
Industrial Automation Crimson® 3.0 HMI/DSP/MC/PTV Widgets: Tech Note 01 Folder Binding and Details Creation



Abstract:

This document describes advanced widget functionality. Refer to the Crimson 3 User Manual for basic widget definition and detailed usage instructions.

Products:

Red Lion Controls CR1000 Human Machine Interface (HMI), CR3000 HMI, Data Station Plus (DSP) (excludes LE), G3 HMI, G3 Kadet HMI, Graphite® Controller, Graphite HMI, Modular Controller (MC) (excludes LE), and ProductVity Station™ (PTV)

Use Case:

CR1000 HMI, CR3000 HMI, DSP (excludes LE), G3 HMI, G3 Kadet HMI, Graphite Controller, Graphite HMI, MC (excludes LE), and PTV

Required Software:

Crimson 3.0 or Crimson 3.1

Required Operating System:

Microsoft Windows 2000, or above

Introduction

Beginning with Version 3.0, Crimson provides the capability to make ordinary groups of primitives into widgets. The primary steps required to create a widget are:

- Tag Configuration - A folder is created and the desired tags are mapped to the folder
- Display Configuration - A display page is created using the previously created tags
- Widgetization and Data Definition - The primitives to be widgetized are created and defined
- Widget Primitive Modification - The desired properties are assigned to each widget
- Widget Folder Binding - Widget data items are populated with their associated tags and then bound to the tags

NOTE: Widget support applies to only Crimson 3.0 or above and is only available to Windows 2000 or above users. Please update your version of Crimson to the latest; available online at www.redlion.net

Tag Configuration

Configure a tag by referring to Figure 1 and performing the following:

1. Create a new folder by clicking the *New Folder* icon at the top of the Navigation Pane.
2. Create and map your tags, or move existing tags, into this folder.

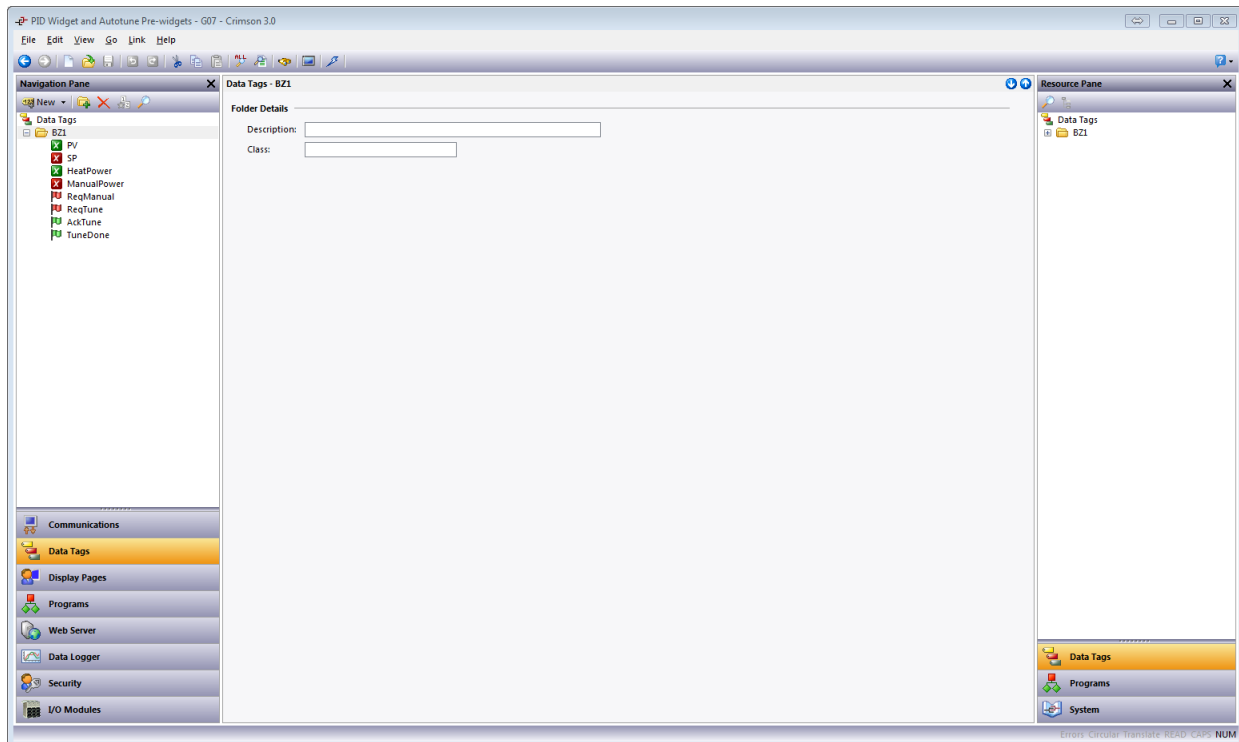
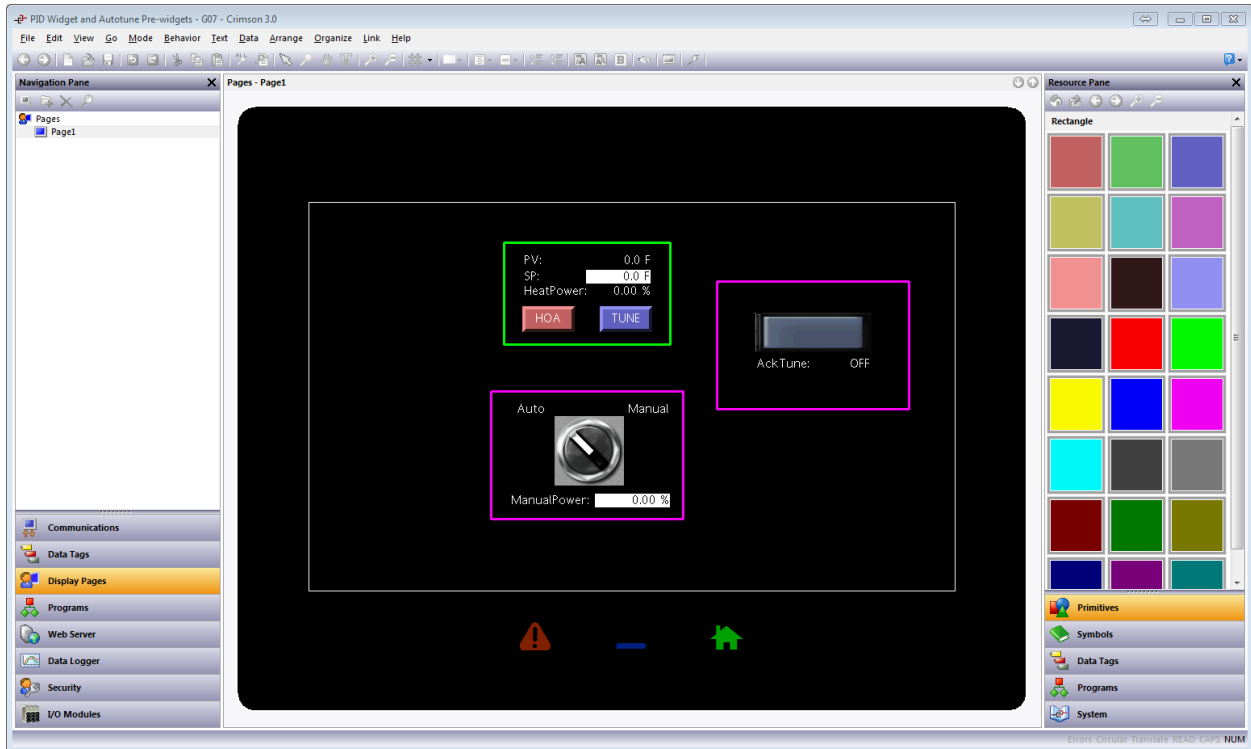


Figure 1.

Display Configuration

Display pages are typically configured using items dragged from the Resource Pane. Configure a display by performing the following:

1. Refer to Figure 2 and configure the display page using the desired tags.



Note: The green rectangle shown in the screen above is going to be the 'main' widget, while the fuchsia rectangles will be their own widgets that will be used for Details creation in later steps.

Figure 2.

Widgetization and Data Definition

Widgets are created via the Widget Data Definition screen, shown in Figure 3.

1. Refer to Figure 3 and select the primitives to be widgetized.
2. Right-click in the selection.
3. Click *Widgetize*.
4. On the Data tab, click *Edit* below Definition.
5. Click *Edit* next to Count.
6. Specify the number of data items within the widget.
7. Click *OK*.
8. Fill out the Widget Data Definition items.
 - a. It is recommended (for reasons specified in future steps) that the related tag names be used in the Name column.
 - b. The Description column is what will be displayed during manual widget binding.
 - c. The Data Type should match the tag type. For Flag tags choose Integer for the Data Type.
 - d. Flags should be set based on requirements. Choosing Tag will allow all tag properties (format, limits, colors, etc....) to be used within the widget.
9. Under Binding, specify *Enabled* for the Folder Binding value.
10. (optional) Under Details, specify *Enabled* for the Details Creation value.
11. (optional) Under Details, for the Details Widget value: type the names of the previously created widgets (.wid files), separated by commas, that will be created as details pages.

Note: When the 'main' widget is bound, new pages (named Details1, Details2, etc....) will be created; each page will have a widget added to the page and bound to tags within the folder.

12. Click *OK* twice.

Widget Data Definition

Extent

Count:

Definitions

	Name	Description	Bind To	Data Type	Flags	
1:	<input type="text" value="PV"/>	<input type="text" value="Process Value"/>	<input type="text" value="Per Name"/>	<input type="text" value="Integer"/>	<input type="text" value="Tag"/>	<input type="button" value="Edit..."/>
2:	<input type="text" value="SP"/>	<input type="text" value="Set Point"/>	<input type="text" value="Per Name"/>	<input type="text" value="Integer"/>	<input type="text" value="Tag, Writable"/>	<input type="button" value="Edit..."/>
3:	<input type="text" value="HeatPower"/>	<input type="text" value="Heat Power"/>	<input type="text" value="Per Name"/>	<input type="text" value="Integer"/>	<input type="text" value="Tag"/>	<input type="button" value="Edit..."/>
4:	<input type="text" value="Details1"/>	<input type="text" value="HOA Page"/>	<input type="text" value="Per Name"/>	<input type="text" value="Page"/>	<input type="text" value="None"/>	<input type="button" value="Edit..."/>
5:	<input type="text" value="Details2"/>	<input type="text" value="Tune Page"/>	<input type="text" value="Per Name"/>	<input type="text" value="Page"/>	<input type="text" value="None"/>	<input type="button" value="Edit..."/>

Binding

Folder Binding:

Required Folder Class:

Binding Prefix:

Details

Details Creation:

Details Widget:

Figure 3.

Widget Primitive Modification

1. Click the widget to select it, as shown in Figure 4. The area selected will be highlighted by a red box.

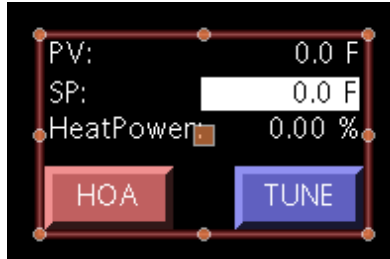


Figure 4.

2. Click an item within the widget to select an individual primitive, as shown in Figure 5; PV in this example. The area selected will be highlighted by a red box.

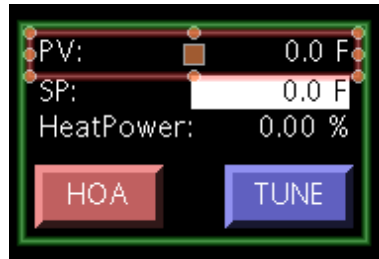


Figure 5.

3. Right-Click the selected primitive.
4. Click *Properties* from the drop-down menu (not shown); the corresponding Properties window will be displayed, as shown in Figure 6.

5. On the Data Tab, next to Value, click *Tag*.

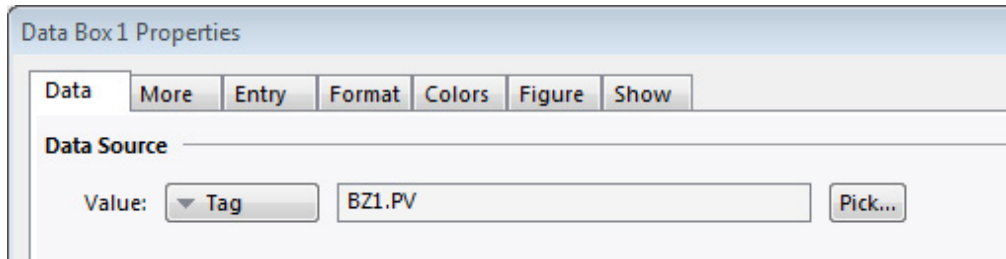


Figure 6.

5. Choose *General*.
6. Delete the folder name and "." separator as shown in Figure 7 and press *Enter*. Repeat this process for all of the properties associated with the tag.

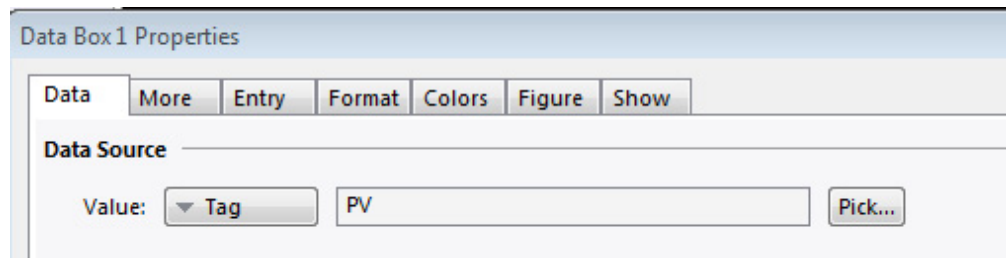


Figure 7.

7. Click *OK*.
8. Repeat steps 2 thru 8 for all primitives within the widget.

Note: The items defined in the widget properties are not available via the drop-down options or in the Resource pane, so their names must be typed in the general fields. For example, the details pages are not available for the Goto Page actions of buttons, so the GotoPage or ShowPopUp functions are required. The same holds true for tag colors; PV.Fore would have to be typed in the general field because it is not available via the drop-down.

9. With the widget selected, as in Step 1, click *Organize-Save Widget* or enter *CTRL+Q*.
10. Save the file in the default location: *C:\ProgramData\Red Lion Controls\Crimson\3.0\Widgets*

Widget Folder Binding

Crimson's ability to organize tags in folders provides a kind of object-oriented design. Tags that represent the properties of an object can be grouped, or bound, together in a folder that represents the object itself. To bind a widget to a folder, perform the following:

1. Add the widget to a page, as shown in Figure 8.
2. Right-click the widget.
3. Choose *Bind Widget*
4. Drag the tag folder from the Resource Pane into the Select Folder window; note that:
 - a. All of the widget's data items will be populated with the associated tags.
 - b. If Details Creation was enabled, new pages would have been created, had widgets been added to the page and those widgets bound to their associated tags.

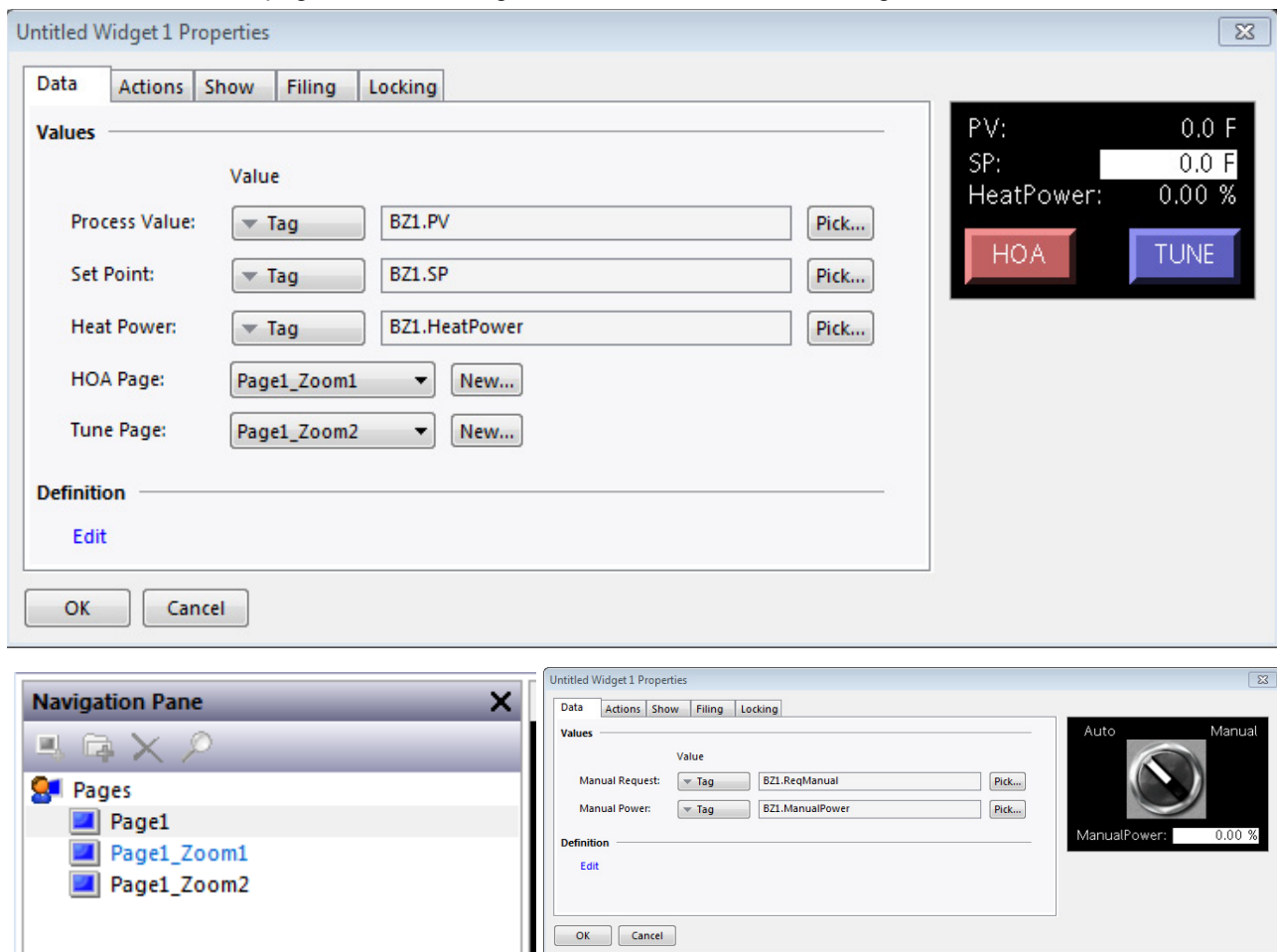


Figure 8.

Disclaimer

It is the customer's responsibility to review the advice provided herein and its applicability to the system. Red Lion makes no representation about specific knowledge of the customer's system or the specific performance of the system. Red Lion is not responsible for any damage to equipment or connected systems. The use of this document is at your own risk. Red Lion standard product warranty applies.

Red Lion Technical Support

If you have any questions or trouble contact Red Lion Technical Support by emailing support@redlion.net or calling 1-877-432-9908.

For more information: <http://www.redlion.net/support/policies-statements/warranty-statement>

