Allen-Bradley

IntelliCENTER ActiveX

Topic	Page
Introduction	1
Getting Started	4
FactoryTalk View SE Application Integration	5
Advanced IntelliCENTER ActiveX Configuration	13
Advanced IntelliCENTER ActiveX Integration - VBA	21
FactoryTalk View ME IntelliCENTER ActiveX Integration	23
Object Model - IntelliCENTER Elevation View ActiveX Control	24
Object Model - IntelliCENTER Monitor ActiveX Control	31
Object Model - IntelliCENTER Document ActiveX Control	35
Object Model - IntelliCENTER Spreadsheet ActiveX Control	39
Additional Resources	41

Introduction

IntelliCENTER® software offers the integration of preconfigured faceplates into an HMI environment using ActiveX controls. This document provides instructions on how to integrate the IntelliCENTER ActiveX objects into a FactoryTalk® View SE application. For information on FactoryTalk View SE, see FactoryTalk View SE User Guide ViewSE-UM006.

What Is ActiveX?

ActiveX is a software framework that allows the use of preconfigured objects to be deployed into an ActiveX container, such as FactoryTalk® View. The preconfigured objects allow for easy and repeatable deployment into a design environment.

Benefits of Using IntelliCENTER ActiveX Controls

IntelliCENTER ActiveX control integration into an HMI environment provides the main benefits of IntelliCENTER without the need to install IntelliCENTER software on each HMI.

- Elevation view provides a graphical representation of your entire MCC lineup
- Monitor view displays an overview of the monitored intelligent device and lets you change configuration parameters
- Documentation views are customized for your IntelliCENTER MCC, it includes manuals, CAD drawings, and a spare parts list.
- Event log records and stores events that occur within the IntelliCENTER MCC

For a full list of the benefits of IntelliCENTER software, see Features and Benefits of IntelliCENTER Software in the IntelliCENTER Software User Manual, publication MCC-UM002.

This HMI integration is possible due to the IntelliCENTER ActiveX objects that use the same configuration database as the standalone IntelliCENTER software. All connections to the end devices are done through RSLinx® to allow the ActiveX objects to gather specific device information. The ActiveX objects do not require any entries into the FactoryTalk View tag database.

What ActiveX Controls are Provided?

IntelliCENTER software provides four main ActiveX controls that are configured to any IntelliCENTER MCC lineup or intelligent device available in your IntelliCENTER database.

- Elevation ActiveX Control
  - This control provides a means to show the elevation view to monitor the status of a lineup.
- Monitor ActiveX Control
  - This control provides the features of the standard Monitor view in the IntelliCENTER software. You can use it to view, trend, and change parameters.
- Documents ActiveX Control
  - This control provides a means to view drawings, manuals, spare parts, and events for a specific lineup or device.
- Spreadsheet ActiveX Control
  - This control provides a way to view data for one device or an entire lineup.

IMPORTANT ActiveX Controls are used to monitor and edit by using the tools. However, there are some functions that you can only change with a complete installation of the IntelliCENTER software. For more information, see IntelliCENTER Software User Manual, publication MCC-UM002.

System Requirements

IntelliCENTER ActiveX functionality requires the same system requirements as the standalone IntelliCENTER software. For a list of the specific requirements, see System and Equipment Requirements in the IntelliCENTER Software User Manual, publication MCC-UM002.

ActiveX Requirements

IntelliCENTER ActiveX controls can be used on an HMI that does not have the complete installation of the IntelliCENTER software. To verify that all prerequisites are met, see the list of required software:

- RSLinx® Classic that is installed locally - needed for the ActiveX control to make a connection to the end devices
- Client Access to IntelliCENTER data files - these files can be installed locally on the client or accessible on a shared network drive
- IntelliCENTER ActiveX controls installed on each client

IMPORTANT The client that runs the runtime HMI display requires Windows operating system at 32 bit or above in order for the ActiveX control to work properly.

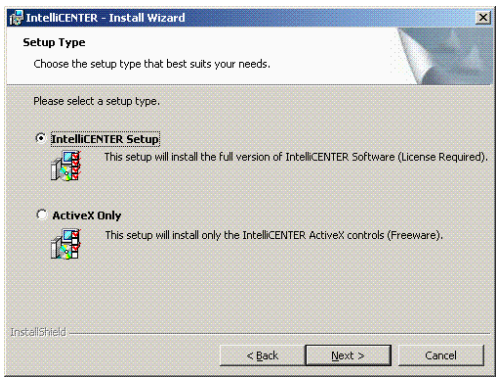
Getting Started

This manual provides guidance on basic and advanced ActiveX integration techniques into a FactoryTalk View SE application.

Software Install

The IntelliCENTER ActiveX controls are contained on the IntelliCENTER software program disk. The ActiveX controls are installed the same way as the full version of IntelliCENTER software except for the selection that is made in the dialog box in Figure 1.

Figure 1 - IntelliCENTER ActiveX Install



Instead of selecting 'IntelliCENTER Setup', select 'ActiveX Only' to install the ActiveX controls.

For the complete steps on how to install IntelliCENTER software, see New IntelliCENTER Software Installation in the IntelliCENTER Software User Manual, publication MCC-UM002.

RSLinx Configuration

IntelliCENTER ActiveX functionality requires RSLinx® Classic to be installed and configured on each client HMI. Configure the RSLinx Classic Driver and point to the driver that includes all devices within the IntelliCENTER MCC lineup. If the configuration has not been completed on the client HMI, see Set-Up RSLINX Classic Driver in the IntelliCENTER Software User Manual, publication MCC-UM002.

Installing IntelliCENTER Data Files

The IntelliCENTER data files for each IntelliCENTER MCC lineup, which is going to be accessed from the HMI, must be accessible from the HMI client. The data files are made accessible by installing the IntelliCENTER data files locally, or by installing them on a shared network drive that the client has access to.

To install the IntelliCENTER data files, see Installing IntelliCENTER Software Data Disks in the IntelliCENTER Software User Manual, publication MCC-UM002.

FactoryTalk View SE Application Integration

Several methods can be used when using the IntelliCENTER ActiveX controls in a FactoryTalk View SE application. The IntelliCENTER ActiveX control can be added to an existing display, and the properties can be modified locally or by using HMI tags. Another method allows the IntelliCENTER ActiveX control to be integrated into the HMI through Visual Basic for Applications, or VBA.

Adding IntelliCENTER ActiveX Controls to Display

The most basic method to integrate the IntelliCENTER ActiveX controls within FactoryTalk View SE is to add the object directly onto the display. See the FactoryTalk View SE User Guide ViewSE-UM006 for guidance on how to add ActiveX objects to a display.

To learn more about ActiveX within the FactoryTalk View SE environment, see the following sections in the FactoryTalk View SE User Guide ViewSE-UM006:

- Working with ActiveX objects
- Deploying the ActiveX components automatically at runtime

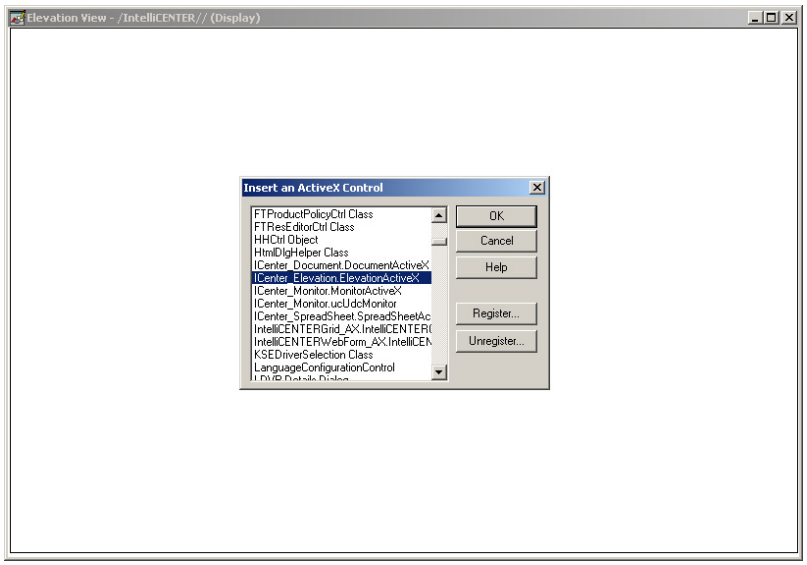
Modifying IntelliCENTER ActiveX Controls

The IntelliCENTER ActiveX controls that are added to the ActiveX library include the elevation view, monitor view, documents view, and spreadsheet view. Each of these ActiveX controls can be configured to a specific lineup or a specific intelligent device. Configuration is done through the IntelliCENTER ActiveX control properties.

Elevation View

The Elevation view ActiveX control is added to an HMI screen by using the ICenter_Elevation.ElevationActiveX control, as seen in Figure 2.

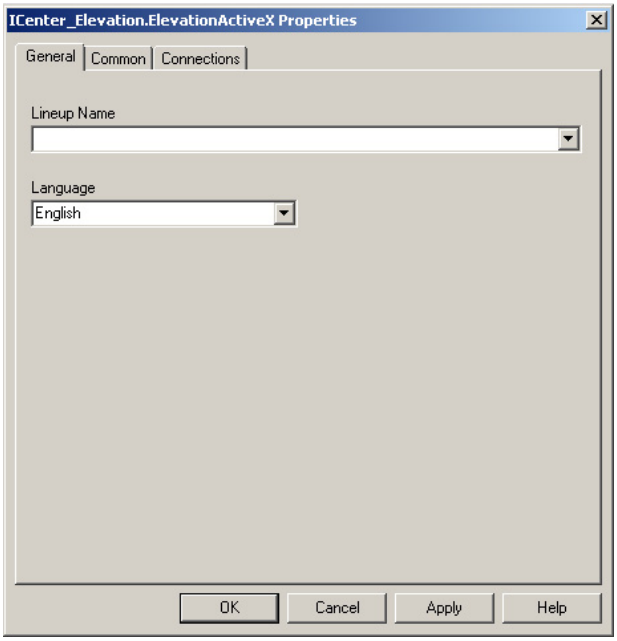
Figure 2 - IntelliCENTER ActiveX Elevation View Control



Once the Elevation view ActiveX control is added to the display, the properties are configured based on the IntelliCENTER MCC lineup that this ActiveX control displays, see Figure 3.

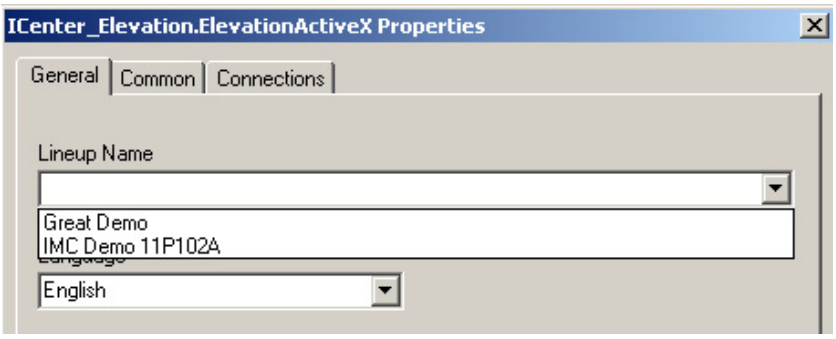
IMPORTANT Size the Elevation view ActiveX control large enough so that the information can be easily viewed. If made too small, the information that is provided in the ActiveX control is unreadable.

Figure 3 - IntelliCENTER Elevation View ActiveX Properties



The desired lineup is selected from the pull-down menu as seen in Figure 4. This list is auto created based on the IntelliCENTER data files that have been installed. The language of the Elevation view can also be changed by using the pull-down menu.

Figure 4 - IntelliCENTER Elevation View ActiveX Control Property Lineup Configuration

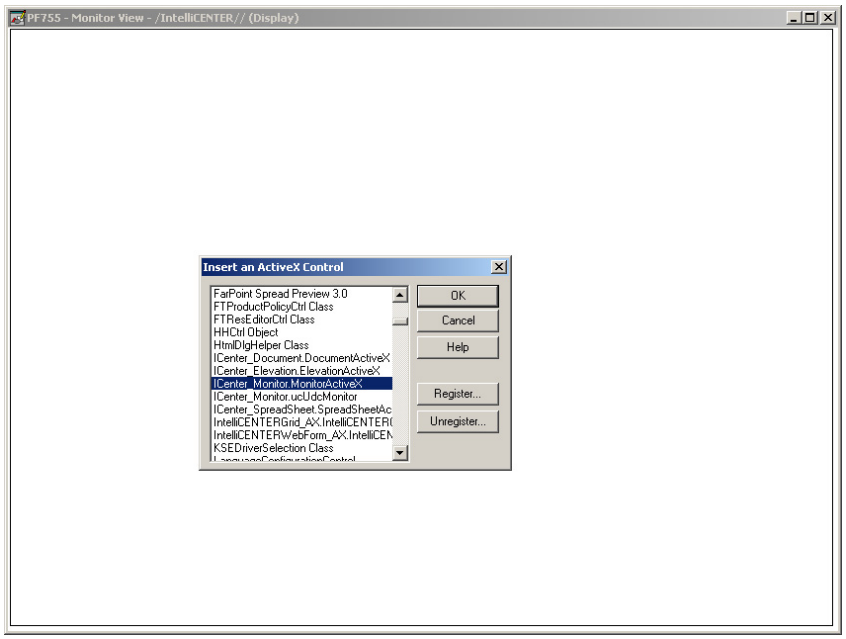


No other configuration is required for the Elevation view ActiveX control to access the information for the selected IntelliCENTER MCC lineup. The information about the lineup is accessed from the IntelliCENTER database through the communication path that is configured in RSLinx Classic.

Monitor View

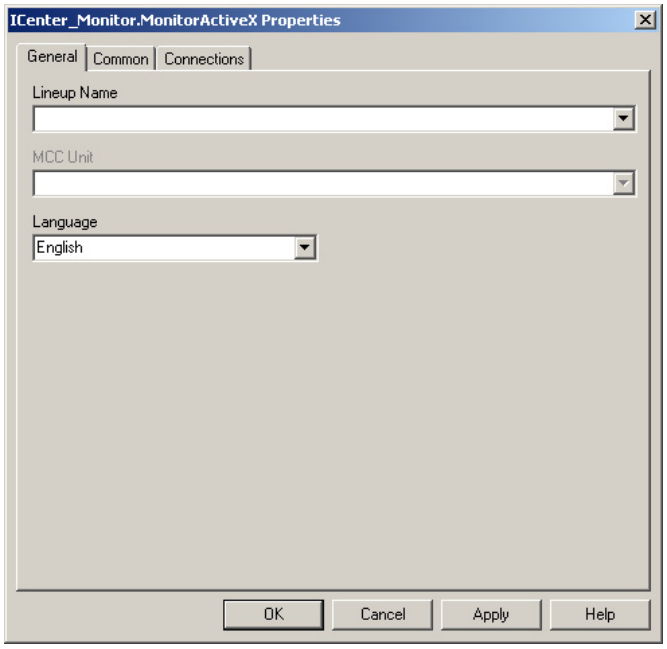
The IntelliCENTER Monitor view ActiveX control is added similarly as the Elevation view ActiveX control. To add the Monitor view ActiveX control, select ICenter_Monitor.MonitorActiveX from the ActiveX library as seen in Figure 5.

Figure 5 - IntelliCENTER ActiveX Monitor View Control



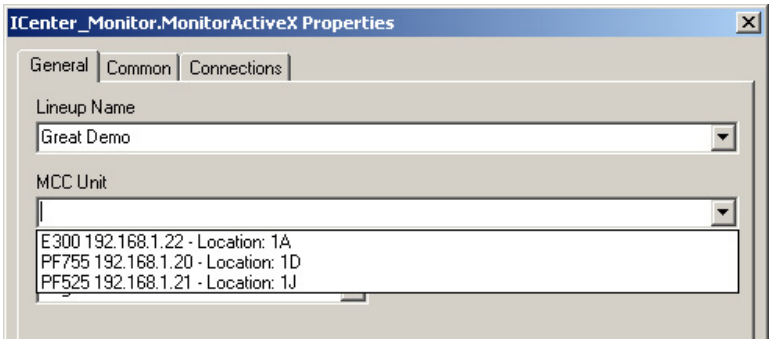
The Monitor view ActiveX control properties are used to configure the IntelliCENTER MCC lineup and specific intelligent MCC unit that is displayed with this instance of the ActiveX control. The properties page can be seen in Figure 6.

Figure 6 - IntelliCENTER Monitor View ActiveX Properties



After you select the Lineup Name from the list, the specific MCC unit is able to be selected from the pull-down menu as seen in Figure 7. This list is auto-generated the same as the lineup list.

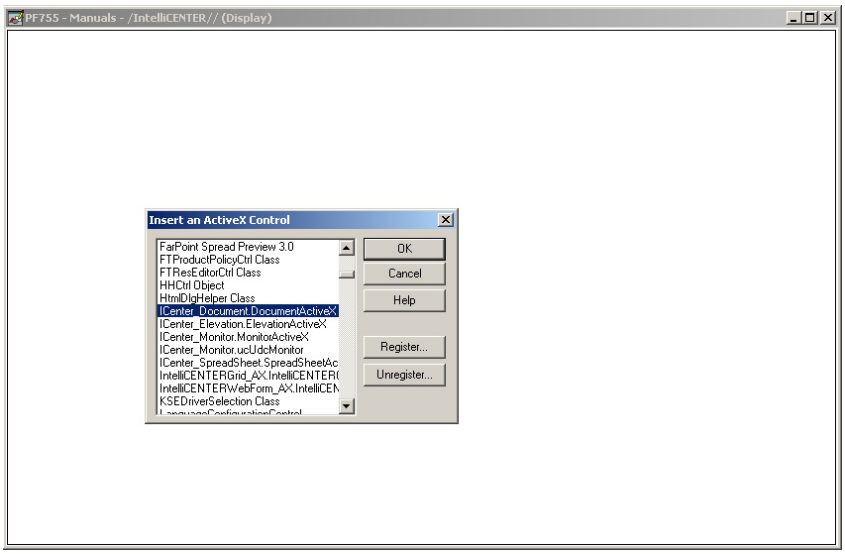
Figure 7 - IntelliCENTER Monitor View ActiveX MCC Unit Selection



Documents View

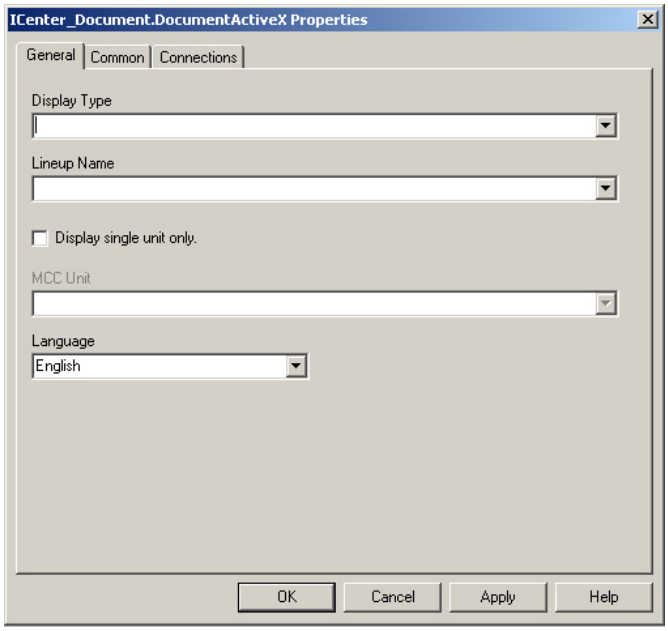
The various documents that are provided with the IntelliCENTER data files can be accessed through the HMI by using the ICenter_Document.Document ActiveX control. The Document view ActiveX control can be used for the CAD drawings, manuals, spare parts, and events lists for the entire IntelliCENTER MCC lineup or a specific IntelliCENTER MCC unit.

Figure 8 - IntelliCENTER ActiveX Document View Control



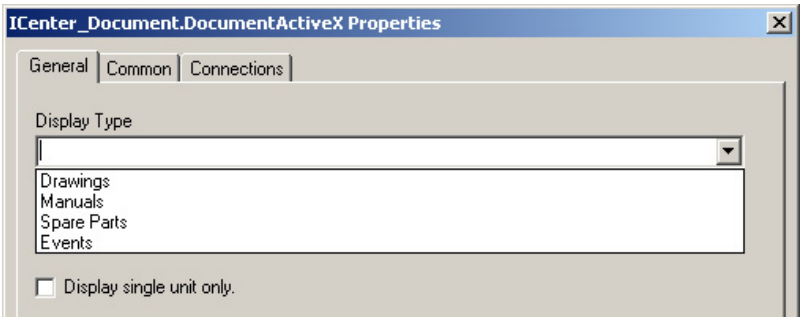
Once the Document view ActiveX control has been added to the display, the properties are used to configure which type of document, lineup, and/or MCC unit that this instance of the ActiveX control is used to display.

Figure 9 - IntelliCENTER Document View ActiveX Properties



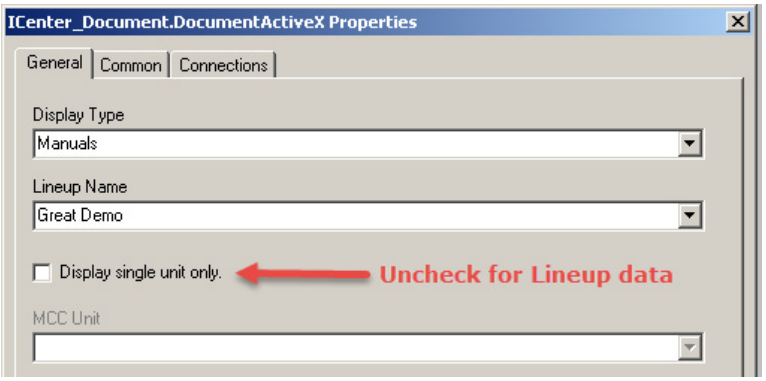
As stated, the Document view ActiveX control can be used to view the CAD drawings, manuals, spare parts, or events. This view is configured by using the 'Display Type' property setting as seen in Figure 10.

Figure 10 - IntelliCENTER Document View ActiveX Display Type Configuration



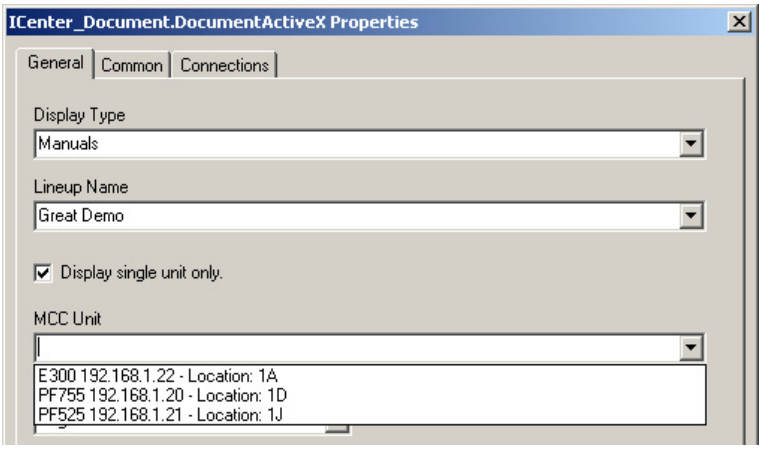
The Document view ActiveX control can be used to show the documents of an entire IntelliCENTER MCC lineup or one IntelliCENTER MCC unit. To show the documents of an entire lineup, make sure the checkbox next to 'Display single unit only' is unchecked as noted in Figure 11.

Figure 11 - IntelliCENTER Document View ActiveX Lineup vs. Single Unit



If 'Display single unit only' is selected, then the MCC Unit pull-down menu, which is seen in Figure 12, requires you to select a unit within the selected lineup.

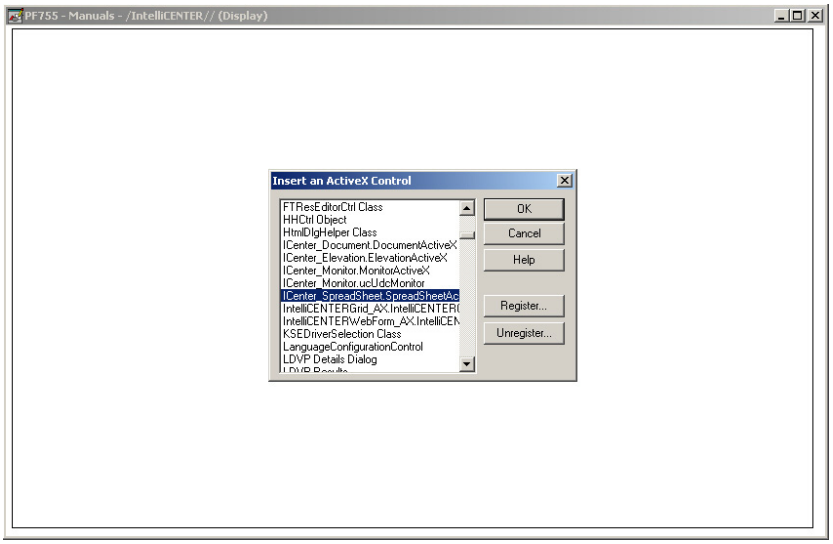
Figure 12 - IntelliCENTER Document View ActiveX MCC Unit Selection



Spreadsheet View

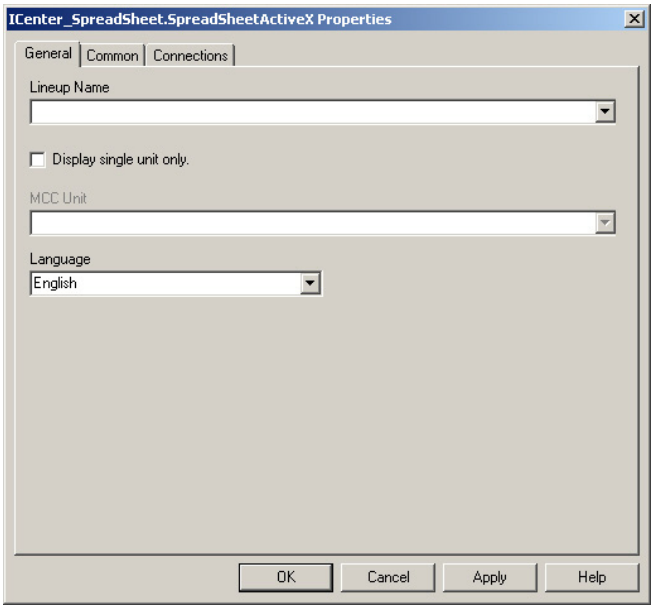
The last IntelliCENTER ActiveX control that can be added to an HMI is the Spreadsheet view. This view can be added with the ICenter_SpreadSheet.SpreadSheetActiveX control as seen in Figure 13.

Figure 13 - IntelliCENTER Spreadsheet View ActiveX Control



The Spreadsheet view ActiveX control properties are similar to the Monitor view ActiveX control, select either to display data for the entire IntelliCENTER MCC lineup or one IntelliCENTER MCC unit.

Figure 14 - IntelliCENTER Spreadsheet View ActiveX Properties



Advanced IntelliCENTER ActiveX Configuration

The IntelliCENTER ActiveX controls can take advantage of tag connections to allow for advanced configuration of each of the four available ActiveX controls. The tag connections, along with the use of macros, allow for a display to be used for multiple IntelliCENTER MCC lineups and/or MCC units.

FactoryTalk View HMI Tag Creation

HMI tags allow the HMI developer to change the values of connections that are made to the IntelliCENTER ActiveX control based on which IntelliCENTER MCC lineup or MCC unit the developer wants to display. By changing the value of the HMI tags, the control changes what data it loads. The data being written to the HMI tags indicate what ID represents the particular MCC lineup and/or unit. The ID is obtained from the IntelliCENTER ID program included with the IntelliCENTER software install disk. See Determine Lineup IDs and Device IDs for details on how to use this program.

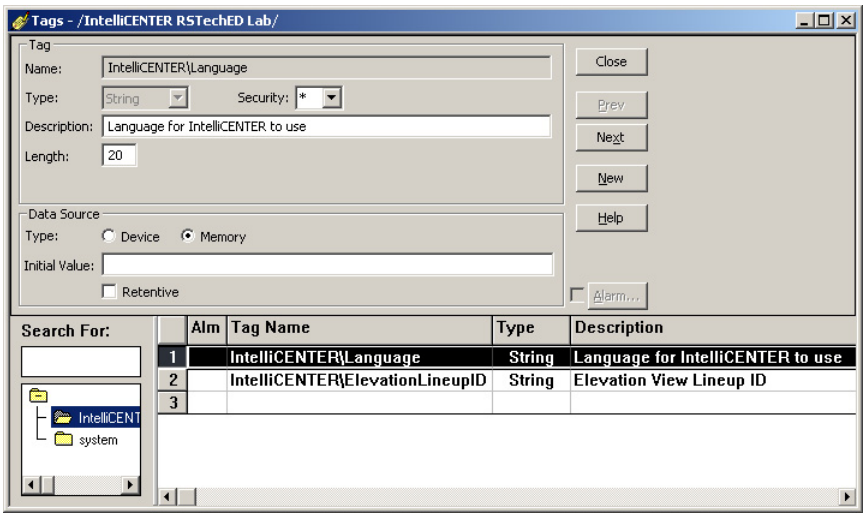
For more information on HMI tags, how to create them and their benefits, see 'Creating HMI tags' in the FactoryTalk View SE User Guide, publication ViewSE-UM006.

IntelliCENTER ActiveX HMI Tag configuration

The HMI tags that are created for use with the IntelliCENTER ActiveX controls require the following configuration:

- Type: String
- Length: 50
- Data Source Type: Memory

Figure 15 - HMI tag configuration that is required for IntelliCENTER ActiveX objects



Tags can be created for the following:

- ICLineupID
- ICLanguage
- ICDeviceID
- ICDisplayType
- ICSingleDevice

See pages 24...41 to determine the values that each tag type can handle.

IntelliCENTER ActiveX Object Connections

Use the Connections tab found in each IntelliCENTER ActiveX control property to map an ActiveX property to one of the HMI tags created earlier. The Connections tab lets you use HMI tags to set the lineup, MCC unit, language, display type, or single unit depending on which IntelliCENTER ActiveX control is being used.

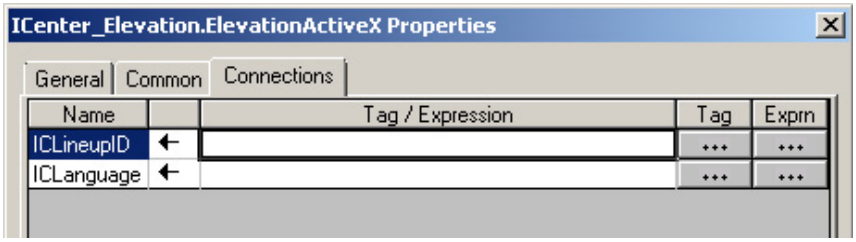
For more information on how to assign tags to object connections, see 'Assigning tags and expressions to objects' in the FactoryTalk View SE User Guide, publication ViewSE-UM006.

IMPORTANT	If no tag is connected to the ICLanguage connection, it defaults to the IntelliCENTER software setting made in the IntelliCENTER Preferences program.

Elevation Connections

The elevation ActiveX control has two connections: ICLineupID and ICLanguage.

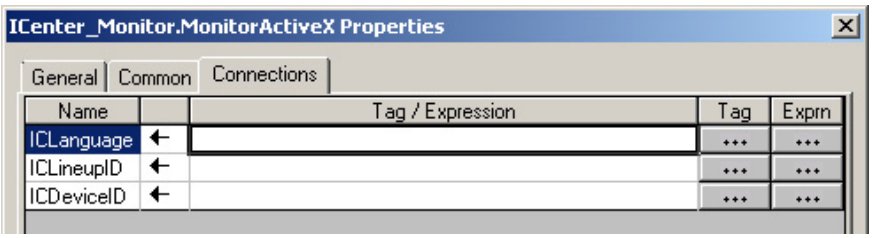
Figure 16 - Elevation Connections



Monitor Connections

The monitor ActiveX control has three connections: ICLineupID, ICLanguage, and ICDeviceID

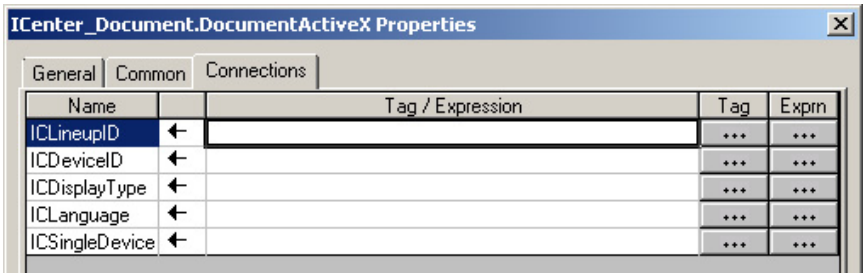
Figure 17 - Monitor Connections



Documents Connections

The documents ActiveX control has five connections: ICLineupID, ICLanguage, ICDeviceID, ICDisplayType, and ICSingleDevice.

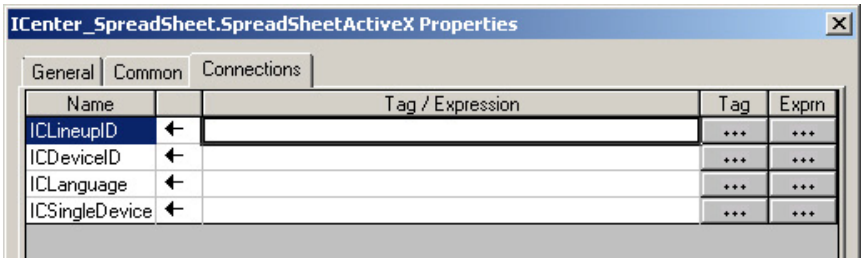
Figure 18 - Documents Connections



Spreadsheet Connections

The spreadsheet ActiveX control has four connections: ICLineupID, ICLanguage, ICDeviceID, and ICSingleDevice.

Figure 19 - Spreadsheet Connections



Determine Lineup IDs and Device IDs

The IntelliCENTER database uses IDs or GUIDs (globally unique IDs) to identify each piece of data that is stored in the database. Each lineup, device, drawing, manual, spare parts list, and event have IDs. The ICLineup ID and ICDeviceID properties, within the connections of the IntelliCENTER ActiveX control, need these ID values to setup the ActiveX controls to display for a particular device or lineup.

To modify the ID for the IntelliCENTER MCC lineup or device, the IntelliCENTER ID program is used. This program is included with the IntelliCENTER software installation. The program, once started, provides a list of the lineups that are currently installed along with the units in each of those lineups, when selected.

Figure 20 - IntelliCENTER ID Program

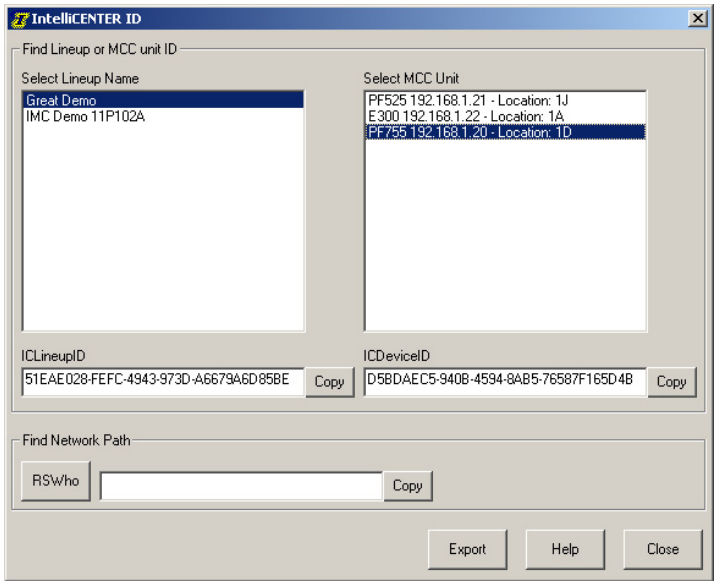


Figure 20 shows that the ICLineupID for the 'Great Demo' lineup is seen in the text box along with the ICDeviceID for the 'PF755, 192.168.1.20 - Location 1D'. When another lineup or device is selected, the associated ID is displayed. You can copy the ICLineupID and ICDeviceID by clicking Copy.

IMPORTANT	By clicking Copy, the text is copied into the standard clipboard. Make sure to paste the first ID before a second copy is made. If you make another copy before you paste the first copy, the first copy is cleared.

The copied ID is assigned to the HMI tags used for the connections in the IntelliCENTER ActiveX control. Multiple IDs can be assigned to an HMI tag by using macros within the HMI display, which is described in HMI Macro Programming.

FactoryTalk View SE Display Settings

The display settings of each HMI display that incorporates an IntelliCENTER ActiveX control requires the following startup commands when using HMI tags:

- Pause
- Invoke
  - Show() method

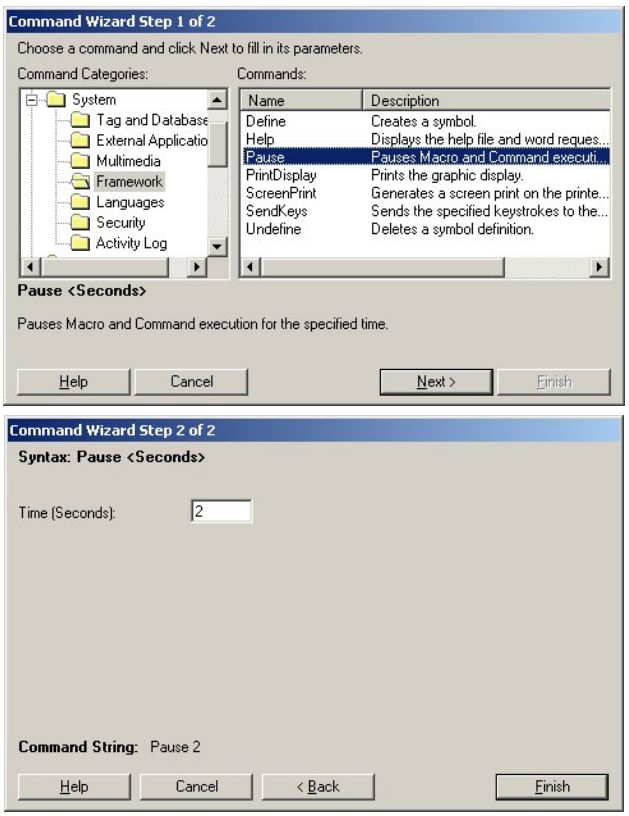
TIP The pause command is only needed when using macro programming.

For more information on runtime behavior of HMI displays, see 'Setting up the runtime behavior of a graphic display' in the FactoryTalk View SE User Guide, publication ViewSE-UM006.

Display Settings Behavior

The HMI display that incorporates the Elevation view ActiveX control uses a 2-second 'Pause' to make sure that the tag values are set before the ActiveX control is initialized.

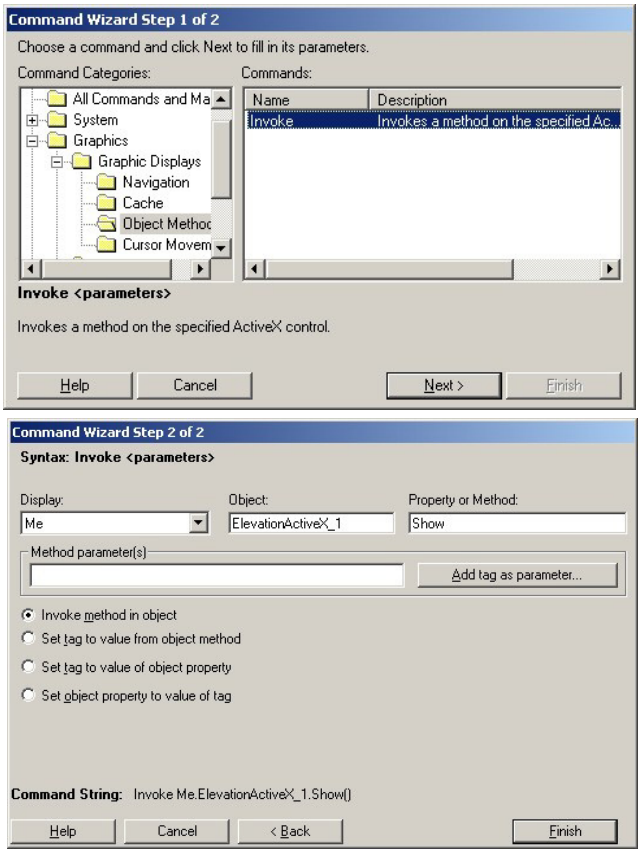
Figure 21 - Pause Startup Command



The Invoke command is added to the startup command after the pause command has been entered. The display is 'Me', the object is the name of your Elevation view ActiveX control and the method is Show(). See Figure 22 for an example.

IMPORTANT This same method can be used for any IntelliCENTER ActiveX controls.

Figure 22 - Invoke Startup Command



This final startup command is as follows:

Pause 2:Invoke
Me.ICenter_Elevation.Show()

HMI Macro Programming

The macro programming within FactoyTalk View SE is used to change the value of the HMI tags that are connected to the IntelliCENTER ActiveX objects being used. Configure the macro to run synchronously to prevent the ActiveX control from accessing the tag before the macro has written to it.

For more information on HMI macros, see 'Creating and using macros' in the FactoryTalk View SE User Guide, publication ViewSE-UM006.

Here is an example of a macro that was created to set the language and ICLineupID of the elevation ActiveX control:

```
Set IntelliCENTER\Language 'spanish'

Set IntelliCENTER\ElevationLineupID
'DDE203D0-A43E-4F96-B280-EDAC365BB129'

Display Elevation
```

The lineup ID is obtained from the IntelliCENTER ID program that is described in Determine Lineup IDs and Device IDs section.

Running the Macro

The programmed macro is assigned to execute on the press action of the designated push button that is created on the HMI display.

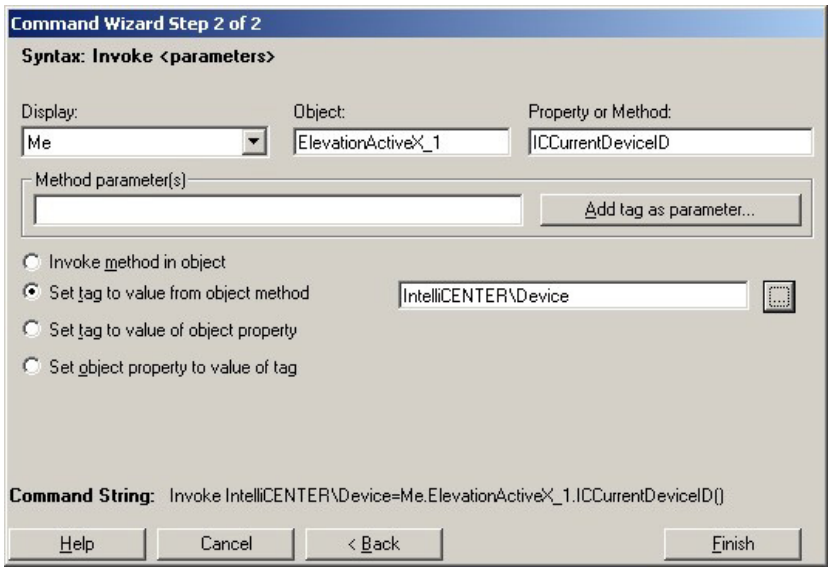
IMPORTANT If any of the IntelliCENTER ActiveX controls are on a display that is used more than once, adjust the display settings to run multiple copies.

Monitor Display Settings

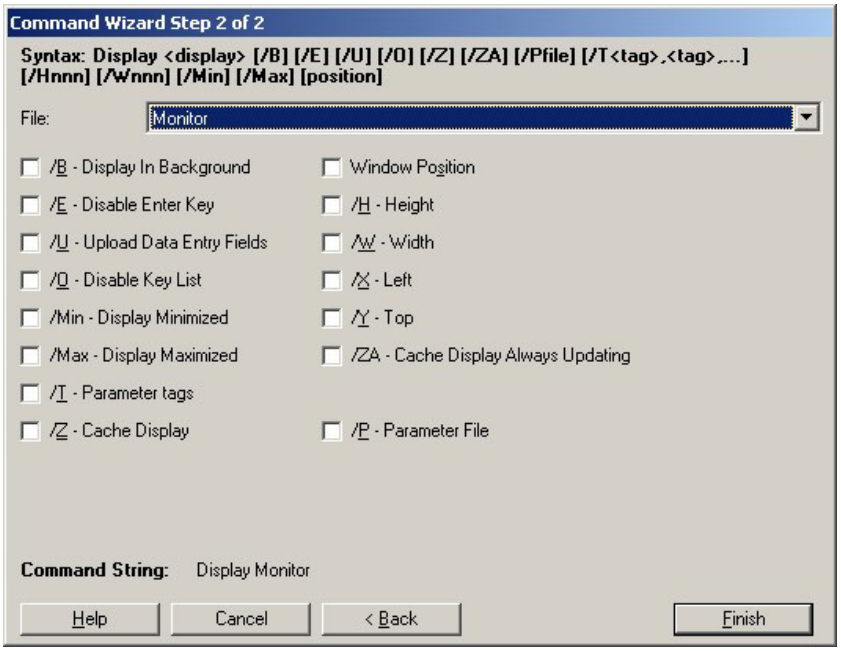
When the Monitor view ActiveX control is accessed through the elevation display that houses the Elevation view IntelliCENTER ActiveX control, the attribute ICCurrentDeviceID can be used instead of a macro to set ICDeviceID connection value.

The ICCurrentDeviceID passes the DeviceID of the unit that has been selected in the Elevation view ActiveX control. This attribute can be accessed by configuring the press action of the designated push button. The push button that is designed to open the Monitor view ActiveX control has two press-action commands.

- Obtaining DeviceID



- Displaying proper graphic display



IMPORTANT This same method can be used for any IntelliCENTER ActiveX controls.

Advanced IntelliCENTER ActiveX Integration - VBA

The IntelliCENTER ActiveX control can be added to an HMI environment by using Visual Basic for Applications (VBA). When you use VBA, the ActiveX control is added to the user forms instead of the display screens. The VBA user form incorporates an IntelliCENTER ActiveX control and can be initiated through a button.

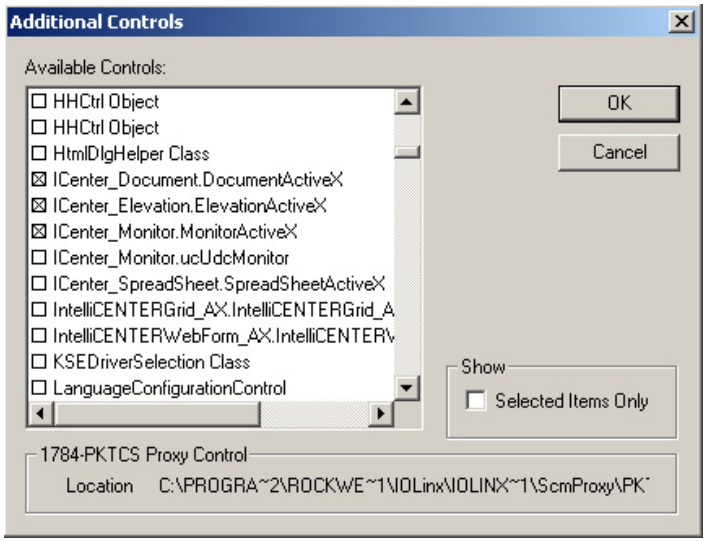
IMPORTANT When you integrate IntelliCENTER ActiveX controls by using VBA, you do not require the 3 second pause that is required for the method described in Advanced IntelliCENTER ActiveX Configuration.

More information about VBA within FactoryTalk View SE can be found in the Appendix of the FactoryTalk View SE User Guide, publication ViewSE-UM006.

VBA Elevation ActiveX Control

The Elevation view ActiveX control is integrated into a VBA user form that a push button initiates. This integration is done by adding a push button and by editing the VBA code for the button to create a user form. Within the user form, the first step is to add the IntelliCENTER tools to the Toolbox. The tools include the IntelliCENTER Elevation, Monitor, and Document ActiveX controls as seen in Figure 23.

Figure 23 - Adding IntelliCENTER Tools to the VBA Editor Toolbox



Once the controls are added to the Toolbox, the Elevation view ActiveX control can be added to the UserForm. The ICLineupID and ICDeviceID are required to define the lineup that is associated with the Elevation view ActiveX control. This information can be obtained from the IntelliCENTER ID program, see Determine Lineup IDs and Device IDs for information on how to use the program.

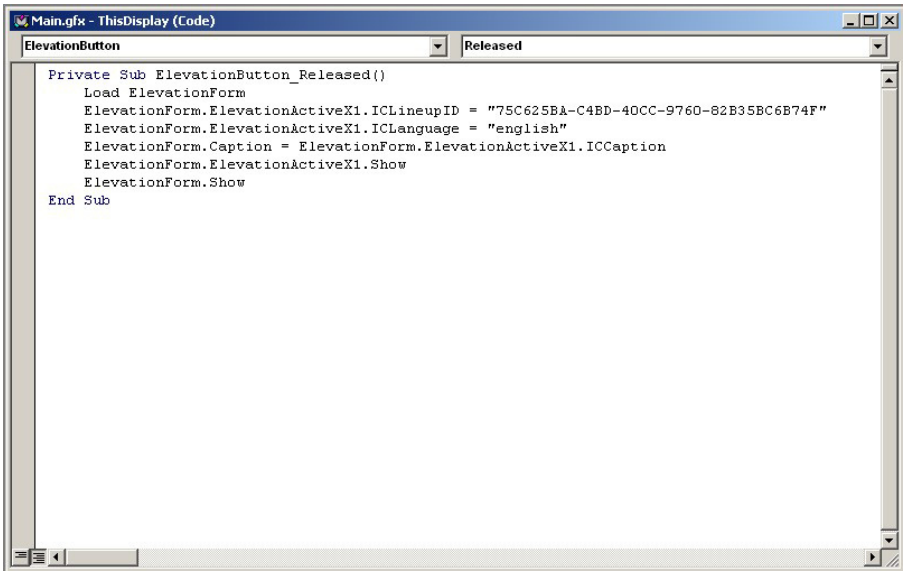
Setting HMI Tag Values in VBA

Define the following values by using VBA in the button release events for the Elevation ActiveX control to work properly:

- Set the Elevation ICLineupID
- Set the Elevation IC Language
- Call the Elevation Show method
- Call the user form show method

Figure 24 is an example of what the code looks like.

Figure 24 - Elevation ActiveX Control VBA Code



This same process can be used for the Monitor, Document, and Spreadsheet IntelliCENTER ActiveX controls.

Lineup and Device ID within VBA

The IntelliCENTER ID program provides an export feature that allows for all lineup ID and device ID information to be consolidated into a .CSV file. This file can be used as an array within VBA to reference the various lineups and devices programmatically.

FactoryTalk View ME IntelliCENTER ActiveX Integration

These instructions are used with FactoryTalk® View ME, version 5 or later.

The steps that are laid out in this Application Note referenced FactoryTalk View SE. The IntelliCENTER ActiveX controls can be used in the FactoryTalk View ME environment. But the runtime file cannot be executed on a system that does not run a 32-bit or above operating system.

There are several differences to note between FactoryTalk View SE and FactoryTalk View ME.

- Screen Resolution must be 1028x768 or greater. The IntelliCENTER ActiveX objects do not adjust if the resolution is lower. If the objects adjust, some information does not display.
- Macro Programming - There is no need for the Pause or Show methods to be implemented.

Object Model - IntelliCENTER Elevation View ActiveX Control

Provides a means to show the Elevation view to monitor the status of a lineup.

Remarks

The Elevation ActiveX control properties are set through the control property display or through programmable methods. The control automatically shows itself if all properties are set when going into a runtime environment. The Show method can be called after the control has entered a runtime environment and had changes made to its properties. If you decide to set properties after runtime is established, the Show method must be called to refresh the control.

Properties

The following are the properties of the Elevation ActiveX control.

ICCaption Property

Returns a String that contains the object name and the object lineup name.

- Syntax

  object.ICCaption
- Remarks

  The ICCaption property is read only and can only be called during runtime.

ICCurrentDeviceID Property

Returns a String that contains the ID of the currently selected device.

- Syntax

  object.ICCurrentDeviceID
- Remarks

  The ICCurrentDeviceID property is read only and can only be called during runtime.

ICLanguage

Returns or sets a string that contains the language to be used on the object.

- Syntax

  object.ICLanguage [= string]
- Remarks

  You can set the ICLanguage property by using the object property page or you can set it manually. The value must match one of the valid language strings in the IntelliCENTER database. The values in the database are: 'english', 'spanish', 'french,' and 'portuguese'. The default value of the property is 'english'.

ICLineupID

Returns or sets a String that contain the ID for the lineup.

- Syntax

  object.ICLineupID [= string]
- Remarks

  When setting the ICLineupID property, use an object property page or the IntelliCENTER ID program.

ICLineupName Property

Returns a String that contains the name of the object lineup.

- Syntax

  object.ICLineupName
- Remarks

  This property is read only and can only be called during runtime.

Methods

The following are the methods of the Elevation ActiveX control.

ICMaximize Method

Changes the size of the object to a maximized size inside the container object.

- Syntax

  object.ICMaximize
- Remarks

  The object height and width are set to a maximized size on the container with this method. The object must be placed at position 0,0 [left, top] on the container for the object to be displayed properly.

FullRtClickMenu Method

Used to display the entire right-click menu when you right-clicking on a unit.

- Syntax

  object.FullRtClickMenu(Boolean)

- Remarks

  This method can be used to enable the full right-click menu as opposed to the limited, default right-click menu. By enabling the full right-click menu, it is assumed that the HMI developer is programming more to catch the events that are related to the right-click menu that is shown. See ViewDrawings, ViewEvents, ViewManuals, ViewSpares, and ViewWebpage events that are related to this method.

ICCurrentDeviceType Method

Returns the device type of the currently selected device type.

- Syntax

  object.ICCurrentDeviceType

- Remarks

  This method can be called to indicate what type of device has been selected in the Elevation view. If you want to perform a function for a particular device type, this method can be useful when using ICCurrentDeviceID.

ICOffline Method

Returns or sets a Boolean, which determines if the Monitor view shows values in offline (disconnected) mode.

- Syntax

  object.ICOffline(Boolean)

- Remarks

  This method can be useful when testing ActiveX controls without having access to the actual MCC or network to do the testing. The control loads, but does not attempt to connect with the device if this method is called with the value of true passed.

IsNoMonView Method

Returns whether a particular device has a Monitor view or not.

- Syntax

  object.IsNoMonView(String)

  Where the string being passed is the ICDeviceID for a particular device within the Elevation view.

- Remarks

  This method can be used to determine if a Monitor view is supported for a particular device. For instance, if a user clicks a unit within the Monitor view, the ICCurrentDeviceID can be read. Then by using that ID, this method can be called to determine if the device has a supported Monitor view.

KillComms Method

Used to stop communications to the end devices.

- Syntax

  object.KillComms
- Remarks

  This method can be used to force the ActiveX control into a 'disconnected' state with the target devices.

ScrollLeft Method

Used to scroll the Elevation view once, to the left.

- Syntax

  object.ScrollLeft
- Remarks

  This method can be useful when the Elevation view contains many sections and a scrollbar is used to change which sections are being displayed. An HMI developer can use the ScrollLeft method to tie to a large button to more easily move the Elevation view to the left.

ScrollRight Method

Used to scroll the Elevation view once, to the right.

- Syntax

  object.ScrollRight
- Remarks

  This method can be useful when the Elevation view contains many sections and a scrollbar is used to change which sections are being displayed. An HMI developer can use the ScrollRight method to tie to a large button to more easily move the Elevation view to the right.

SetICLineupIDViaLineupName Method

Another technique to set the ICLineupID value.

- Syntax

  object.SetICLineupIDViaLineupName(string)

  Where the parameter relates to the name of the lineup that the user wants to set ICLineupID to.
- Remarks

  HMI developers can use this method when they have different copies of the same lineup with different lineup IDs, but the same lineup name. The developer can set the ID by setting the lineup name and have it apply to the lineup ID.

StartComms Method

Used to start communication to an end device.

- Syntax

  object.StartComms
- Remarks

  This method can force the ActiveX control into a 'connected' state with the target devices.

Show Method

Shows

- Syntax

  object.Show ()
- Remarks

  This method must be called after changes are made to any properties. By calling Show, the object reloads by using the new property values.

Events

The following are the events of the Elevation ActiveX control.

LoadComplete Event

This event is raised when the object has completed all initialization for display and communications.

- Remarks

  This event signals that the object has completed loading and communicating to the device. Now, properties such as ICCaption can be read.

LoadMonitorView Event

This event is raised when the user double-clicks on a unit which supports the Monitor view.

- Remarks

  This event returns the ID of the device that was double-clicked in the Elevation view. The HMI developer can then use this ID and write to an HMI tag, then load the related Monitor view display.

ViewDrawings Event

This event is raised when you select to view drawings from the right-click menu.

- Remarks

  This event relates to when the FullRtClickMenu method has been set to true. This lets you see new items in the right-click menu. One of these items relates to viewing the drawings. If the menu is clicked, this event fires. The HMI developer can then use this ID and write to an HMI tag, then load the related Document display.

ViewEvents Event

This event is raised when you select to view events from the right-click menu.

- Remarks

  This event relates to when the FullRtClickMenu method has been set to true. This lets you see new items in the right-click menu. One of these items relates to viewing events. If the menu is clicked, this event fires. The HMI developer can then use this ID and write to an HMI tag, then load the related Document display.

ViewManuals Event

This event is raised when you select to view manuals from the right-click menu.

- Remarks

  This event relates to when the FullRtClickMenu method has been set to true. This lets you see new items in the right-click menu. One of these items relates to viewing manuals. If the menu is clicked, this event fires. The HMI developer can then use this ID and write to an HMI tag, then load the related Document display.

ViewSpares Event

This event is raised when you select to view spares from the right-click menu.

- Remarks

  This event relates to when the FullRtClickMenu method has been set to true. This lets you see new items in the right-click menu. One of these items relates to viewing spares. If the menu is clicked, this event fires. The HMI developer can then use this ID and write to an HMI tag, then load the related Document display.

ViewWebpage Event

This event is raised when you select to view a device webpage from the right-click menu.

- Remarks

  This event relates to when the FullRtClickMenu method has been set to true. This lets you see new items in the right-click menu. One of these items relates to viewing device webpage. If the menu is clicked, this event fires. The HMI developer can then use the path provided to open up a webpage being referenced.

Object Model - IntelliCENTER Monitor ActiveX Control

Provides a means of monitoring a device in a lineup.

Remarks

The Monitor control properties can be set through the control property page or through programmable methods. The control automatically shows itself if all its properties are set when going into a runtime environment. The control Show method can be called after the control has entered a runtime environment and had changes made to its properties. If you decide to set properties after runtime has been established, the Show method must be called to refresh the control. The parameters for the device can be edited by using the right click menu.

Properties

The following are the properties of the Monitor ActiveX control.

ICCaption Property

Returns a String that contains the object name and the object lineup name.

- Syntax

  object.ICCaption
- Remarks

  The ICCaption property is read only and can only be called during runtime.

ICDevice Property

Returns a String that contains the name of the MCC unit of the object.

- Syntax

  object.ICDevice
- Remarks

  This property is read only and can only be called during runtime.

ICDeviceID Property

Returns or sets a String that contains the GUID for the device.

- Syntax

  object.ICDeviceID [= string]
- Remarks

  When setting the ICDevice property, use an object property page or the IntelliCENTER ID identifier utility.

ICLanguage

Returns or sets a string that contains the language to be used on the object.

- Syntax

  object.ICLanguage [= string]
- Remarks

  You can set the ICLanguage property by using the object property page or you can set it manually. The value must match one of the valid language strings in the IntelliCENTER database. The values in the database are: 'english', 'spanish', 'french,' and 'portuguese'. The default value of the property is 'english'.

ICLineupID

Returns or sets a String that contains the ID for the lineup.

- Syntax

  object.ICLineupID [= string]
- Remarks

  When setting the ICLineupID property, use an object property page or the IntelliCENTER ID program.

ICLineupName Property

Returns a String that contains the name of the object lineup.

- Syntax

  object.ICLineupName
- Remarks

  This property is read only and can only be called during runtime.

Method

The following are the methods for the Monitor ActiveX control.

AuxOnlyID

Used to display only the Monitor view for one of the auxiliary devices within a unit, if multiple devices are present.

- Syntax

  Object.AuxOnlyID(string)

  Where the string being passed refers to an auxiliary device within a unit (for a given ICDeviceID).
- Remarks

  Use this method to cause the Monitor view to display only for a particular device within a unit (for a given ICDeviceID). If the passed value matches the ID of a child device within the unit, then only that device is shown.

ICOffline Method

Returns or sets a boolean that determines if the Monitor view shows values in offline (disconnected) mode.

- Syntax

  object.ICOffline(Boolean)
- Remarks

  This method can be useful when testing ActiveX controls without having access to the actual MCC or network to do the testing. The control loads, but does not attempt to connect with the device if this method is called with the value of true passed.

IgnoreAuxDevices Method

Used to display only the Monitor view for the main device within a unit, if multiple devices are present.

- Syntax

  Object.IgnoreAuxDevices(Boolean)
- Remarks

  If true, then only the main device within a unit is displayed by the Monitor view. This method only matters if there are multiple devices within a particular unit (ICDeviceID).

KillComms Method

Used to stop communications to the end devices.

- Syntax

  object.KillComms
- Remarks

  This method can be used to force the ActiveX control into a 'disconnected' state with the target devices.

StartComms Method

Used to start communications to an end devices.

- Syntax

  object.StartComms
- Remarks

  This method can be used to force the ActiveX control into a 'connected' state with the target devices.

Show Method

Shows

- Syntax

  object.Show ()
- Remarks

  This method must be called after changes are made to any properties. When you call Show, the object to reloads by using the new property values.

Events

The following are the events for the Monitor ActiveX control.

LoadComplete Event

This event is raised when the object has completed all initialization for display and communications.

- Remarks

  This event signals that the object has completed loading and communicating to the device. Now, properties such as ICCaption can be read.

Object Model - IntelliCENTER Document ActiveX Control

Allows you to view drawings, manuals, spare parts, and events for a specific lineup or device.

Remarks

The Document control properties can be set through the control property page or through programmable methods. The control shows itself automatically if all its properties are set when going into a runtime environment. The control Show method can be called after the control has entered a runtime environment and had changes made to its properties. If you decide to set properties after runtime has been established, the Show method must be called to refresh the control.

Properties

The following are the properties for the Documents ActiveX control.

ICCaption Property

Returns a String that contains the object name and the object lineup name.

- Syntax

  object.ICCaption
- Remarks

  The ICCaption property is read only and can only be called during runtime.

ICDevice Property

Returns a String that contains the name of the MCC unit of the object.

- Syntax

  object.ICDevice
- Remarks

  This property is read only and can only be called during runtime.

ICDeviceID Property

Returns or sets a String that contains the GUID for the device.

- Syntax

  object.ICDeviceID [= string]
- Remarks

  When setting the ICDeviceID property, use an object property page or the IntelliCENTER ID identifier utility.

ICDisplayType Property

Returns or sets a string value that contains the display type of the object.

- Syntax

  object.ICDisplayType [= string]

  The string values that determine the display type are as follows:

Display Type	Description
cad	Display for drawings
event	Display for the event log
manual	Display for device manuals
spares	Display for spare parts

- Remarks

  The display types are case-sensitive and can be set using the object property page or manually.

ICLanguage

Returns or sets a string that contains the language to be used on the object.

- Syntax

  object.ICLanguage [= string]
- Remarks

  You can set the ICLanguage property by using the object property page or you can set it manually. The value must match one of the valid language strings in the IntelliCENTER database. The values in the database are: ‘english’, ‘spanish’, ‘french,’ and ‘portuguese’. The default value of the property is ‘english’.

ICLineupID

Returns or sets a String that contains the ID for the lineup.

- Syntax

  object.ICLineupID [= string]
- Remarks

  When setting the ICLineupID property, use an object property page or the IntelliCENTER ID program.

ICLineupName Property

Returns a String that contains the name of the object lineup.

- Syntax

  object.ICLineupName
- Remarks

  This property is read only and can only be called during runtime.

ICSingleDevice Property

Returns or sets a Boolean value that contains the value that determines if the object is associated with one device or the entire lineup.

- Syntax

  object.ICSingleDevice[= Boolean]
- Remarks

  The default value for the ICSingleDevice property is False. If false, all devices are displayed. If true, ICDeviceID must be set and only information for that device is displayed.

Methods

The following are the methods for the Documents ActiveX control.

ICMaximize Method

Changes the size of the object to a maximized size inside the container object.

- Syntax

  object.ICMaximize
- Remarks

  The object height and width are set to a maximized size on the container with this method. The object must be placed at position 0,0 [left, top] on the container for the object to be displayed properly.

Show Method

Shows

- Syntax

  object.Show ()
- Remarks

  This method must be called after changes are made to the properties. When you call Show, the object to reloads by using the new property values.

Events

The following is the event for the Documents ActiveX control.

LoadComplete Event

This event is raised when the object has completed all initialization for display and communications.

- Remarks

  This event signals that the object has completed loading and communicating to the device. Now, properties such as ICCaption can be read.

Object Model - IntelliCENTER Spreadsheet ActiveX Control

Displays information for one device or an entire lineup in tabular form.

Remarks

The Spreadsheet control properties can be set through the control property page or through programmable methods. The control shows itself automatically if all its properties are set when going into a runtime environment. The control Show method can be called after the control has entered a runtime environment and had changes made to its properties. If you decide to set properties after runtime has been established, the Show method must be called to refresh the control. The control is read only - no editing can be done to the information content displayed.

Properties

The following are the properties for the Spreadsheet ActiveX control.

ICCaption Property

Returns a String that contains the object name and the object lineup name.

- Syntax

  object.ICCaption
- Remarks

  The ICCaption property is read only and can only be called during runtime.

ICDevice Property

Returns a String that contains the name of the MCC unit of the object.

- Syntax

  object.ICDevice
- Remarks

  This property is read only and can only be called during runtime.

ICDeviceID Property

Returns or sets a String that contains the GUID for the device.

- Syntax

  object.ICDeviceID [= string]
- Remarks

  When setting the ICDevice property, use an object property page or the IntelliCENTER ID identifier utility. See ICSingleDevice property.

ICLanguage

Returns or sets a string that contains the language to be used on the object.

- Syntax

  object.ICLanguage [= string]
- Remarks

  You can set the ICLanguage property by using the object property page or you can set it manually. The value must match one of the valid language strings in the IntelliCENTER database. The values in the database are; 'english', 'spanish', 'french,' and 'portuguese'. The default value of the property is 'english'.

ICLineupID

Returns or sets a String that contains the ID for the lineup.

- Syntax

  object.ICLineupID [= string]
- Remarks

  When setting the ICLineupID property, use an object property page or the IntelliCENTER ID program.

ICLineupName Property

Returns a String that contains the name of the object lineup.

- Syntax

  object.ICLineupName
- Remarks

  This property is read only and can only be called during runtime.

Methods

The following are the methods for the Spreadsheet ActiveX control.

ICMaximize Method

Changes the size of the object to a maximized size inside the container object.

- Syntax

  object.ICMaximize
- Remarks

  The object height and width are set to a maximized size on the container with this method. The object must be placed at position 0,0 [left, top] on the container for the object to be displayed properly.

Show Method

Shows

- Syntax

  object.Show ()
- Remarks

  This method must be called after changes are made to any properties. When you call Show, the object reloads by using the new property values.

Events

The following is the event for the Spreadsheet ActiveX control.

LoadComplete Event

This event is raised when the object has completed all initialization for display and communications.

- Remarks

  This event signals that the object has completed loading and communicating to the device. Now, properties such as ICCaption can be read.

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation.

Resource	Description
FactoryTalk ViewSE User Guide, publication ViewSE-UM006	Provides information on using FactoryTalk View SE software.
IntelliCENTER software user manual, publication MCC-UM002	Provides information on how to install and use IntelliCENTER software.
IntelliCENTER EtherNet/IP Motor Control Centers Reference Manual, publication MCC-RM001	Describes the EtherNet/IP IntelliCENTER motor control center with a focus on the system architecture and integration into your plant.
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications website, http://www.ab.com	Provides declarations of conformity, certificates, and other certification details.

You can view or download publications at http://www.rockwellautomation.com/literature/. To order paper copies of technical documentation, contact your local Allen-Bradley distributor or Rockwell Automation sales representative.

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com/ for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation Assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/rockwellautomation/support/overview.page, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to help ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

Rockwell Automation maintains current product environmental information on its website at http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication MCC-AT002A-EN-P - October 2015