from -2147483648 to 2147483647
3	#VTypeDouble	Double pre-cision floating point	Values range from about -10^308 through +10^308
4	#VTypeText	Text	Any string of bytes whose values range from 0 to 255. Typically used to hold text strings.
5	#VTypeVariable	Variable	A handle to the data represented by a variable declaration, not to any particular instantiation of that declaration. Can be used to access variable metadata (type information, for example) or default values.
6	#VTypeFunction	Function	A pointer to the code for a par-ticular function within a VTScada statement. Used by functions such as GetOneParmText to

			manipulate the code itself. Used when compiling and edit- ing script code, not for typical VTScada programs.
7	#VTypeObject	Object value	An instance of a module
8	#VTypeStream	Stream	A handle to a stream (of which there are several types). See Streams.
9	#VTypeModTree	Module tree	A handle to the modules in a state diagram
10	#VTypeStateDgrm	State diagram	A graphical depiction of VTScada code
11	#VTypeModule	Module	The code and variables that make up a unit of a VTScada pro- gram. See Modules.
12	#VTypeModState	Code Value (a) Module and state	A handle to a state within a mod- ule. See States.
13	#VTypeModStateStmnt	Code Value (b) Module, state, and statement	A handle to a statement within a state. Cannot refer to any arbit- rary function, as type 6 can. See Statements and Graphic Objects.
14	#VTypeRefParm	Reference parameter	When a steady-state call is made to a module, each of the actual parameters in the call is "bound" to its corresponding formal para- meter.
15	<undefined>	Array	Refers to an entire list of con- secutive data values. Each data value has a consecutively numbered index address and may be any VTScada value. See Array Variables

16	#VTypePath	Path	A series of vertex values. See Path Variables.
17	#VTypeTraj	Trajectory	A combination of a Normalize value and a Path value. See Trajectory Variables.
18	#VTypeRotate	Rotate	Specifies a rotation amount, measured in degrees, around a point. See Rotate Variables
19	#VTypeBrush	Brush	Brush values are used in layered graphics statements that paint areas of the screen with a uniform color or pattern. See the Brush function.
20	#VTypePen	Pen	Pen values are used in layered graphics statements that draw lines. Defines the color, style and thickness of a line. See the Pen function.
21	#VTypeNormalize	Normalize	A graphical scaling value. See Normalize.
22	#VTypePoint	Point	A location, stored as an (X, Y) pair. See Point.
23	#VTypeVertex	Vertex	A group of three Point values. See Vertex.
24	#VTypeTransform	Transform	A transformation matrix, used to map coordinates from one area of the screen to another. Can only be obtained from the GetTransform function. Used by the GetPathBound function.
25	#VTypeCodePtr	Code pointer	A handle to an active graphics statement in a particular module

			or state. Similar to type 13, but with the additional information of the module instance as represented by value type 7.
26	#VTypePtr	Pointer	Stores data by reference instead of by value, allowing, for example, multiple values to reference the same piece of data as opposed to multiple copies of the data.
27	#VTypeEditor	Editor	A handle to an editor object, as created by MakeEditor.
28	#VTypeParseStack	Parser stack	Used by the compiler to allow the compilation to be suspended in the middle of a statement to handle specific code sections such as I/O addresses.
29	#VTypeTag	Tag	(Unused) Intended to provide engine-level support for scaled variables that could be implemented using a GUI.
30	#VTypeBitmap	Bitmap	A handle to an image object as returned from MakeBitmap.
31	#VTypeFont	Font	A handle to a font object, as returned by the Font function.
32	#VTypeVTSdb	VTScada database	A handle to the VTScada database as returned by the DBSystem function.
33	#VTypeODBCHndl	ODBC Handle	Provides a connection to an ODBC database.
34	#VTypeSAPIStrm	SAPI text-to-speech stream	A type of stream for use with Speech Application Programming Interfaces

35	#VTypeComClient	COM Client Interface	An object that provides an interface to a COM client application
36	#VTypeCryptoProv	Cryptographic Provider	A handle to the particular cryptographic service provider that includes the key specification to use.
37	#VTypeCryptoKey	Cryptographic Key	May be either a Session Keys or a Public/Private Key. See Cryptographic Keys.
38	#VTypeDLLhandle	DLL Handle	A pointer to a structure returned from the LoadDLL function. Used to call functions within the DLL that was loaded. See DLL.
39	#VTypeDeflateHandle	ZLib Compression Handle	Used by the Deflate function
40	#VTypeThread	Thread Handle	A script-level hook to the data structure used to represent a thread in a dump
41	#VTypeBreakWatch	Source Debugger Breakpoint Handle	References a set location in the source debugger. See Working with Breakpoints and Data Breakpoints
42	#VTypeMiniDumpHandle	Minidump Data Handle	A pointer to a data structure that holds information from a crash dump
43	#VTypeTimeStamp	Timestamp	A numeric representation of time, measured in seconds since January 1, 1970
44	#VTypeXMLproc	XML Processor Handle	Serves as a conduit between an XML document and an application. See VTScada Engine XML API

45	#VTypeTypeDefinition	Dynamic Module Definition	Deprecated. A handle to the definition of a form of module used as a data container. Created by the MakeType function. This storage is used almost exclusively for handling XML and cannot contain script code (unlike other forms of Module).
46	#VTypeTypeInstance	Dynamic Module Instance	Deprecated. An instance of a dynamic module, created using the MakeTypeInstance function. This is an object value (type 7) that can only be used to store data – it cannot contain or execute script. Typically these are used when generating module trees for delivery via XML. It is a form of data container, however in general structures (defined by the Struct function) and Dictionaries (type 47) are more efficient and convenient for this role.
47	#VTypeDictionary	Dictionary	A key-based data container of flexible size, used either on its own to hold volatile data collections or in the definition of structures (see Structures). ValueType will not return this value unless the dictionary is a "pure" dictionary. A pure dictionary is one for which the root value has not been set. Otherwise, it returns the ValueType of

the dictionary's root instead. See
Dictionaries

48	#VTypeComProperty	COM Property	A value exposed by a COM Interface "object". This may be accessed similarly to a typical VTScada value but is maintained by the COM object, not the VTScada engine.
49	<undefined>	Module in Context	Contains both a module value and an instance of the context module where scope should be resolved. Normally, scope will be the parent module in which the Module was declared. A Module in Context is used for widgets and plug-ins in VTScada where the widget is declared in AppRoot.SRC, but linked into a tag type such that the widget becomes a Module in Context in the tag instance. References to variables in the widget will then refer to variables in the tag rather than to variables in AppRoot where the widget was declared. If a Module In Context value is called in steady-state, the parent instance will provide the associated context.
50	#VTypeHistorianHandle	Historian Connection Handle	For the VTScada proprietary data store, this will be invalidated on an "out of disk space" error, or

			on loss of access to the file storage. For other databases, this will be invalidated on any connection loss.
51	#VTypeXMLNode	Dictionary Structure	A WEB_XML_ADDRESS that points to a WEB_XML_NODE. When ValueType() runs against a value and finds a WEB_XML_ ADDRESS it treats it the same as a WEB_VALUE_ADDRESS, which sits in front of an array or structure. It then searches through the *_ADDRESS to find what it points to and returns the type of that item, in this case an USER_ XML_NODE
52	#VTypePPPHandle	PPP Connection Handle.	May be passed into the function, PPPStatus() to obtain an information structure. May be passed to the function, CloseStream() to forcibly close off a connection. Passing it into CloseStream completely invalidates the handle and all data associated with it.(see: PPPStatus and CloseStream)

Value and Type Conversions

The following table shows VTScada value and type conversions.

Value to Convert	Convert To	Condition of Original Value	Returned Value

Code Pointer	Module	Valid value	Valid value
	Module State	Valid value	Valid value
	Module State		
	Statement	Valid value	Valid value
	Object	Valid value	Valid value
	Text	Valid value	Name of module
Double	Long	Valid value	Valid value
	Short	Valid value	Valid value
	Status	Valid value	Valid value
	Text	Valid value	String representing numeric value
Edit Block	Stream	May only be used in SRead with % option, or in StrLen	Valid value
Long	Double	Valid value	Valid value
	Short	Valid value	Valid value
	Status	Valid value	Valid value
	Text	Valid value	String representing numeric value
Module	Text	Valid value	Name of module
Module State	Module	Valid value	Valid value
	Text	Valid value	Name of module
Module State Statement	Module	Valid value	Valid value
	Module State	Valid value	Valid value
	Text	Valid value	Name of module
Normalize	Double	Valid value	The scaled value
	Long	In the range of −2 147 483 648 to 2 147 483 647	The value

	Short	In the range of −32 767 to 32 767	The value
	Outside of range	Invalid	
	Status	0	0 (false)
		Non−0	1 (true)
	Text	Valid value	String representing scaled value
Object	Module State	Valid value	Module and state in which that object exists
	Statement	Valid value	Module state and statement number that object is executing
	Text	Valid value	Name of the module of which that object is an instance
Short	Double	Valid value	Valid value
	Long	Valid value	Valid value
	Status	Valid value	Valid value
	Text	Valid value	String representing numeric value
Status	Double	Unconnected socket	1
		Connected socket	Invalid
	Long	Unconnected socket	1
		Connected socket	Invalid
	Short	Unconnected socket	1
		Connected socket	Invalid
	Status	Unconnected socket	1
		Connected socket	Invalid
	Text	Unconnected socket	"1"
		Connected socket	String of stream contents
Tag	Double	Valid value	The scaled value
	Long	In the range of −2 147 483 648 to 2 147 483 647	The value

		Outside the range	Invalid
	Short	In the range of -32 767 to 32 767	The value
		Outside the range	Invalid
	Status	0	0 (false)
		Non-0	1 (true)
	Text	Valid value	String representing scaled value
Text	Double	Number string	Number in string
	Long	Number string	Number in string
	Short	Number string	Number in string
	Status	Non-0 number string	0 (false)
		0 number string	1 (true)
Variable	Module	Module variable	The module in which the module variable resides
	Text	Valid value	Name of variable

Uninstall VTScada

Your VTScada installation includes a wizard that can assist you in uninstalling the entire VTScada suite, or selected VTScada components. Uninstalling VTScada is a simple, two-step process that removes all components of the VTScada software from your workstation. The uninstall process does not, however, affect any VTScada applications you've created. These applications and their resources will remain untouched. Follow these steps to completely uninstall VTScada from your system.

1. Ensure that VTScada is not running.
2. Navigate to the VTScada product or installation directory.
3. Locate the Uninstall.exe file and run it.
4. Ensure that the Automatic radio button is selected.
5. Click the Next button. The Perform Uninstall dialog opens.

6. Click the Finish button. The Uninstall Wizard removes all VTScada components from your workstation.

> Note: Although the Uninstall Wizard removes all VTScada components from your workstation, it leaves any applications you have created and their data untouched. These applications can be imported into other versions of VTScada at a later date.

Language Support

The VTScada has been created using only English. Some developers have created application that present a portion of the user interface in languages other than English. To do so:

To use another language for your application's user interface:

1. Ensure that you are using fonts that support the full range of characters (including accented characters) used by the desired language. Modify the built-in the Font tags to use those fonts.
2. Ensure that the default Windows system font uses the correct selection for your language.
3. Edit your pages and pop-up pages to use the appropriate words for your language. Page titles can be edited as required using page properties.

To use another language for the VTScada configuration dialogs and messages:

1. Open the file, C:\VTScada\Setup.INI and replace the label properties with the appropriate words for your language.
   Note: do not edit the property names. Replace only the values attached to those names.
2. Re-start VTScada in order to load the new labels. Setup.INI is read only when VTScada restarts.
3. Using the advanced mode of the Edit Properties page of your application's Application Configuration dialog, replace label values with the appropriate

words for your language.

You will need to copy many of the properties from the OEM layer.

Related Information:

...Using a Non-English Character Set – Configure Windows (and thereby VTScada) to use alternative character sets.

Using a Non-English Character Set

To display non-English characters in VTScada, you may to do the following: (Steps describe how to use a Chinese character set.)

1. Update your Windows to include Chinese character set if you haven't done that when you install your Windows.
2. Go to Control Panel -> Clock, Language, and Region -> Region and Language Setting.
3. Select 'Administrative' tab and click on the "Change system Locale..." button.
4. If the client is from Main land China, select "Chinese (Simplified, PRC)" from the droplist.

If the client is from Hong Kong, select Chinese (Traditional, Hong Kong S.A.R) and so on. The screen should look like the following after your selection.
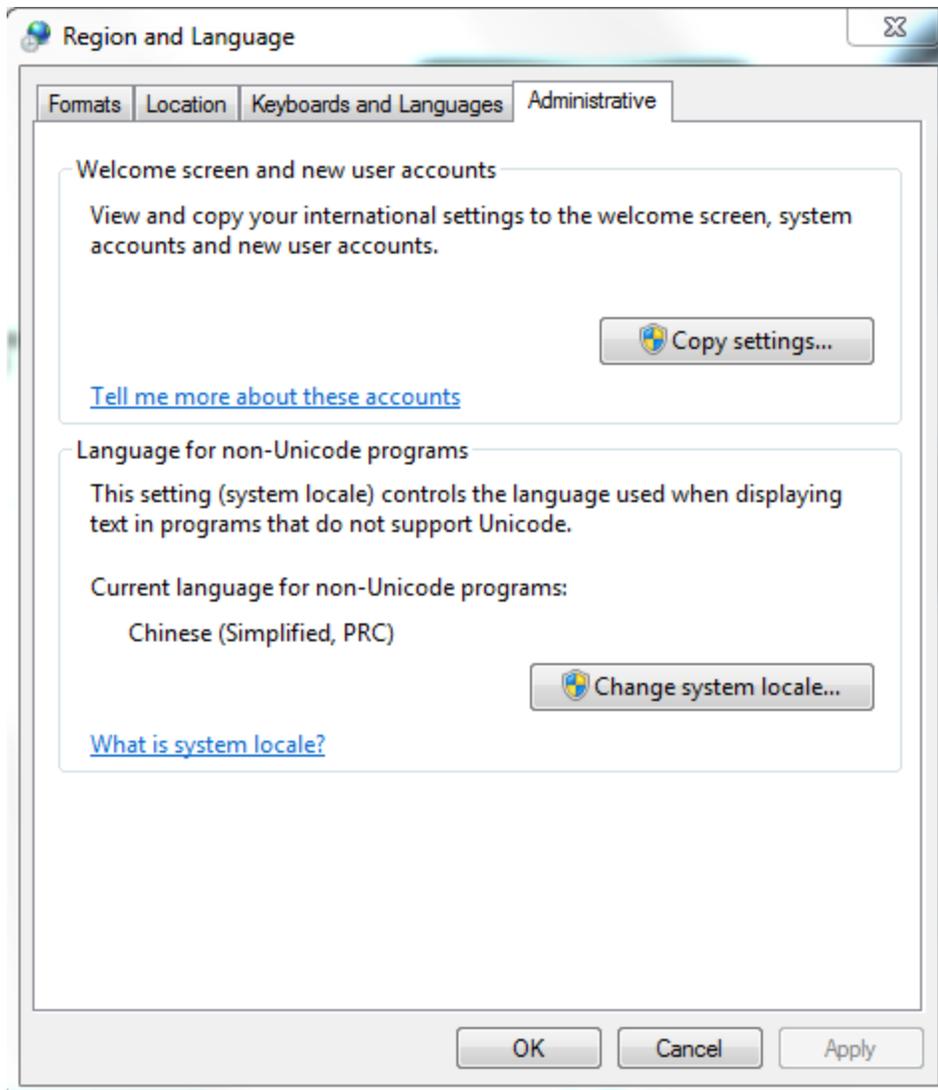
Windows will ask you to restart your system.

5. If you want to input Chinese characters in VTScada, you need to go to Control Panel -> Region and Language setting, select 'Keyboards and Languages" tab and click on "Change keyboards..." button.

6. In the "Text Services and Input Languages" dialog, Add Chinese input in the "Installed services" and keep English as Default Input Language.

7. Run VTScada and you can edit text messages, button labels or Page Notes in Chinese.

Note: tag names do not support Chinese characters.

切换页面

5.0

0 10 20 30 40 50 60 70 80 90 100

早上好
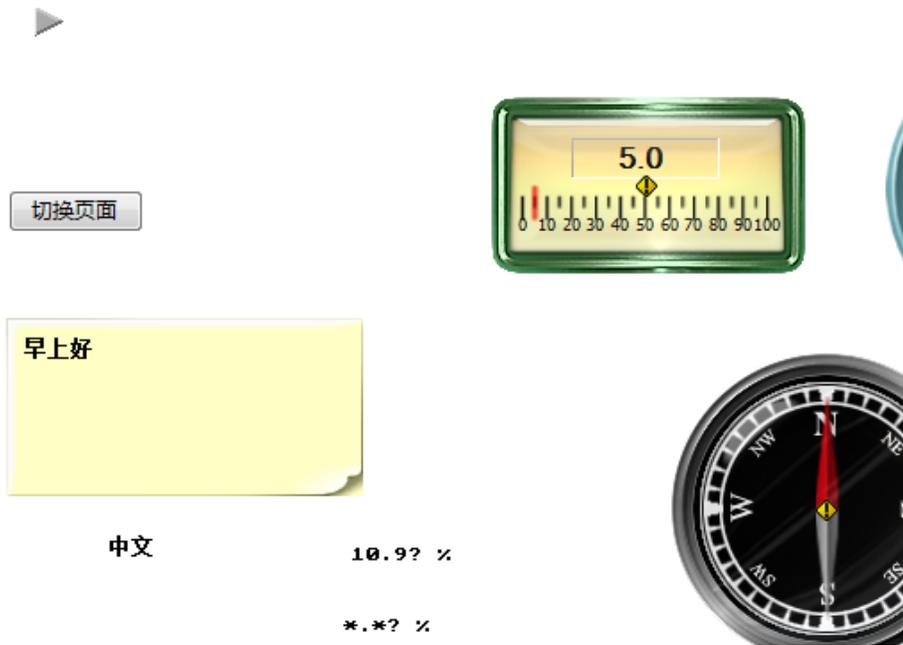
中文

10.9? %

.? %

Related Information:

...Language Support – Creating a user-interface with a language other than English.

Index