FactoryTalk View Machine Edition and PanelView Plus - Advanced Lab

For Classroom Use Only!
FactoryTalk View Machine Edition and PanelView Plus - Advanced Lab

Contents

Before you begin .................................................................................................................................................. 7
About this lab ...................................................................................................................................................... 7
Tools & prerequisites ......................................................................................................................................... 8
Document conventions ....................................................................................................................................... 10
FactoryTalk® View Machine Edition ............................................................................................................... 11
PanelView™ Plus 6 ........................................................................................................................................... 11
FactoryTalk® ViewPoint .................................................................................................................................. 13
Converting the compiled application to an application you can work on ....................................................... 15
Opening the application in FactoryTalk View Studio ......................................................................................... 21
Modifying the design time communications configuration .............................................................................. 23
Modifying the historical data logging .............................................................................................................. 39
Adding machine troubleshooting information .................................................................................................. 45
Running the application on the PanelView Plus 6 Terminal ............................................................................. 59
Building the runtime .MER file ....................................................................................................................... 59
Download the runtime .MER file to a PanelView Plus 6 terminal ................................................................. 62
Run the application on a PanelView Plus 6 terminal ....................................................................................... 65
Testing the motor changes .............................................................................................................................. 67
Viewing a PDF file on the terminal .................................................................................................................. 68
Examining the data logs using the FTP Server ................................................................................................. 72
Starting and stopping the DataStore Plus ActiveX .......................................................................................... 78
Open the application ........................................................................................................................................ 80
Examine the graphic display containing the ME_IPInfo ActiveX control ........................................... 81
Change which graphic display is called at startup ................................................................................ 86
Examine the Global Connections ........................................................................................................ 88
Running the application on the PanelView Plus 6 Terminal ............................................................... 92
Building the runtime .MER file ........................................................................................................... 92
Download the runtime .MER file to a PanelView Plus 6 terminal ....................................................... 95
Run the application on a PanelView Plus 6 terminal ............................................................................. 98
Setting up a FactoryTalk ViewPoint ME application ........................................................................... 103
Publishing a FactoryTalk View ME Application .................................................................................. 106
Configuring FactoryTalk ViewPoint ME security ................................................................................ 110
Creating a runtime file with ViewPoint content .................................................................................... 116
Running an application in a browser ..................................................................................................... 124
Using browser features with FactoryTalk ViewPoint .......................................................................... 131
Trending ................................................................................................................................................ 135
Inactivity timeout .................................................................................................................................. 140

Web Based Language Translation (20 minutes) .................................................................................... 148
Open the application and export the languages ..................................................................................... 149
Open the FactoryTalk View Translation Utility ................................................................................... 156
Import the translated strings into Studio ............................................................................................... 164
Test run the application ........................................................................................................................ 167

Email ActiveX control (20 minutes) ...................................................................................................... 169
Create the Runtime (.MER) file .......................................................................................................... 183
Download the runtime .MER file to a PanelView Plus 6 terminal ....................................................... 185
Run the application on a PanelView Plus 6 terminal ............................................................................. 188

Auto Logout to designated screen (15 minutes) ................................................................................. 193
Open the application and examine the configuration ............................................................................. 193
Test Run the Application ....................................................................................................................... 197

Web Browser ActiveX control (20 minutes) .......................................................................................... 203
Create the Runtime (.MER) file .......................................................................................................... 216
Download the runtime .MER file to a PanelView Plus 6 terminal ................................................................. 218
Run the application on a PanelView Plus 6 terminal ...................................................................................... 221

**Working with security (15 minutes) ................................................................. 227**

Open the InstantFizz Advanced application ......................................................................................... 228

View the security code for a display ........................................................................................................... 229

Create a new user account and assign security codes .............................................................................. 230

Exercise security ...................................................................................................................................... 233

Using security groups ............................................................................................................................... 236

Testing the application on desktop ........................................................................................................... 241

**Improving productivity with parameter passing (30 minutes) .......................................................... 242**

Open the InstantFizz Advanced application ......................................................................................... 243

Add the PIDE faceplates ............................................................................................................................ 244

Testing the PIDE faceplates ....................................................................................................................... 246

Investigate the numeric input configuration .............................................................................................. 248

Investigate “carry forward” parameters .................................................................................................. 251

Fixing the PIDE faceplate errors ............................................................................................................. 253

Testing the Application ............................................................................................................................ 261

**Logging data (20 minutes) .................................................................................................................. 262**

Open the InstantFizz Advanced application ......................................................................................... 263

Creating the Data Log Model ................................................................................................................... 264

Configure application to start the Data Log Model on startup .............................................................. 270

Examine the Trend object properties ...................................................................................................... 271

Testing the application ............................................................................................................................ 275
Additional Information A: How does data logging work? ......................................................... 278
Additional Information B: Differences in Data Models ................................................................. 281
Additional Information C: Application Documenter .................................................................... 283
Additional Information D: Studio overview ................................................................................ 285
Additional Information E: Configuration/Setup Guide ................................................................. 287
Lab Setup and Configuration Information .................................................................................... 288
Lab Information .......................................................................................................................... 288
Hardware Configuration per Student ........................................................................................... 288
Computer/Host Settings ............................................................................................................... 289
Basic Setup Diagram .................................................................................................................. 289
Application/Programming ........................................................................................................... 290
Additional Equipment Required per Workstation ....................................................................... 290
RSLinx - DDE/OPC Topic Configuration .................................................................................... 290
RSLinx - Driver Configuration ...................................................................................................... 291
RSLinx Enterprise - Shortcut Configuration .................................................................................. 291
Application Versions .................................................................................................................. 291
Required Pre-Lab Configuration .................................................................................................. 292
Before you begin

During this lab, you can choose to learn about the FactoryTalk® View Studio for Machine Edition (ME) skills that are needed to add additional functionality to an existing FactoryTalk View Machine Edition application. You will review some of the more advanced capabilities and new functionality in FactoryTalk View ME v6.0 and have an opportunity to explore the features of FactoryTalk View ME within FactoryTalk® ViewPoint, our “thin client,” or browser-based, HMI solution. FactoryTalk ViewPoint ME is used to publish web applications for FactoryTalk View ME projects that run on PanelView™ Plus terminals.

Lastly, a set of optional topics are available where you can pick and choose which FactoryTalk View ME features you wish to work with and learn more about.

All chapters take advantage of the new PanelView Plus 6 terminals that are located at each workstation.

About this lab

Scenario 1 expands your FactoryTalk View ME knowledge and explores some of the new features available starting in the v6.0 release.

Scenario 2 utilizes an activeX control to automate the custom startup of a FactoryTalk View ME application.

Scenario 3 will take you through the basic and advanced features of FactoryTalk ViewPoint ME including what new features are available starting in the v2.0 release.

Pick and Choose allows you to select any section, in any order, to cater to the areas of FactoryTalk View ME you would like to concentrate on.

This lab takes approximately 2 hours to complete.
**Tools & prerequisites**

**Hardware**
This hands-on lab does not require a Logix5000 controller; however, a Logix 500 controller could be used in place of SoftLogix 5800.

A PanelView Plus 6 terminal is required.

**Software**
To complete this lab you must use the following hardware and software:

- A Microsoft Windows 7 professional computer
- Ethernet connection between computer and PanelView Plus 6 terminal
- FactoryTalk View Machine Edition Studio v6.10 (CPR9 SR4)
- FactoryTalk ViewPoint ME v2.10 (CPR9 SR4)
- FactoryTalk Services Platform 2.50 (CPR9 SR5)
- RSLinx Enterprise v5.40 (CPR9 SR4)
- RSLinx Classic v2.59 (CPR9 SR5)
- RSLinx 5000 v19.01
- SoftLogix 5800 v19.01
- FactoryTalk View Translation Utility version 1.3.3
Important User Information

This documentation, whether, illustrative, printed, “online” or electronic (hereinafter “Documentation”) is intended for use only as a learning aid when using Rockwell Automation approved demonstration hardware, software and firmware. The Documentation should only be used as a learning tool by qualified professionals.

The variety of uses for the hardware, software and firmware (hereinafter “Products”) described in this Documentation, mandates that those responsible for the application and use of those Products must satisfy themselves that all necessary steps have been taken to ensure that each application and actual use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards in addition to any applicable technical documents.

In no event will Rockwell Automation, Inc., or any of its affiliate or subsidiary companies (hereinafter “Rockwell Automation”) be responsible or liable for any indirect or consequential damages resulting from the use or application of the Products described in this Documentation. Rockwell Automation does not assume responsibility or liability for damages of any kind based on the alleged use of, or reliance on, this Documentation.

No patent liability is assumed by Rockwell Automation with respect to use of information, circuits, equipment, or software described in the Documentation.

Except as specifically agreed in writing as part of a maintenance or support contract, equipment users are responsible for:

- properly using, calibrating, operating, monitoring and maintaining all Products consistent with all Rockwell Automation or third-party provided instructions, warnings, recommendations and documentation;
- ensuring that only properly trained personnel use, operate and maintain the Products at all times;
- staying informed of all Product updates and alerts and implementing all updates and fixes; and
- all other factors affecting the Products that are outside of the direct control of Rockwell Automation.

Reproduction of the contents of the Documentation, in whole or in part, without written permission of Rockwell Automation is prohibited.
Document conventions

Throughout this workbook, we have used the following conventions to help guide you through the lab materials.

**Note:** If the mouse button is not specified in the text, you should click on the left mouse button.

<table>
<thead>
<tr>
<th>This style or symbol:</th>
<th>Indicates:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words shown in bold italics (e.g., <strong>RSLogix 5000 or OK</strong>)</td>
<td>Any item or button that you must select, click on, or a menu name from which you must choose an option or command. This will be an actual name of an item that you see on your screen or in an example.</td>
</tr>
<tr>
<td>Words shown in bold (e.g., <strong>Communication Setup</strong>)</td>
<td>This is the name of an item that you see on your screen or in an example.</td>
</tr>
</tbody>
</table>
| Words shown underlined and enclosed in single quotes (e.g., ‘**Controller1**’) | An entry that you must type in the specified field. This is information that you must supply based on your application (e.g., a variable).  
  **Note:** When you type the text in the field, remember that you do not need to type the quotes; simply type the words that are contained within them (e.g., Controller1). |
| This is sample text.                               | Text that appears inside of a gray box is supplemental information regarding the lab materials or learning goals; the information is not required for you to complete the lab exercises. The supplemental text may provide you with helpful hints that can make it easier for you to use this product. |
**FactoryTalk® View Machine Edition**

FactoryTalk® View Machine Edition (ME) is a machine-level HMI product that supports both open and dedicated operator interface solutions for monitoring and controlling individual machines or small processes. It provides a consistent operator interface across multiple platforms, including Microsoft® Windows® CE and 32-bit Microsoft® Windows® 7, XP, Vista solutions.

FactoryTalk View Machine Edition contains two components:

- **FactoryTalk View® Studio** - This is the development environment containing the tools you need for creating all aspects of a human-machine interface (HMI), including graphic displays, trends, alarm reporting and real-time animation. It also provides tools for testing individual displays and entire applications. When development is completed, a run-time (.MER) file created to run on a PanelView Plus or personal computer.

- **FactoryTalk View® Machine Edition Station** - This is the run-time environment. FactoryTalk View Machine Edition Station executes the run-time (.MER file) application. FactoryTalk View Machine Edition Station is embedded in PanelView Plus terminals. Run-time applications may also be executed on a personal computer. Executing run-time applications on a personal computer requires additional software licenses.

**PanelView™ Plus 6**

The PanelView Plus are operator interface terminals designed to optimize system development, performance, and efficiency. The PanelView Plus 6 line is the latest addition to Rockwell Automation’s versatile family of Allen-Bradley PanelView operator interface displays for machine level operator terminal applications in industrial environments.

Enhancements to the hardware platform, embedded operating system and development environment enable users to:

- **Take time and costs out of application development** - PanelView Plus 6 dramatically reduces development, setup time and troubleshooting time through features like tag re-use, complete Symbol Factory graphic library and pre-built face plates.

- **Run their processes more effectively** - Improved hardware performance delivers up 30% faster screen response, enabling operators to navigate through screens more quickly, and can help avoid maintenance calls resulting from mistakenly pressing inputs multiple times. In addition, new capabilities wring more value from your process.

- **Reduce maintenance costs** - New features like on-board pdf capability and remote user access enable context-sensitive help and can avoid on-site visits to get processes back up and running faster when things go wrong.
**FactoryTalk® ViewPoint**

FactoryTalk® ViewPoint is an add-on to FactoryTalk View ME running on PanelView Plus that provides for a fully scalable, fully animated, view of existing applications from a Web browser.

To make information about your plant or process available on demand from a Web browser in your office, home, or hotel, all you have to do is select the FactoryTalk View graphic displays you want to make ready for the Web, and then publish the displays to the FactoryTalk ViewPoint Server which runs on a PanelView Plus.

There is no installation of any Rockwell Software products on the browser computer: all you need to connect to a published FactoryTalk ViewPoint Web application is the name (or IP address) of the PanelView Plus hosting the FactoryTalk ViewPoint Server that stores the application.

Once you enter a simple address directly into your Web browser using the name or IP address, the browser will connect to the published web application and open the initial display selected. Use navigation buttons in the application to view other published displays, or use the web browser's navigation tools.

Starting in FactoryTalk ViewPoint v2.0 displays can be either read-only or read-write depending on how the application is configured. Also starting in v2.0 is how security can be configured for display access and read-write capability according to user group membership.
Scenario 1: Modifying an existing application – 45 min.

About This Scenario

You are an engineer sent to site from an OEM who has been asked by the end-user to modify an existing application by adding the Setpoint of the temperature from Zone 6 to the historical datalogging and improve the look of a motor control. The end user would also like the Shift Manager to have access to product documentation at runtime.

You arrive on site, however you do not have the application to work on with you. The application is loaded in the PanelView Plus terminal and you know that you can upload the application and recreate the design time application files.

For the purpose of this lab we have already used the Transfer Utility to upload the runtime application .MER file to your lab computer.

Use the Transfer Utility to:

- Transfer your run-time application file (with the extension .mer) to and from the PanelView Plus terminal where the project will run
- Specify that the downloaded .mer file will run automatically after the download is completed and whenever the terminal is restarted
- Transfer Windows True Type font files (with the extensions .ttf and .ttc) to a terminal
- Compare a run-time application on a terminal with a run-time application on the desktop computer
- You can download multiple projects (one at a time) to the same run-time computer, and you can download projects while a project is running. Even a new copy of a project currently running on the run-time computer can be downloaded for use the next time the project is started.
- The Transfer Utility uses the RSLinx Enterprise Communication tree to specify the upload/download source and destination locations.
Converting the compiled application to an application you can work on.

In this section you will convert the compiled MER file into the design time files used by FactoryTalk View Studio.

The Application Manager tool will not only convert the application but also the security group/users that are in the application on the terminal.

The customer’s MER file is located on the C: drive of your computer along with other lab files, you will restore this runtime application back to design time.

1. Click on Start > Programs > Rockwell Software > FactoryTalk View > Tools > Application Manager

The Application Manager is a tool that you can use to perform operations on FactoryTalk View Machine Edition (ME) and Site Edition (SE) applications.

Supported operations

- Rename: an ME application, an SE local application, or an SE network application
- Delete: an ME application, an SE local application, or an SE network application
- Copy: an ME application or an SE local application
- Back up: an ME application or an SE local application
- Restore: an ME application, an SE local application, an ME project archive created with the Project Transport Wizard, or a development application (.med) from a run-time application (.mer)
2. Select the *Restore runtime application* radio button and click *Next*.

![Image showing the Application Manager window with Restore runtime application selected and Next button highlighted.]

3. Click on *browse button* to the right of the *Specify the runtime application to restore* field.

![Image showing the Application Manager window with the Specify the runtime application to restore field and browse button highlighted.]

Options available:
- Restore the runtime application
- Restore the runtime application and FactoryTalk Local Directory

Specify the password if the runtime application is protected with a password:
4. Browse to and select the **C:\Lab Files > ViewME - Advanced > ViewME > Block_Machine_Advanced_Begin.mer** and click **Open**.

5. Select the **Restore the runtime application and FactoryTalk Local Directory** radio button.

For future reference: Use caution when you restore a FactoryTalk Directory, all users and groups defined on this computer will be replaced, as well as database definitions and security policies. This may affect other SE local or ME applications running on this computer. Be sure you have backed up any other such applications on this computer, and close any other programs that may access FactoryTalk Local Directory before continuing.
6. Click **Next**, you will receive the following popup:

![Application Manager popup with invalid password](image)

This means that the compiled application on the PanelView Plus is protected by a password; this allows the OEM to protect their intellectual property.

7. Click **OK** to close the dialog box.

8. As the OEM engineer you have access to the password, type in the password ‘Data’ (with capital D) in the password area and click **Next**.
9. Click **OK** to accept the following dialog.

You get this warning because you are going to convert the MER file back to design time and you chose to also restore the FactoryTalk Local Directory which contains all the users, groups and policies. The FactoryTalk Local Directory will be replaced with the one in the MER file.

10. Leave the default suggested name and click **Finish**.
The Application Manager is now converting the compiled application to an application that can be used by Studio.

Since you are also restoring the FactoryTalk Directory you will get a popup to enter User Name and Password for the Local FactoryTalk Directory currently existing on the computer.

11. Enter ‘labuser’ for the **User name**.

12. Enter ‘rockwell’ for the **Password**.

![Log On to FactoryTalk](image)

13. Click **OK**, the application will begin conversion and display a progress bar. The conversion may take a few minutes depending on the size of the application.

14. When the progress bar closes the conversion is complete.

You have converted the MER file back into a development application, including the users and passwords.
Opening the application in FactoryTalk View Studio

1. From the Start menu, select Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio or Programs > FactoryTalk View Studio as shown below.

2. If prompted select Machine Edition on the Application Type Selection screen.
3. Select *Block_Machine_Advanced*.

4. Make sure that *English (United States) en-US* is selected from the *Language*: drop down list.

5. Click *Open*.

Additional Information part D at the back of the manual provides an overview of Studio features.
Modifying the design time communications configuration

Restoring the runtime .MER file back to design time restores the runtime communications information however it does not restore the design time communications configuration information. The design time communications configuration information can vary from one development computer to another and is not stored within the MER file. You will now configure the design time communications configuration information so that you can browse for tags or test run the application within FactoryTalk View Studio.

1. Click the + to expand RSLinx Enterprise in the project explorer and double click Communication Setup.
2. Make sure the **Design (Local)** tab is selected.

3. Expand **EtherNet, EtherNet**.

4. Expand **192.168.1.1**.

5. Expand the **1789-A17/A Virtual Chassis**.

6. Highlight the controller in slot 6.

7. Click **Apply**.

8. Click **Yes** when prompted to save changes.
9. Click **OK** to save changes and close the **Device Shortcuts** dialog.

Note: If you do not click OK to close the Device Shortcuts dialog the changes you have made will not be saved.
Modifying the motor configuration

The end-user wants to take advantage of the new features available in FactoryTalk View Machine Edition version 6.0 to improve the appearance of a motor control used in the application. In this section you will modify a Global Object used in the application to include a new motor image from Symbol Factory and apply shading animation.

What are Global Objects?

Global Objects allow you to link the appearance and behavior of a graphic object to multiple copies of that object in the same application. This feature comes in handy when developing repetitive objects within the same application. When you make changes to the original objects, the copies are changed as well.

1. Open the Display 20 Extruder by double clicking on it.
The **Symbol Factory** graphics library, new in FTView ME v6 provides new sophisticated graphical objects. You are going to modify this Global Object and replace the motor image with one from the **Symbol Factory** library.

2. Right click on the motor control object on the screen and select *Edit Base Object*.

The Global Object screen **Screen Objects** will be opened and the Base Object highlighted.
3. Click on the **Objects** menu option and select **Symbol Factory**.

FactoryTalk View Studio for Machine Edition v6.0 includes Software Toolbox’s Symbol Factory library. This library contains over 5,000 images. Most of these images are vector graphics and can be used as you would any other image on a FactoryTalk Machine Edition display. The vector graphics can be animated using FactoryTalk View Studio for Machine Edition standard animation options (e.g. Fill, Color, Visibility). Approximately 1,000 images are bitmaps that you can use with buttons, multi-state indicators, etc. to create more visually appealing applications.

The Symbol Factory Viewer looks like this. There are three main areas to the viewer:

**Preview**
- A larger view of the selected image.

**Categories**
- Collections of related images

**Symbols**
- Displays the images available within the selected **Category**.
4. On the left side of the Symbol Factory screen you can see all the categories. Scroll down and select *Motors*.

5. Select *Motor 1*

6. Click *Copy* on the top right corner.

After you click *Copy* you will be brought back to the Global Objects screen called Screen Objects.

7. Right click somewhere in the background area and select *Object Explorer*. 

---

**Display Settings...**
**Key Assignments**
**Property Panel**
**Object Explorer**
**Paste**
**Show Grid**
**Snap On**
**Grid Settings...**
**Zoom In**
**Zoom Out**
**Cancel Zoom***
8. Expand the Grouped_Motor group and select Old_Motor. Right click on it and click Paste. You have now pasted the new Motor in this global object.

Your screen will look like this:

Object Explorer

The Object Explorer provides a tree-list naming all the objects in the selected display, and allows you to select and highlight objects from the list.

Groups are listed as expandable items, with the + icon. To view the objects in a group, click the + icon or double-click the group name. The list expands to show the objects and groups within a group.

You can use the Highlight Settings to highlight specified objects on both the graphic display and the Object Explorer tree list. You can select objects by object type, by animation type, or by tag name.

This helps with editing complex displays.
9. Right click on the Group16 that has been added and select *Property Panel*.

You are now going to modify the **Height**, **Width** and **GroupName** of the Motor.

1. Select the **Properties** tab of the Property Panel.

2. Modify the group properties to the values shown below. You do not need to set the **GroupLeft** and **GroupTop** values.
3. Click **Yes** when prompted that changing the name of the group will break links.

Instead of setting the animations (like color, visibility, fill,… ) again for the new motor you can copy animations. This can save you a lot of development time.
4. Right-click on the **Old_Motor** in the object explorer and select **Copy Animation**.

5. Select the **New_Motor** and right-click on it.

6. Select **Paste Animation**.
7. Now that the animations are copied you can delete the Old_Motor. Select the Old_Motor and right click on it and select **Delete**.
The end-user indicated that he doesn’t like that the motor is shown as solid gray and green. The new Symbol Factory motor graphic can use much better-looking gradient color shading.

In FactoryTalk View 5.1 and below, when you wanted to color animate an object shading was not used. In FactoryTalk View 6.0 you have three options: Solid (as in FTView ME 5.1), Shaded and Original.

8. Within **Object Explorer** right click on the **New_Motor** and select **Animation > Color**
9. For both the values 0 and 1 select Fill Style **Shaded** from the drop down list.

All the animations use a tag or tag expression to provide a value to the selected animation.

The animations supported by each object type vary:

* **Visibility** – show or hide the object.

* **Rotation** – move the object around a center by degrees

* **Width** – shrink or enlarge an object horizontally

* **Height** – shrink or enlarge an object vertically

* **Horizontal Position** – move an object on the display in the ‘x’ plane

* **Vertical Position** – move an object on the display in the ‘y’ plane

* **Horizontal Slider** – controls a tag value based
on object location within a set range based on the ‘x’ coordinate.

**Vertical Slider** – controls a tag value based on object location within a set range based on the ‘y’ coordinate.

**Fill** – floods an object based on a value proportional to the minimum and maximum value for the animation.

**Color** – change an object’s color based on a value.

The **OLE Verb** and **Touch** animations are only available to FactoryTalk View Site Edition applications.

---

**Color Animation Options**

The **Color** animation allows you to configure up to nineteen separate color animations. The animations are tied to the discrete values of the selected tag. For each separate value you can configure:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>Set the object’s line color</td>
</tr>
<tr>
<td>Fill</td>
<td>Set the object’s fill color</td>
</tr>
<tr>
<td><strong>Fill Style</strong></td>
<td>Select how the object will be filled. <strong>Fill Style</strong> options are</td>
</tr>
<tr>
<td><strong>Solid</strong></td>
<td>Flood the object with the selected Fill color</td>
</tr>
<tr>
<td><strong>Shaded</strong></td>
<td>Maintains the object shading when flooding with a selected Fill color</td>
</tr>
<tr>
<td><strong>Original</strong></td>
<td>Removes the Solid or Shaded animation from the object and displays it in its original color and shading.</td>
</tr>
</tbody>
</table>
10. Now click *Apply* and *Close*.

11. Drag and drop the **New_Motor** so that it sits in the same place as the original motor as shown below:

![Motor Image]

12. Close and save the **Screen Objects** Global Objects display.

13. Close the **20 Extruder** display and **Symbol Factory**.

The changes to the Global Objects will take affect the next time the **20 Extruder** display is opened. Open the display if you want to see the new graphic and close it when you are finished.

You just finished modifying a Global Object on the screen. First you changed the image of the motor using a new image from the Symbol Factory library and then you modified the animation using shading.
Modifying the historical data logging

In this section you will modify the DataStore Plus ActiveX control to add one additional tag to log data for.

The datalogging that is used here is an ActiveX control that is open all the time but is not visible because the pop-up display containing the ActiveX is positioned off the screen.

If you want to know more on how this is done, have a look at the Additional Information part A at the back of the manual.

FactoryTalk View Machine Edition has two data logging options, the DataStore Plus ActiveX control that you are going to use in this lab and the built-in Datalog model.

NOTE: The most significant differences between the DataStore Plus ActiveX control and the built-in Datalog model is that the DataStore ActiveX cannot be used to provide historical data to the built-in Trend object in FTView ME.

The DataStore Plus ActiveX control is very similar to the RSView32 and FTView SE data log model and contains the following advanced features:

- File management – users can specify when new files are created and old files are deleted
- Customization of file headers - allows users to create their own header names for data in the CSV or TXT files.
- Allows periodic logging based on the time interval specified by users
- Each record is automatically date and time stamped with UTC and/or System Date and Time information
- Overwriting oldest data when the file reaches the size specified by users
- Optimized logging - users can choose to log only assigned connections

More information about the differences in data log models can be found in Additional Information part B at the back of the manual.
DataStore Plus is one of the ActiveX controls that are included with FactoryTalk View Machine Edition. The list below shows the original 8 ActiveX controls that will be installed with FactoryTalk View Studio. These ActiveX controls are also registered on the PanelView Plus 6 terminal by default.

<table>
<thead>
<tr>
<th>ActiveX Control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program Launcher</td>
<td>Launch another program on the run-time platform from inside of an ME project</td>
</tr>
<tr>
<td>DataStorePlus</td>
<td>Log tag data to a CSV file. It contains advanced features such as file management, periodic logging, and optimized logging</td>
</tr>
<tr>
<td>Recipe</td>
<td>Save, load, and delete recipes from a .CSV file</td>
</tr>
<tr>
<td>ME_IPInfo</td>
<td>Read TCP/IP configuration information from the PanelView Plus terminal to display in HMI application</td>
</tr>
<tr>
<td>MacroTen</td>
<td>Allow placeholders to be used in macros</td>
</tr>
<tr>
<td>Set Bit Cursor Point</td>
<td>Emulate the functionality of the Set Bit Cursor Point object found in the PanelView “e” terminals</td>
</tr>
<tr>
<td>Custom String Popup Keyboard</td>
<td>Allow users to use a custom ‘Pop-up Keyboard’ in an ME application using a user-defined file that provides key assignments of potentially any languages, i.e. Chinese, Korean, French, etc.</td>
</tr>
</tbody>
</table>
| DisplayP           | Open and close On Top and Replaced displays and use parameters.  
  *FactoryTalk View Machine Edition v5.1 and above provides this capability as built-in functionality.* |

Because the end-user wants to be able to copy the data from the terminal while it’s still running, the OEM chose the Datastore Plus ActiveX because it allows you to start and stop the logging so that the data can be copied. The built-in Data Log functionality cannot be stopped and restarted or accessed at runtime.
1. Open the **99 Data Logging** graphic display.

![99 Data Logging graphic display](image)

2. To add the new Temperature to the datalog, double click on the **MEDataStorePlus1** object.

3. Select the **Connections** tab.

   NOTE: The `{Maintenance\Datalog_trigger}` tag assigned to the Trigger will be used to manually start and stop the datalogging. The CSV files will be written to **My Documents** on the terminal as configured by the FileLocation connection.
4. Click the **browse button** next to the **Float06** field, the Tag browser will open.
5. Drill down into the process as shown in the image below to add the 
{::{BlockComms}Program:Zone6_Temperature.Temperature_Zone6.CV} tag to the DataStore Plus ActiveX.

6. Press OK when finished and OK to close the properties dialog.

7. Close and save the 99 Data Logging graphic display.
The following further defines and highlights the uses and functionality provided in the Tag Browser.

**Folder List**

<table>
<thead>
<tr>
<th>Folder</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>L15</td>
<td>Diagnostic Items, Online, system</td>
</tr>
</tbody>
</table>

**HMI Server** (e.g. L15)
Contains tags defined in the HMI Server; referred to as *memory tags*. The FactoryTalk Machine Edition runtime manages and updates these tags.

**System**
Contains default HMI tags for system information, such as, time of day, date, etc.

**<Communication Shortcut> (e.g. L15)**
Contains tags that exist in the memory of the device the shortcut represents.

**Diagnostic Items**
Predefined diagnostic and troubleshooting tags useful for monitoring communication and controller status.

**Online**
Tags contained in a Logix Controller’s memory; the tags are directly referenced by the FactoryTalk Machine Edition runtime.

**Tag Area**

<table>
<thead>
<tr>
<th>Name</th>
<th>Access Rights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmd_StopOp</td>
<td>ReadWrite</td>
</tr>
<tr>
<td>Cmd_StopProg</td>
<td>ReadWrite</td>
</tr>
<tr>
<td>EnableIn</td>
<td>ReadOnly</td>
</tr>
<tr>
<td>EnableOut</td>
<td>ReadOnly</td>
</tr>
<tr>
<td>Trip_Scaled_Speed_M1...</td>
<td>ReadWrite</td>
</tr>
<tr>
<td>Set_AccelOp</td>
<td>ReadWrite</td>
</tr>
<tr>
<td>Set_AccelProg</td>
<td>ReadWrite</td>
</tr>
<tr>
<td>End_Coop_EvalTimeReady</td>
<td>ReadWrite</td>
</tr>
</tbody>
</table>

**Name**
Column containing the tag’s name. Use the **Tag filter** function to quickly find tags that start with the user-entered filter.

**Access Rights**
New! In V18 firmware, Logix Controllers implements Access Rights for tags. This column displays a tag’s Access Rights property. Tags are either *ReadOnly* or *ReadWrite*.

You have successfully modified the DataStore Plus Activex control to log data for one additional tag.
**Adding machine troubleshooting information**

The end-user has asked you if it’s possible to show information on the PanelView Plus about the PowerFlex drive, this information must be secured and only available to the Shift Manager.

You know that there is a PDF document available that would help the customer a lot. With this new PanelView Plus 6 terminal you can easily show PDF files on the screen.

To trigger the Program Launcher ActiveX that will launch the PDF file we need to select a tag for this purpose.

1. Under the **HMI Tags** folder double-click **Tags**.

2. Once the HMI database is open, double click on **Maintenance**.
As you can see in the HMI database there are already memory tags created that can be used, for example the **Maintenance.pdf1**.

3. Close the **Tag Database** editor.
HMI database

In the HMI tag database, you can define data you want FactoryTalk View to monitor. There is one HMI tag database per HMI server. Each entry in the database is called an HMI tag. An HMI tag is a logical name for a variable in a device or in local memory (RAM) that is referenced in the HMI tag database.

All parts of the system use tag values. Graphics uses tag values to control the animation in a display or update a trend graph; Alarming monitors HMI tag values and compares them to "acceptable" limits; Data Log stores tag values to create a historical record.

HMI tags are required when you need extra information included with the tag value. This extra information can be a minimum/maximum value range or value units. If you do not need this information, you can use direct references to tags in other data servers.

HMI tag types

You can create three types of HMI tags, based on the data that they can store:

Analog tags - store a range of values. These tags can represent variable states such as temperature or the position of rotary controls.

Digital tags - store two states. Zero will be written as False and any non-zero number will be written as True. Use digital tags to represent devices that can only be on or off, such as switches, contacts and relays.

String tags - store an ASCII string, series of characters, or whole words (maximum 82 characters). These tags can represent devices that use text, such as a bar code scanner which uses an alphanumeric product code.

HMI tags you do not create

A set of system tags is created automatically when you create an HMI server project, and stored in a folder called 'system' in the tag database.

System tags - store information generated while the system is running, including alarm reset time and date, system time and date, and current user information. There are analog, digital, and string system tags. You can use them anywhere you use other types of tags, but you cannot edit system tags.

Data sources

When you create an HMI tag you must specify what its data source will be at run time. Data source can be:

Device - receives its data from a programmable controller through a direct driver or an OPC server. It can also receive data from another Windows program through an OPC server.

Memory - data comes only from the value table rather than from a programmable controller or another program.

System - the system tag is created by the system and stored in a folder called System.
Additional Information part C at the back of the manual explains how you could use the **Application Documenter** to determine if the Maintenance\pdf is in use anywhere else in the application. For purposes of the lab you know it is safe to use the tag.

4. Go to **Displays** and open screen **70 Maintenance**.

5. Click on the **Objects** menu and select **ActiveX Control**.

6. Draw a square in the right bottom corner of the **70 Maintenance** display.
7. In the list of ActiveX controls, select **ME Program Launcher** and click **OK**.

8. Click **OK** when you see the warning.

ActiveX controls are created using programming software. The ActiveX needs to be compiled to be able to run on pc or on PanelView Plus 6. This compilation is done based on Operating system and CPU. So the ActiveX used on the desktop is compiled differently than the one compiled for use on the PanelView Plus 6.
9. Double click on the gray button that was just created on the screen. Go to the Common tab and uncheck all the check boxes in the Other section.
10. Select the **Connections** tab and fill in the fields exactly as show below using the tag browser and entering a '0' for the **NewInstance** field.

11. Click **OK** when finished.

![Connections Tab](image)

The PDF document has already been installed on the PanelView Plus 6.

- The PDF viewer that is installed on the PanelView Plus 6 is called FoxIt Reader.

**Note:** The HMI Tag **Maintenance\PDFLocation** contains the FoxIt Reader command line “"My Documents\Manuals\Powerflex.pdf"” -g -p 23 –zw

- **-g** is to prevent operator from browsing to other pdf files. This is done for security reasons.
- **-p 23** Opens the PDF file to page 23.
- **-zw** Opens the PDF file and fits the view to the page width.
The following connections are set at run-time to control the operation of the Program Launcher ActiveX control

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProgramLocation</td>
<td>This connection contains the complete path and filename of the application program you wish to execute (ex. &quot;ctlpnl.exe&quot;) using the control. This property can be an explicit string or a string tag.</td>
</tr>
<tr>
<td>ProgramParameter</td>
<td>This connection contains optional command-line parameters to be used with the application configured in the ProgramLocation connection. This property can be an explicit string or a string tag.</td>
</tr>
<tr>
<td>FocusEnable</td>
<td>While another application is running, the FactoryTalk View Machine Edition project may not be visible if it is in the background or minimized. When the assigned tag transitions from a zero to non-zero value, the FactoryTalk View Machine Edition project is restored to the foreground.</td>
</tr>
<tr>
<td>LaunchEnable</td>
<td>When the assigned tag transitions from a zero to non-zero value, the control launches the application configured in the ProgramLocation connection appending the optional command-line parameter(s) from the ProgramParameter connection. The application is launched maximized and given focus. This connection must be a tag.</td>
</tr>
<tr>
<td>NewInstance</td>
<td>This connection configures how the control will act if the application configured in the ProgramLocation connection is already executing when the FactoryTalk View Machine Edition application attempts to launch another instance. If the configured tag is set to zero, the control will restore the application configured in the ProgramLocation connection to the foreground. If the configured tag is set to one, the control will attempt to launch an additional instance.</td>
</tr>
</tbody>
</table>
MaximizeEnable
Optional Connection
If the application specified in ProgramLocation becomes hidden behind the FactoryTalk View Machine Edition project or minimized this property allows the application to be maximized. To restore a minimized application set the tag configured in MaximizeEnable to a non-zero value and the tag configured in NewInstance to zero, before triggering the LaunchEnable tag.

CloseProgram
Optional Connection
When the assigned tag transitions from a zero to non-zero value, the CloseProgram property will close the application configured in the ProgramLocation connection.

If multiple ActiveX controls are used to launch multiple application, only the control that launched a particular instance of an application can be used to close that application.

Now you are going to create a button that will toggle the Maintenance\PDF1 tag.

1. In the Objects > Push Button menu, select a Momentary push button. Draw a box at the bottom of the screen.
2. Double click on the newly created *Momentary Push Button* and change the properties of the *General* tab as shown below.
3. Select the **States** tab, **State0** will be selected.

4. Click the browser next to the **Image** field and select the **pdf_icon** image.

![Image modification process]

5. After making the Image modification for State0 click the **Copy** button. This allows you to copy settings from one configured state to another. You will be copying the configuration of **State0** to **State1**.
6. Here you can select which settings you would like to copy to State1. Leave all the defaults and click OK.

7. The Paste button is now available. Select State1 and click Paste.

8. You can Paste to the currently selected State or to All States. Click on OK to paste to the Current State which is State1.
9. Go to the **Common** tab and change the **Height** and **Width** as shown below.

![Common tab with Height and Width fields highlighted](image)

10. On the **Connections** tab add the tag *Maintenance*pdf1 for the **Value** connection. Since you have used the tag browser before it shouldn’t be a problem anymore to find that tag.

11. Click **OK** when finished.

![Connections tab with Maintenancepdf1 tag selected](image)
The result of your actions is shown below.

12. Close and Save the **70 Maintenance** display.

You have modified the DataStore Plus ActiveX and you have added a way to launch a pdf file from your PanelView Plus 6 terminal.

Congratulations! You have finished scenario 1. Let’s have a look at the application running on the PanelView Plus 6 terminal.
Running the application on the PanelView Plus 6 Terminal

Now you will test the entire application on your PanelView Plus terminal. To do this, you have to build a run time application (MER file) and transfer it to the PanelView.

Building the runtime .MER file

1. Select *Create Runtime Application* on the *Application* menu:

2. On the *Create Runtime Application* dialog, verify that the runtime file name is ‘Block_Machine_Advanced.mer’. The file is saved to the default Runtime file location, *C:\Users\Public\Public Documents\RSView Enterprise\ME\Runtime*.

3. Click *Save*. 
The *Conversion to development application* options on the Create Runtime Application dialog allow later recovery of the design files from the runtime project using the Application Manager, if allowed. Options are:

**Always allow conversion [Default]**
The design information is always included with the runtime, so that it may be restored from the .MER. The resulting .MER requires more terminal memory to store the file.

**Never allow conversion**
Design information cannot be recovered from an MER created with this option selected. The .MER created requires the least amount of terminal memory.

**Conversion protected by password**
When using Application Manager to extract the design information from the runtime file, the user will be prompted for the configured password. The resulting .MER requires more terminal memory to store the file.

The *FactoryTalk ViewPoint version* option allows the user to specify whether or not published displays will be included in the MER file.

**Include ViewPoint Content**
Must be selected for the application to include the ViewPoint published displays.
4. On the **Select Languages** dialog, make sure that **English** is selected as the initial runtime application language and click **Finish**.

A progress dialog will be displayed while the runtime file is being created.

Note: This application contains multiple languages however this lab does not use the Language Switching feature so we are only selecting English.
Download the runtime .MER file to a PanelView Plus 6 terminal

1. To download the runtime .MER to the PanelView Plus terminal at your workstation, select Transfer Utility on the Tools menu:

![Transfer Utility](image)

Note that you can also click the Transfer Utility icon on the toolbar.

The Transfer Utility is opened:

![Transfer Utility Window](image)
2. Click the **Source File**: browse button to select the runtime .MER file to download. Select the **Block_Machine_Advanced.mer** file you previously created and click **Open**.

3. Double-click the **EtherNet, Ethernet** driver to expand the item.
4. Select the PanelView Plus found at IP address **192.168.1.20** by clicking on it once. (It may look different from the screen shot below.)

5. Click **Download** to transfer the runtime file to your PanelView Plus terminal.

If you are asked if you want to overwrite the existing file, click **Yes**.

6. You will see a progress bar, and when the download process completes a confirmation dialog is shown. Click **OK** to acknowledge the dialog.

7. Click the **Exit** button to close the **Transfer Utility**.
Run the application on a PanelView Plus 6 terminal

In this section, you will work with a PanelView Plus terminal. Follow the steps below on your terminal to run the FactoryTalk View ME runtime application you just created.

1. If FactoryTalk View ME Station is not already running on the terminal, double-tap the **FTView ME Station** icon on the desktop.

2. Press the **Load Application [F1]** button.

3. Select the **Block_Machine_Advanced.mer** file from those available from the terminal’s Internal Storage.
4. Press **Load [F2]** to load the runtime file into memory.

5. When you are prompted, press **Yes [F7]** to overwrite the terminal’s current communication configuration with the configuration contained within the Block_Machine_Advanced.mer file.

6. Once successfully loaded, press the **Run Application [F2]** to start executing the runtime file.

While the terminal is starting the application, an update dialog is displayed:

![Run Application [F2]](image)

After the start-up processing completes, you should see your application’s startup display.

![Application Startup Display](image)

Note that the Default user is logged in and there are limited navigation icons across the top of the display.

Your screen will look similar to the one above depending on values in the controller.
Testing the motor changes

1. Tap on the **block** icon in the top banner.

2. Tap the **Extruder** menu item to open the **Extruder** display.

3. Tap the **Start Pelletier** Maintained Push Button and note that the motor you created animates depending on the state of the Pelletier. Your display may show different values than the ones shown below.
Viewing a PDF file on the terminal

1. You will need to login as **Supervisor** in order to be able to navigate to the **Maintenance** display and start the Foxit Reader.

2. In the top banner tap the icon top open the login dialog box.
3. Use the on-screen keypad to enter ‘Engineer’ as User and ‘1234’ as Password.

4. Tap the enter key when done.

![Login:](image)

5. In the top banner you will now see that the icon appears when you tap on that icon you will go to the Maintenance screen where you can see the PDF icon.

![Maintenance screen](image)

6. On the Maintenance screen tap on the PDF icon. This will open the new Foxit Reader showing the PowerFlex document.
Single-Phase Input Power

The PowerFlex 70 drive is typically used with a three-phase input supply. Single-phase operation of the drive is not currently rated under the UL 508C listing. Rockwell Automation has verified that single-phase operation with output current denoted by 50% of the three-phase ratings identified in the tables in the Power Ratings and Branch Circuit Protection below.

Power Ratings and Branch Circuit Protection

208 Volt AC Input Protection Devices (See page 24 for Notes)

<table>
<thead>
<tr>
<th>Drive Catalog Number</th>
<th>HP Rating</th>
<th>Input Ratings</th>
<th>Output Amps</th>
<th>Dual Element Time Delay Fuse</th>
<th>Non-Time Delay Fuse</th>
<th>Circuit Breaker&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Motor Circuit Protector&lt;sup&gt;b&lt;/sup&gt;</th>
<th>100M Motor Protector with Adjustable Current Range&lt;sup&gt;c&lt;/sup&gt; (CNR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20AV572</td>
<td>0.5</td>
<td>0.5</td>
<td>1.1</td>
<td>2.2</td>
<td>3.5</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>20AV573</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV574</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV575</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV576</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

240 Volt AC Input Protection Devices (See page 24 for Notes)

<table>
<thead>
<tr>
<th>Drive Catalog Number</th>
<th>HP Rating</th>
<th>Input Ratings</th>
<th>Output Amps</th>
<th>Dual Element Time Delay Fuse</th>
<th>Non-Time Delay Fuse</th>
<th>Circuit Breaker&lt;sup&gt;a&lt;/sup&gt;</th>
<th>Motor Circuit Protector&lt;sup&gt;b&lt;/sup&gt;</th>
<th>100M Motor Protector with Adjustable Current Range&lt;sup&gt;c&lt;/sup&gt; (CNR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20AV572</td>
<td>0.5</td>
<td>0.5</td>
<td>1.1</td>
<td>2.2</td>
<td>3.5</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>20AV573</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV574</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV575</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20AV576</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>1.2</td>
<td>1.8</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

70 of 294
7. With the Foxit reader, if you double tap on the PDF file you can zoom in and out.

On the right bottom corner you see a page overview, the white area is what you currently are looking at. Moving your finger across the page there will show different areas of the page or you can move the zoomed in page itself.

8. When you are zoomed out and you slide with your finger from right to left on the screen you can change pages forward, if you slide left to right, you go backwards.

9. Tap the X in the upper right corner to close the Foxit Reader.
Examining the data logs using the FTP Server

Turn to your desktop computer to start this next section. You will be connecting to the PanelView Plus terminal using Internet Explorer to explore how data log files can be managed over a network using the FTP Server. The FTP Server can be used to transfer files over the network to and from your terminal when viewed within Windows Explorer.

Note that you can use the Network Servers option from the Control Panel on the PanelView Plus 6 terminal to see which servers are running and start/stop them.

In the example below the FTP Server and the ViewPoint Server buttons are red showing that they are stopped, all other server buttons are green showing that they are running.
Note that you can use the **Server Config** option from the Control Panel on the PanelView Plus 6 terminal to configure the properties of the VNC, WEB, FTP and File services.
1. If Internet Explorer is not already open from a previous section launch it and enter the following address: 
   \texttt{ftp://192.168.1.20} where 192.168.1.20 is the IP address of your PanelView Plus 6 terminal and you are using the \texttt{ftp://} syntax.

You will see the following

2. Select \textit{Page} and then \textit{Open FTP Site in Windows Explorer}.
3. Click **OK** when the **FTP Folder Error** pop-up appears.

![FTP Folder Error](image)

4. Right click in the open space to the right and select **Login As...**

![Login As...](image)

5. Enter the following for the **User name:** and **Password:** and click **Log On.**
   
   ‘user1’
   
   ‘Pass1’

![Log On As](image)
Your window is now updated to show you the files from the PanelView Plus 6 terminal.

6. Double click on the **My Documents** directory.

Selecting the **Detail** view, the date and time stamp in the Block_log.csv file name indicate that there is current data log activity, this is because the DataStore Plus Trigger is set to run by default when this application starts. Your files will have different date stamps than the ones shown above.
The Datastore Plus ActiveX CSV file naming format:
You will see files with filename YYYY.MM_DD_HR_MIN_NNNN.<filename>.csv".
Example: When the property "File Location" is set to "\My Documents\Block_log.csv",
then ActiveX control will create new file named as "\My Documents\2010_12_15_02_02_0001_Block_log.csv ".
YYYY is the year, based on the System year in the systems local timezone.
MM is the month, based on the System Month in the systems local timezone.
DD is the day, based on the System Day in the systems local timezone.
HR is the hour in 24-hour clock format, based on the systems local timezone.
MIN is the minute, based on the System Minute in the systems local timezone.
NNNN is the sequential file identifier. This number indicates the sequence the files were created in, beginning with 0000. The application may use up to 9999 file sets per day. At midnight, the sequence starts at 0000 again.
You can copy the file to your desktop and open it. Alg1 to 6 are the 6 temperature tags that are defined in the DataStore Plus ActiveX. Alg6 is the tag you previously added at the request of the end-user. This ActiveX can also log the date and time in UTC time which, does not use summer/winter times. This option can be set on the DataStore Plus ActiveX properties dialog.

Note that if you refresh Internet Explorer you will see that the file size is increasing for the most current datalog file. Your list of files will look different than these screen captures.
Starting and stopping the DataStore Plus ActiveX

1. Turn to your PanelView Plus 6 Terminal.

2. If you are currently still logged in as Engineer you will now see that the icon in the top banner, when you click on that icon you will go to the Maintenance screen where you can see that the datalogging is running.

3. Click the toggle switch to turn off datalogging.

4. Turn back to Internet Explorer and use the refresh button and note that the file size for the most current datalog has stopped increasing, it is now be safe to copy that file to your computer. You cannot copy CSV files while the DataStore Plus is logging data.
5. Turn back to your PanelView Plus 6 Terminal and click the **toggle** switch again to turn on datalogging.

![Datalog Switch](image)

6. Return to your Internet Explorer window on your desktop computer.

7. Use the refresh button to see that the data log has started updating.

8. Shut down the application on your PanelView Plus by tapping on the *Rockwell Automation* logo located in the upper right corner of the display.

![Rockwell Automation Logo](image)

Congratulations! You have completed the modifications that the end-user requested.
Scenario 2: Using the ME_IPInfo ActiveX control to auto start an initial graphic display – 20 min.

About This Scenario

You are an engineer who wants to create and maintain a single Machine Edition application for the entire plant. You want the application to have a unique sub-set of displays for each part of the plant and you want the application to start with one of these sub-sets depending on where the PanelView Plus 6 terminal is located in the plant.

To do this you will modify the existing Block_Machine_Advanced application to make use of the ME_IPInfo ActiveX control along with Macros and Global Connections.

The overall premise for this lab is that the application will contain subsets of Graphic Displays with “Display Numbers” specific to a particular terminal in the application such as “100, 101, 102…200, 201, 202, 203……300, 301, 302…….” When the application starts the ME_IP_info ActiveX control located on the 98 Start up graphic display retrieves the last octet of the IP address for the terminal and places that number into an HMI memory tag. In this lab the tag is called “Ipocet4”. Global Connections have been configured to monitor for the IP address to be populated, when this occurs “Macro1” is called. The expression contained in “Macro1” determines which intial graphic display to call up based on the IP address such as graphic display 100, 200, 300…. Once the initial graphic display is called up based on the IP address of the terminal the operator never sees any other screens in the application except those accessible from the initial graphic assigned to that IP address.

Open the application

1. Open the Block_Machine_Advanced application in Studio if it is not already open.
Examine the graphic display containing the ME_IPInfo ActiveX control

In this section you will examine how the ME_IPInfo ActiveX control has been configured and how the graphic display has been modified to provide the functionality the end-user wants.

2. Double click the 98 Startup graphic display.
3. Double click the **ME_IPInfo** (the grey box) ActiveX control in the upper left corner of the display.

4. Select the **Connections** tab, note that 4 HMI memory tags have been assigned to the four octets of the IP address. This scenario will use the **IPoctet4** tag to determine which PanelView Plus 6 Terminal is running the application.

   You do not need to make any changes here.

5. Click **OK** to close the properties dialog box.
6. Right click on the 98 Startup display and select **Display Settings**.

On the **General** tab note that the **Display Number** is **98**, this is the number we will use in Global Connections to launch this display.
7. Select the **Behavior** tab and note that the macro **ClearRemoteDisplayNumber** has been assigned as a **Shutdown** macro for the 98 **Startup** display. This macro will run when the display closes.

The **ClearRemoteDisplayNumber** macro sets the **RemoteDisplayNumber** tag assigned to the Global Connections **Remote Display Number** to 0 to allow the operator control of screen navigation. The macro also sets the **InitialConfigSet** tag to 1 to flag that the initial graphic display has been selected.

8. Click **OK** to close the **Display Settings** dialog box.
9. Click the X in the upper left corner of the 98 Startup display to close it.

![Shutdown button]

You now understand some of what is required to meet the needs of the end-user.
**Change which graphic display is called at startup**

Additional Information part A at the back of the manual explains how this application opens a display called ***99 Data Logging*** in order to load the DataStore Plus and email ActiveX controls and keep them loaded the whole time that the application is running. The ***99 Data Logging*** Graphic Display has a macro assigned to it called ***99Startup*** that sets the tag assigned to the Global Connections Remote Display Number connection to 1, which is the ***1 Main*** Display. In this section you will change which Graphic Display is opened by the startup macro after the ***99 Data Logging*** display opens. Doing this will allow the Graphic Display containing the ***ME_IPInfo*** ActiveX to be loaded.

1. Open the ***99Startup*** macro.
2. Change the number in the **Expression** field from 1 to ‘98’ as shown below.

![Table showing expression changes](image)

Change to:

![Table showing expression changes](image)

The macro will now write a 98 to the **RemoteDisplayNumber** tag assigned to Global Connections. The **98 Startup** Graphic Display will load and the ME_IPInfo ActiveX will write the last octet of the PanelView Plus terminals IP address to the tag **IPoctet4**.

3. Click **Close**.

4. Click **Yes** when prompted to save the changes to the **99StartUp** macro.

You have modified the 99Startup display to call up the display containing the ME_IPInfo ActiveX control.
Examine the Global Connections

Global connections provide run-time functions that apply to your entire application. They run as a background activity and read or write to tags assigned to various connections. You will use Global Connections to monitor for changes in tags and to perform tasks that allow the application to automatically call up an Initial Graphic Display depending on the last octet of the terminals IP address.

1. Double click on Global Connections to open the Global Connections editor.
2. Select the **Display** tab and note that the tag **RemoteDisplayNumber** has been assigned to the **Remote Display Number** connection. This connection allows for remote control of graphic displays.

You do not need to make any changes here.

3. Select the **Macro** tab.

4. Double click the browse button under the **Exprn** column and to the right of the **Remote Macro1** field.
5. Enter the expression exactly as shown below:

\[ \text{If (InitialConfigSet == 0) AND (IPoctet4 <> 0) Then 1 Else 0} \]

6. Click the **Check Syntax** button to validate your entry.

7. Click **OK** when done with the expression.

   If you are prompted to create a tag that does not exist then go back and check the syntax to ensure that everything is correct.

8. Click **OK** again to close the Global Connections editor.
When the expression assigned to Global Connections **Remote Macro1** becomes true the **Macro1** macro runs. We will now examine this macro.

9. Double click on **Macro1**.

Macro1 contains an expression that will call a particular Graphic Display depending on the value of the **IPOctet4** tag. From the expression you can see that if IPOctet4 equals 20 which is the last octet of the hard coded IP address assigned to the PanelView Plus terminal then graphic display 35 will be called.

The Else 1000 statement at the end of the expression is a catch all that will call graphic display 1000 if no valid octet is found.

Also note that **Macro1** sets the **InitialConfigSet** tag to 1 showing that the initial display selection has completed.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 RemoreDisplayNumber</td>
<td>If IPOctet4 == 1 Then 1 Else If IPOctet4 == 20 Then 35 Else If IPOctet4 == 3 Then 300 Else 1000</td>
</tr>
<tr>
<td>2 InitialConfigSet</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

You do not need to make any changes here.
10. Click Close to close the Macro1 editor.

You have made modifications in Global Connections in order to complete the functionality the end-user is looking for.

**Running the application on the PanelView Plus 6 Terminal**

Now you will test the entire application on your PanelView Plus 6 terminal. To do this, you have to build a run time application (MER file) and transfer it to the PanelView.

**Building the runtime .MER file**

1. Select **Create Runtime Application** on the Application menu:

2. On the **Create Runtime Application** dialog, verify that the runtime file name is ‘Block_Machine_Advanced.mer’.

The file is saved to the default Runtime file location, C:\Users\Public\Public Documents\RSView Enterprise\ME\Runtime.
3. Click **Save**.

4. Click **Yes** to replacing existing if prompted.
5. On the **Select Languages** dialog, make sure that **English** is selected as the initial runtime application language and click **Finish**.

A progress dialog will be displayed while the runtime file is being created.

Note: This application contains multiple languages however this lab does not use the Language Switching feature so we are only selecting English.
Download the runtime .MER file to a PanelView Plus 6 terminal

1. To download the runtime .MER to the PanelView Plus terminal at your workstation, select **Transfer Utility** on the **Tools** menu:

![Transfer Utility](image)

Note that you can also click the Transfer Utility icon on the toolbar.

The **Transfer Utility** is opened:

![Transfer Utility](image)
2. Click the Source File browse button to select the runtime .MER file to download. Select the Block_Machine_Advanced.mer file you previously created and click Open. Note that the MER file may already be selected for you if you have previously downloaded the same application.

3. Double-click the EtherNet, Ethernet driver to expand the item.
4. Select the PanelView Plus found at IP address 192.168.1.20 by clicking on it once. (It may look different from the screen shot below.)

5. Click Download to transfer the runtime file to your PanelView Plus terminal.

If you are asked if you want to overwrite the existing file, click Yes.

6. You will see a progress bar, and when the download process completes a confirmation dialog is shown. Click OK to acknowledge the dialog.

7. Click the Exit button to close the Transfer Utility.
Run the application on a PanelView Plus 6 terminal

In this section, you will work with a PanelView Plus terminal. Follow the steps below on your terminal to run the FactoryTalk View ME runtime application you just created.

1. If FactoryTalk View ME Station is not already running on the terminal, double-tap the **FTView ME Station** icon on the desktop.

2. Press the **Load Application [F1]** button.

3. Select the **Block_Machine_Advanced.mer** file from those available from the terminal’s Internal Storage.
4. Press **Load [F2]** to load the runtime file into memory.

5. When you are prompted, press **Yes [F7]** to overwrite the terminal’s current communication configuration with the configuration contained within the Block_Machine_Advanced.mer file.

6. Once successfully loaded, press the **Run Application [F2]** to start executing the runtime file.

7. While the terminal is starting the application, an update dialog is displayed.

8. After the start-up processing completes, you should see your application’s startup display.

Remember that **Macro1** is run when the tag **IPoctet4** is not equal to 0 (<>0). Below you can see that Macro1 will set the **RemoteDisplayNumber** tag to **35** if the **IPoctet4** tag = **20** which is the last octet of the hard coded IP address assigned to the PanelView Plus 6 terminal used in this lab. Setting the **RemoteDisplayNumber** tag to 35 will launch the Graphic Display **35 Vacuum Silos Root Menu**.
9. Tap on any of the silos shown on **35 Vacuum Silos Root Menu**.

10. The display **36 Vacuum Silos From Root Menu** will be opened.

Depending on how you have designed your application you will only be able to navigate to the screens usable by the operator at this particular PanelView Plus Terminal even though all terminals in the plant are running the exact same application. What you are looking at is a sub-set of the screens contained in the application. We have designed them to only have access to a section of the plant unique to the terminal at this IP address, there is no way to navigate back to the Main display and see other parts of the plant.
11. Click on the house icon in the upper left corner to return you to the **35 Vacuum Silos Root Menu** display.
12. On the **35 Vacuum Silos Root Menu** display click **Shutdown**.

![Shutdown button](image-url)

Congratulations! You have successfully configured the application to start with a unique Graphic Display depending on which IP address the terminal is using.

About This Scenario

The end-user has asked that you provide a way to allow remote access to an application running on the PanelView Plus 6 terminals without interrupting the operators or the terminals performance. The end-user has requested that two unique displays be created just for FactoryTalk ViewPoint access. In this case the displays have the same screen resolution as the rest of the application however the screens could have a resolution that is unique to a handheld device that is using a compatible web browser.

You have also been asked to provide security that will prevent operators from accessing the ViewPoint Client but allow the Plant Manager to view production data and allow the Shift Managers to start and stop the Pelletier while away from the plant.

You will enable the ViewPoint in the application and provide the following access:

Operators – no access to ViewPoint

Plant Managers – View-only access

Shift Managers – Both View and Write access

FactoryTalk ViewPoint is an add-on to FactoryTalk View ME running on PanelView Plus that provides for a fully scalable, fully animated, read-only view of existing applications from a Web browser.

To make information about your plant or process available on demand from a Web browser in your office, home, or hotel, all you have to do is select the FactoryTalk View graphic displays you want to make ready for the Web, and then publish the displays to the FactoryTalk ViewPoint Server which runs on a PanelView Plus.

There is no installation of any Rockwell Software products on the browser computer all you need to connect to a published FactoryTalk ViewPoint Web application is the name (or IP address) of the PanelView Plus hosting the FactoryTalk ViewPoint Server that stores the application.

Once you enter a simple address directly into your Web browser using the name or IP address, the browser will connect to the published web application and open the initial display selected. Use navigation buttons in the application to view other published displays, or use the web browser's navigation tools.

Setting up a FactoryTalk ViewPoint ME application

In this section, you will follow step-by-step instructions to complete the publishing process. This process involves taking an existing FactoryTalk View ME application and creating a Web application using the FactoryTalk ViewPoint Administration tool.
You will learn how to:

- Publish displays from an existing FactoryTalk View ME application
- Configure FactoryTalk ViewPoint ME Security
- Transfer the published application to the PanelView Plus terminal

**FactoryTalk ViewPoint Administration**

FactoryTalk ViewPoint’s configuration is done through the FactoryTalk ViewPoint Administration tool, launched from FactoryTalk View Studio. Here, the desired displays are selected and published so that they will be available to any browser client connecting to your operator interface terminal.

With FactoryTalk ViewPoint ME, the Administration tool runs on the same computer as FactoryTalk View Studio and is installed using the FactoryTalk ViewPoint ME install available on the ViewPoint CD. The FactoryTalk ViewPoint ME Administration tool is separate from the ViewPoint ME server that runs on PanelView Plus because the components used to convert and publish ViewPoint displays are not supported on the PanelView Plus and are too large to store on the terminal.

For the purpose of this lab, FactoryTalk ViewPoint ME is already installed. The install is available from the FactoryTalk ViewPoint CD or as a stand-alone install. See Answer ID 66110 on the Rockwell Automation knowledgebase for more details.

1. From the **Start** menu, select **Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio** or **Programs > FactoryTalk View Studio** as shown below.
2. Open the existing project called *Block_Machine_Advanced*.

3. Select *ViewPoint Administration* from the *Application* menu to launch the tool.
Publishing a FactoryTalk View ME Application

Use the FactoryTalk ViewPoint Administration tool to publish the displays that will be included in the web application. You can easily select all displays or choose the displays on an individual basis. These selected displays are then published and will be included with the runtime file when it is created.

Let’s go ahead and publish the application.

1. Maximize the FactoryTalk ViewPoint Administration tool window.

A list of displays within the Block_Machine_Advanced application will be displayed.
2. Uncheck the top box under **Web Enable** and to the left of **Block_Machine_Advanced** to uncheck all displays.
3. Scroll down and check the box next to **80 VP Production Data** and **81 VP Motor Start**.

4. Make sure the radio button next to **80 VP Production Data** is set as the Initial Display.

5. Click the **Publish displays** button.

![Publish displays button](image)

Depending on the size and number of displays, the publishing process may take several minutes when performed for the first time. Subsequent publish actions are optimized to analyze and prepare any displays that have been changed or selected to be published. You want to provide the ViewPoint users with two custom displays so you will publish only the **80 VP Production Data** and **81 VP Motor Start** displays.

Observe progress as the tool analyzes the selected displays and prepares these displays to be hosted in the browser. This will take a few minutes.

If any error or warning occur during publishing will be provided with a link to the publishing report which will describe any errors or warnings that may have occurred during publication. Any objects which are not supported in FactoryTalk ViewPoint will be listed as a warning.

There will be no errors for either display. When the publishing process is complete you should see the following display.
Congratulations! You have successfully published the FactoryTalk View ME application. Before we learn how to transfer the published displays to the PanelView Plus 6 terminal, let's configure security.
Configuring FactoryTalk ViewPoint ME security

Starting with ViewPoint 2.0 is the ability to secure individual display access as well as secure the new write capability. You can also continue to secure the entire application if you like, just as you have been able to do with previous versions of ViewPoint.

1. Select Security Settings on the red navigation bar in the FactoryTalk ViewPoint Administration tool window.

Note that in ViewPoint 2.x security is role based rather than user based. As a result only User Groups are shown and not individual users. You can expand on the groups that have a + sign beside them to see the list of users assigned to each group but you cannot individually select the user.
2. Select the **Secured Displays** tab.

Note that there is a checkbox labeled **Enable Write**, you can check this box to enable global write capability to allow all FactoryTalk ViewPoint users to have read-write access. You will not use this option, you will instead secure just the **80 VP Production Data** and **81 Motor Start** displays.

From the **Security Settings** area you can change FactoryTalk ViewPoint security for currently configured FactoryTalk User Groups of the Machine Edition (Local) directory. There are currently 3 Groups (**Operators, ShiftManagers & PlantManagers**) that have been configured in FactoryTalk Security for this application. Note that system defined roles are also available for use (Administrators, Authenticated Users & Windows Administrators). Additional Groups can be added using FactoryTalk View Studio or the Administration Console. Within the **Operators, ShiftManagers & PlantManagers** groups individual users have been created, these are **Operator, Supervisor & Engineer**. You will use individual user names to login into FactoryTalk ViewPoint in this lab.

Let’s configure security to give the **Operators, ShiftManagers & PlantManagers** groups specific access to the **80 VP Production Data** and **81 Motor Start** displays in the published web application.
3. Since security is optional and disabled by default in FactoryTalk ViewPoint, you can enable security by clicking the Enable Application Security checkbox.

Note that by default the Block_Machine_Advanced is highlighted which means that security can be configured for the entire application. For this example you will configure security for the 80 VP Production Data and 81 Motor Start displays.
4. Select **80 VP Production Data**.

5. Click on the checkboxes under the **Write** and **View** columns to set the security options as shown in the screenshot below. You may need to click a checkbox multiple times in order to get the selection you want.
6. Select **81 Motor Start** and set the same settings as you did for **80 VP Production Data**.
7. To see the effective security settings created by inheritance from the overall application security click the **Show Effective Permissions** checkbox. The default is to deny access.

8. Uncheck the **Show Effective Permissions** checkbox.

9. Finally, you need to save the security settings. Click the **Save** button.

Note: you will not be able to save until you uncheck the **Show Effective Permissions** checkbox.

10. Close the FactoryTalk ViewPoint Administration Tool window by clicking on **X** in the upper right-hand corner.
Creating a runtime file with ViewPoint content

Now that the selected displays have been published and security is configured, when the runtime file (*.MER) is created, the published displays and security settings will automatically be included in this file. The transfer of the MER file to the terminal will download the FactoryTalk View ME application as well as the published displays and security settings.

In subsequent creation of the runtime application, FactoryTalk ViewPoint will automatically check to see if any of the selected published displays have been modified since their last publish. If they have, the newer version of those displays are re-published in the background before being included in the MER file. There is no need for you to launch the FactoryTalk ViewPoint Administration again!

FactoryTalk ViewPoint is supported on the PanelView Plus 700-1500 terminals and allows one client connection in the FactoryTalk ViewPoint 1.1 through 2.x releases. Additional client licenses may be sold in a future release. The PanelView Plus terminals must have a minimum of 64MB Flash and 128 MB RAM to use ViewPoint.

As of 5.10.01 firmware (or higher), FactoryTalk ViewPoint is available as an option to load on the supported PanelView Plus terminals and is embedded into the firmware. The firmware upgrade wizard is used to upgrade terminals in the field.

The firmware can be downloaded from - http://support.rockwellautomation.com/ControlFlash/FUW.asp

As of 6.00 there is a single firmware package for all terminal types supported and ViewPoint is included in the firmware package.
1. Now let’s create our runtime application that you will download to the PanelView Plus. Select **Application > Create Runtime Application**.
2. Make sure **Runtime 6.1 Application** is selected for the **Save as type** field.

3. Make sure that the **Include ViewPoint Content** checkbox is selected; this will ensure that the ViewPoint published displays and security settings are included in the MER.

4. Select the **FactoryTalk ViewPoint 2.1** option from the drop down list to enable the write capability in your ViewPoint application.

5. Click **Save**.
6. Select **Yes** if prompted to replace existing runtime file.

7. Verify **English** is the only language selected to include in the runtime application. Click **Finish**.

The selected default language of the application in the FactoryTalk View ME Language Configuration editor, will determine which language is published in FactoryTalk ViewPoint. Support for multiple languages is available as of FactoryTalk ViewPoint 1.1. However, only one language can be published at a time.

As the runtime file is created, you will see the different stages. **ViewPoint** is one of the stages in this process and it may take some time to incorporate all of the published displays.
8. When the runtime file is finished compiling, launch the **Transfer Utility** from the tool bar.

The **Transfer Utility** is opened:
9. If the Block_Machine_Advanced.mer file you previously created is not already selected, click the **Source File:** browse button to select it. Click **Open.**

10. Double-click the **EtherNet, Ethernet** driver to expand the item.
11. Select the PanelView Plus found at IP address **192.168.1.20** by clicking on it once. (It may look different from the screen shot below.)

![Select destination terminal](image1)

12. Click **Download** to transfer the runtime file to your PanelView Plus terminal.

![Download screenshot](image2)

If you are asked if you want to overwrite the existing file, click **Yes**.

13. You will see a progress bar, and when the download process completes a confirmation dialog is shown, Click **OK** to acknowledge the dialog.

![Transfer Utility](image3)

14. Click the **Exit** button to close the **Transfer Utility**.

15. Now, turn to your **PanelView Plus**. Launch **FactoryTalk View ME Station** from the desktop icon on the PanelView Plus if it is not already running.

![FTViewME Station](image4)
16. On the FactoryTalk View ME Station configuration screen, click **Load Application [F1]** and select *Block_Machine_Advanced.mer*. Then click **Load [F2]** and **Yes [F7]** to replace the existing communication settings.

17. When the application is finished loading, click **Run Application [F2]** to start the Project.
Running an application in a browser

The next two sections involve running the ViewPoint ME application in a browser.

You can connect to an existing FactoryTalk ViewPoint application from any computer running a supported Web browser. All you need is the name or the IP address of the operator interface terminal (PanelView Plus 700 -1500) running the FactoryTalk ViewPoint Server for the application.

What is Silverlight?

FactoryTalk ViewPoint uses Microsoft’s Silverlight technology to visualize FactoryTalk View content in a browser. Microsoft Silverlight is a Web presentation technology that was created to run on a variety of platforms to deliver applications for the Web. It enables the creation of rich, visually stunning and interactive experiences that can run everywhere: within browsers and on multiple devices and desktop operating systems.

If Silverlight is not installed on a client computer that connects to the PanelView Plus and the client computer is connected to the internet, the client computer is automatically redirected to the Microsoft Silverlight installation site for download. If the client computer is not connected to the internet, instructions for installing Silverlight are automatically provided in the browser.

For the purpose of this lab, Silverlight has already been installed.
Connecting a Client

1. Launch Internet Explorer on the PC and enter the following address: ‘http://192.168.1.20/ftp’ where 192.168.1.20 is the IP address of your PanelView Plus.

2. Earlier, you had enabled Security and you are now required to login to access the ViewPoint application. Enter the username ‘Operator’ and password ‘operator’.

3. Click the Log On button.

Remember that you did not grant any rights to the Operators group, of which the user Operator is a member. An error is displayed on the log on page:

Unauthorized

You are not authorized to view display [::80 VP Production Data]
4. Click **Log Off** in the upper right side of the ViewPoint title bar.

5. Remember that you granted “View” rights to the **PlantManagers** group, of which the user Engineer is a member. Login again entering the username ‘**Engineer**’ and password ‘1234’.

6. Click the **Log On** button.
You have successfully logged in as the user Engineer.
7. Maximize the browser window if it is not already at full size. Note that the application can be resized in the browser.

FactoryTalk ViewPoint provides a thin client solution for FactoryTalk View. There is no need to install and maintain any Rockwell Software on the client machine. This lowers total cost of ownership, minimizes downtime and improves security. FactoryTalk ViewPoint also provides convenient access to FactoryTalk View applications from anywhere (central office, home, internet) which extends the reach of visualization to remote, casual and mobile users such as plant managers, central maintenance engineers, OEMs and System Integrators.

FactoryTalk ViewPoint provides a rich, interactive browser user experience. All of the navigation built into a FactoryTalk View ME application also works in the browser.

8. Click on the block icon in the upper left corner to navigate to the 81 Motor Start display.
9. Click on the **Stop/Start Pelletter** button a few times.

Notice that nothing happens, this is because you configured the **PlantManagers** group to have only “View” rights to FactoryTalk ViewPoint at the request of the end user.

10. Click **Log Off** in the upper right side of the ViewPoint title bar.
11. Remember that you granted both “View” and “Write” access to the ShiftManagers group, of which the user Supervisor is a member. Login again using the username ‘Supervisor’ and password ‘super’.

You have successfully logged in as the user Supervisor.

12. Click on the block icon in the upper left corner to navigate to the 81 Motor Start display.
13. Click on the **Stop/Start Pelleter** button a few times.

Notice that the button and motor change color as the motor starts and stops, this is because you configured the **ShiftManagers** group to have both “View” and “Write” rights to FactoryTalk ViewPoint at the request of the end user.

Also notice that writes are logged in the activity bar at the bottom on the ViewPoint screen.

Congratulations! You have successfully published the FactoryTalk View ME Block_Machine_Advanced application and can now connect to it using your web browser. Note that FactoryTalk ViewPoint automatically converted and published the displays to be included in the ME application; you did not need to make any modifications to the FactoryTalk View ME application to accomplish this.

Keep the browser open and move to the next exercise to learn how to run the **Block_Machine_Advanced** application using standard browser features.

**Using browser features with FactoryTalk ViewPoint**

We all use the internet frequently as a daily part of our work and home lives, so most people are familiar with how to use Browser functionality. Many applications run in the browser but do not utilize the
features of the browser. One of the major advantages of using FactoryTalk ViewPoint is that it allows you to use the features of the browser that you are already familiar with and does not require the installation of any Rockwell Software. This section focuses on exploring your FactoryTalk ViewPoint project using the features in the browser.

1. If the Internet Explorer on your PC is not currently open to ViewPoint, connect to the FactoryTalk ViewPoint server by typing the following link in Internet Explorer: ‘http://192.168.1.20/FTVP’ where 192.168.1.20 is the IP address of your PanelView Plus.

2. You will be instructed to enter this URL several more times throughout the remainder of the lab. If you get tired of manually entering the URL, just use the Address Bar located at the top of the Internet Explorer window. FactoryTalk ViewPoint supports all of the standard browser features!

Note that if you publish more than one Machine Edition display you can use the browser Back, then Forward buttons to navigate between displays as well as using the navigation buttons in the application.
3. Select **Favorites**.

4. Select **Add to Favorites**…

![Favorites Menu]

5. Enter a name for the new Favorite and click **Add**.

![Add a Favorite]

Note that you can now use Favorites to navigate between displays as well as using the navigation buttons in the application.
Trending

As of version 1.2 FactoryTalk View ME trending is supported in FactoryTalk ViewPoint. During the publishing process, the FactoryTalk View ME trend control is replaced by a compatible Silverlight Web control developed by Rockwell Automation. Therefore, the control will look different in your browser than it does on the PanelViewPlus terminal.

1. Turn to your PanelView Plus 6 terminal. The 35 Vacuum Silos Root Menu display should be open, as shown below, which contains the same trend as the one displayed in ViewPoint.
2. Now turn back to Internet Explorer and note the differences between the native Trend object on the PanelView Plus 6 terminal and the Silverlight Web control supported in the FactoryTalk ViewPoint client.

The FactoryTalk ViewPoint trend displays real-time data for the pens defined in FactoryTalk View Studio for the original trend object. Support for FactoryTalk View ME historical data is not available in this version. The properties that are retained in a FactoryTalk ViewPoint trend, if they were configured in the original FactoryTalk View ME trend object, are:

**General tab**
Chart style (Standard only), Chart update mode (Automatic only)

**Display tab**
No display settings are supported.

**Pens tab**
Tag or Expression definition, Visible (On or Off), Width, Min and Max values

**X-Axis tab**
Chart time span, Major grid lines (number).

**Y-Axis tab**
Automatic (Use best fit for data), Preset (Use pens’ Min and Max) and Custom (Actual min/max) value options, Major grid lines (number), Draw pens on independent scale (only).
Common tab
Size (overall) and Position settings.

Connections tab
Pen connections (tags or expressions) only.

A FactoryTalk ViewPoint trend consists of the trend chart, the pens, a current value legend, a value bar, an X and Y axis, play and pause buttons, and a pan and zoom slider. Note all the components that make up the trend control.

The current value legend on the right edge of the trend chart lists all the pens defined in the original FactoryTalk View trend object. It indicates the tag name or pen description, the value and color.
3. Clear or select the pen’s checkbox to hide or show the associated pen.

4. Click on the green pen **Temperature** to display its value range on the Y-axis, highlight it in the trend chart and if it intersects the value bar, its tooltip will be larger than others on the value bar.
5. Drag the **value bar** across the trend data in a FactoryTalk ViewPoint trend to display values associated with specific trend data points. Note the pen values in the tooltip and current value legend also update to the value where the pen intersects the value bar.

6. Click **Play** to resume viewing data from the point where the trend was paused.

The horizontal slider beneath the X-axis or the chart itself can be used to pan and zoom data.

7. Click and drag the **left handle** on the slider to zoom into the trend data.

8. Click and drag the **slider** to pan the trend data.

9. Pause the trend by clicking on the **Pause** button under the legend and click anywhere on the trend chart to **drag back and forth** as another method to pan the data.

10. Close Internet Explorer.
**Inactivity timeout**

FactoryTalk ViewPoint contains an inactivity timeout setting. If a web browser has been inactive for a period of time, a message is displayed stating that the session has expired. The inactive period is defined by a browser connected to the terminal with no display changes. When the session expires, the client access license is released which allows another client to connect to the terminal. If the application is secure, the currently logged on user will also be logged out.

1. Connect to the FactoryTalk ViewPoint server by typing the following link in Internet Explorer: ‘http://192.168.1.20/ftvp/admin’ where **192.168.1.20** is the IP address of your PanelView Plus. Pay special attention to include the /admin at the end of the URL.

2. Log on with the **User name:** ‘Engineer’ and **Password:** ‘1234’.

![FactoryTalk ViewPoint Authentication](image)
The FactoryTalk ViewPoint Administration is served from the PanelView Plus to your browser. The default inactivity timeout is 20 minutes. The inactivity timeout can be disabled or you can modify the inactivity timeout period.
3. Change the inactivity timeout to ‘2’ minutes and click the **Apply** button.

4. Connect to the running ViewPoint application by typing the following link in Internet Explorer: ‘http://192.168.1.20/FTVP’ where **192.168.1.20** is the IP address of your PanelView Plus (hint: you can avoid typing the URL by using the address bar at the top of the browser window).
5. Before the session expires, open a 2nd tab or browser instance and try to connect to the terminal by typing the following link in Internet Explorer: ‘http://192.168.1.20/FTVP’. Note that you are unable to connect to the terminal because the client license is already in use by tab 1.
6. Switch back to the first tab and wait for the session to expire. Remember, the session will not expire if you switch between displays. When the session expires you will see the message below in your browser.

---

Session has expired.

The session between this browser and the FactoryTalk ViewPoint Server has expired. Press F5 to refresh the browser.

If you are still unable to connect, contact your System Administrator or try connecting again later.
7. After the session expires, the currently logged in user is logged off and the client access license is now available for another client to connect. Go back to the 2nd tab and refresh the browser.

8. You can now log on with the User name: ‘Engineer’ and Password: ‘1234’.

After log on, you can access the FactoryTalk ViewPoint application.
9. Connect to FactoryTalk ViewPoint Administration again by typing the following link in Internet Explorer: 'http://192.168.1.20/FTVP/admin' where 192.168.1.20 is the IP address of your PanelView Plus. Pay special attention to include the /admin at the end of the URL.

10. Change the Inactivity Timeout back to Default settings and click Apply

![Inactivity Timeout Settings](image)

11. Close Internet Explorer (including all open tabs).

Congratulations! At the request of the end-user, you have published a display that is unique to the FactoryTalk ViewPoint clients and secured access to it using the features of ViewPoint 2.10. We hope you have enjoyed using some of the features present in Internet Explorer that help make using FactoryTalk ViewPoint an intuitive and enjoyable experience.
FactoryTalk Machine Edition - pick and choose

This section of the lab is a collection of several smaller exercises designed to guide the user through additional features of FactoryTalk ME. The format allows you to select the features you want to work with and you can complete these sections in any order. To help you choose, we have estimated the time it should take for each of them.

The following table lists the available sections and the page number each starts on.

<table>
<thead>
<tr>
<th>This feature…</th>
<th>Starts on page…</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Based Language Translation (20 minutes)</td>
<td>148</td>
</tr>
<tr>
<td>Email ActiveX control (20 minutes)</td>
<td>169</td>
</tr>
<tr>
<td>Auto Logout to designated screen (15 minutes)</td>
<td>169</td>
</tr>
<tr>
<td>Web Browser ActiveX Control (20 minutes)</td>
<td>203</td>
</tr>
<tr>
<td>Working with security (15 minutes)</td>
<td>227</td>
</tr>
<tr>
<td>Improving productivity with parameter passing (30 minutes)</td>
<td>242</td>
</tr>
<tr>
<td>Logging data (20 minutes)</td>
<td>262</td>
</tr>
</tbody>
</table>
Web Based Language Translation (20 minutes)

Completing this section requires approximately 20 minutes.

The FactoryTalk View Language Translation Utility

Rockwell’s FactoryTalk View Machine Edition and Supervisory Edition applications support multilingual capability at design time as well as run-time language switching between up to twenty different languages. To facilitate localization an HMI designer exports all of an application’s strings to a spreadsheet (XLS file). Up until now the translation has been manually performed on the exported strings, and the updated file (XLS) is imported back into the project.

The FactoryTalk View Translation Utility now provides a way to partially automate the translation process and provides a ‘gist’ translation of application strings exported to XLS format. The utility shall use internet resources to provide a ‘gist’ translation of the exported strings. A ‘gist’ translation is a rough translation that does not generally follow grammatical rules or synonym selection.

The translation utility is available in AID 481190 - FactoryTalk View Translation Utility for Supervisory Edition and Machine Edition
Open the application and export the languages

In this section, you will open an existing Machine Edition application which contains multiple languages and export those languages to an Excel spreadsheet.

1. Using the Start menu select *All Programs > FactoryTalk View Studio* to open *FactoryTalk View Studio*. 

![Start Menu Screenshot]
After the FactoryTalk View Studio for Machine Edition, application opens you will see a screen similar to the screenshot shown below:

2. Select the **Language_Translate** application from the list.

3. Ensure that the **Language:** field drop down list is set to **English**

4. Click **Open.**

A display called **Main Screen** has already been created that contains graphic objects you will use in this lab.
5. Double-click the **Main Screen** display to open it.
The **Main Screen** contains a sampling of graphic objects from one of the FactoryTalk View Machine Edition sample projects. Note the four national flags at the bottom of the display; these are four Language Switch buttons that will allow language switching at runtime.

Note the “**EN_Dictionary**” text object, we will use this object to show how a custom dictionary can be developed and used to translate your own custom words that a web based translation dictionary may not translate in the way you want.
6. From within Studio navigate to the *Tools > Languages* menu option

When the Language Configuration dialog box opens note that there are four languages installed in this application, with English as the current or default language. Also note that for each language the ID begins with two lower case letters such as `en` for English, these lower case letters will be used by the custom dictionary later in this lab to define the languages installed in the application.

7. Click the **Export** button.
8. Click the radio button next **Export strings for all languages to an Excel spreadsheet**.

9. Ensure that the **Optimize duplicate strings** checkbox is checked, this checkbox must be used with the translation utility.

10. Click **Next**.

![String Import Export Wizard - Export Operation](image)

Note the default location for exported language files is the C:\Users\Public\Documents\RSView Enterprise\Strings folder. You will use the default location for this lab.
11. Click Finish.

A progress bar may appear along with a message stating that the export will take some time. Note that there is no dialog box to prompt when the export has completed. For this lab the export should take about one minute and will generate the Excel spreadsheet when complete.

While the language export is processing open the FactoryTalk View Translation Utility.
Open the FactoryTalk View Translation Utility

In this section, you will open and configure the FactoryTalk View Translation Utility along with using a custom dictionary to translate the exported string spreadsheet.

1. Using the Start menu select All Programs>FactoryTalk View Translation Utility> FactoryTalk View Translation Utility.
2. Click the browse button next to the Select Exported Language File (XLS) field.

3. Browse to the C:\Users\Public\Documents\RSView Enterprise\Strings folder and select the Language_Translate_XX.XLS file. Note that the _XX numbering of the file may be different than what is shown below.

4. Click Open.
Note that once the spreadsheet has been loaded by the Translation Utility the default language is listed and a selection of languages is listed in the **Select Target Languages**. The languages already install in the application will have a check in the checkbox however more languages can be selected for translation. Note that if more languages are selected they must be added in the application using Studio before the spreadsheet can be imported.

You will leave the default settings for this lab.
Earlier in this lab we talked about using a custom dictionary. On the “Main_Screen” display there is a Text object which contains the English text EN_Dictionary as shown below. You as the engineer want to ensure that when you translate this specific piece of text it translates to what you want it to say. You will use a custom dictionary for this purpose and reference it within the FactoryTalk View Translation Utility.

The contents of the custom dictionary XLS file are shown below. As mentioned earlier the custom dictionary uses the first two letters of the language ID to perform the conversion. Note that during the translation if the keyword EN_Dictionary is found the FactoryTalk View Translation Utility will replace that text string with the following:

SP_Dictionary for Spanish
CN_Dictionary for Chinese
FR_Dictionary for French
5. Check the *Use Custom Dictionary* checkbox.

6. Click the browse button next to the *Use Custom Dictionary* field.
7. Browse to the `C:\Lab Files\ViewME\Advanced\ViewME\Pick and Choose` folder and select the `Custom_Dictionary.XLS` file.

8. Click **Open**.
9. Click **Translate**.

A dialog box will appear stating the cautions to be followed with automated language translation.

10. Click **Continue**.

This application will take about two minutes to translate all strings, the larger the number of strings in the application the longer the translation will take. The dialog box will show you which language and which strings are being translated as the translation progresses.
The following dialog box will appear when the translation is complete.

11. Click **View File** to open the translated Excel file.

![View File button highlighted](image1)

12. Scroll down and find the strings translated by the custom dictionary.

![Translation dictionary table](image2)

13. Close Excel

14. Click **Exit** to close the **Translation Utility**.

![Exit button highlighted](image3)
**Import the translated strings into Studio**

In this section you will import the translated strings back into the Machine Edition application.

1. Go back to Studio and reopen the **Language Configuration** dialog box if necessary.

2. Click **Import**.

3. Select the **Import strings from an Excel spreadsheet into all languages defined in the spreadsheet** radio button.

4. Click **Next**.
5. Click the browse button next to the **Source Folder** field.

![String Import Export Wizard - Select File]

6. Select the **Language_Translate_XX.XLS** file. Note that the _XX_ numbering of the file may be different than what is shown below.

7. Click **Open**.
8. Click **Finish**.

![String Import Export Wizard - Select File](image)

A Progress bar will be displayed during the import.

9. Click **OK** to close the Language Configuration dialog box when the import is complete.
Test run the application

In this section you will test your application and see the runtime results of the language translation.

1. From within Studio select the menu option Application > Test application.

2. Select the checkbox beside each language.

3. Click Finish.
A progress bar will appear as the application is prepared for the test run, in a few minutes the application will open.

4. Click on the four Language Switch buttons at the bottom of the display and note how the text strings change.

5. When you have finished working with the application click the **Shutdown** button or press the Ctrl and X keys on the keyboard to stop the test run.

Congratulations! You have successfully used the features of FactoryTalk View Studio for Machine Edition and the FactoryTalk View Translation Utility to translate language strings.
Email ActiveX control (20 minutes)

Completing this section requires approximately 20 minutes.

FactoryTalk Machine Edition Email ActiveX
The Email ActiveX control uses a SMTP server to send an e-mail or text message. The SMTP server chosen for this lab is hosted by Gmail – smtp.gmail.com. The Gmail SMTP server requires a user name and password. Therefore, a Gmail account was setup. The Gmail SMTP server also requires the use of Secured Socket Layer or SSL (SSL was set and the port used was 465 – the SSL port). Whether anonymous, SSL, or user name & password are required is going to depend on the SMTP server.

Once the FactoryTalk View ME application starts, a display that contains the e-mail ActiveX is launched first. This ensures the ActiveX has focus right away at application startup. The initial display is setup to “appear” in a non-viewable area of the application (X position = -250 & Y position = -250). This gives the effect of running in the background. This initial display also executes a startup macro that assigns the actual initial display number to the Remote Display Number global connection. The actual initial display is called Main.

Once an alarm is generated, the corresponding alarm message is copied to the Email body HMI tag (Message to Tag check box is checked in Alarm Setup). This is the e-mail message that is sent. You could also copy your own text from the controller to Email body instead of using FactoryTalk Machine Edition alarm messages. This lab used the Message Notification connection in Alarm Setup. Once a message is copied into the Message connection (hmi tag SMTP_EMAIL\Body), Message Notification gets set (hmi tag SMTP_EMAIL\Alarm_Message_Notification). As long as this is set, any additional messages that need to be sent are queued up by FactoryTalk View Machine Edition - up to 128 messages. A global connection macro, Macro2, runs whenever Message Notification gets set, which triggers the Email ActiveX. The Email ActiveX Handshake connection is used by another Global Connection to run Macro3 which clears the Email ActiveX trigger and the Email ActiveX Handshake. The Message Notification connection is cleared 1 second later by the ME Alarm system so that the next queued message, if any, can be sent.
Launch Application Manager to restore the runtime file.

1. Click on **Start > Programs > Rockwell Software > FactoryTalk View > Tools > Application Manager**

2. Select the **Restore runtime application** radio button and click **Next**.

3. Click browse button to the right of the **Specify the runtime application to restore** field.
4. Browse to and select the C:\Lab Files > ViewME - Advanced > ViewME > Block_Machine_Advanced_Begin.mer and click Open.

5. Select the **Restore the runtime application and FactoryTalk Local Directory** radio button. Type in the password ‘Data’ (with capital D) in the password area and click **Next**.
6. Click **OK** to accept the following dialog.

You get this warning because you are going to convert the MER file back to design time and you chose to also restore the FactoryTalk Local Directory which contains all the users, groups and policies. The FactoryTalk Local Directory will be replaced with the one in the MER file.

7. Change the name to ‘Block_Machine_Advanced_email’ and click **Finish**.
Since you are also restoring the FactoryTalk Directory you will get a popup to enter User Name and Password for the Local FactoryTalk Directory currently existing on the computer.

8. Enter ‘labuser’ for the User Name.

9. Enter ‘rockwell’ for the password

10. Click **OK**, the application will begin conversion and display a progress bar. The conversion may take a few minutes depending on the size of the application.
Open the restored application.

1. Click on **Start > Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio**

2. Select the **Block_Machine_Advanced_email** application file.

3. Select **English** as the default language.

4. Click **Open**.
Let’s Explore the E-mail ActiveX connections.

1. Open the 99 Data Logging display by double clicking it.

2. Using the Object Explorer, find ME_EmailSenderControl1 and double click it to bring up its properties.
3. Click the **Connections** tab to view the tags associated with the ActiveX control.

![ME_EmailSender Control Properties](image)

The control uses an SMTP server to send an e-mail or text. Depending on the specific SMTP server, it may require a User id, Password, SSL, and SSL port number. Or, it may send the e-mail or text anonymously and use the standard port number. This demo used G-Mail as the SMTP server.

**SMTPSrvrAddress** is smtp.gmail.com.

**SMTPUserid** is Panelview.Plus6@gmail.com.

**SMTPUserPwdd** is the password for the G-mail account.

**SSL Enable** is set to a 1 as G-mail requires the use of SSL (Secured Socket Layer).

**SMTPPort** is set to 465, which is the SSL port number.

**From_Address** is set to the e-mail address of the account that is sending the e-mail. In this case it is Panelview.Plus6@gmail.com.

**From_Name** is the name of the person or entity sending the e-mail. The demo used “PanelView Plus SMTP Demo”.

These parameters were passed to the ActiveX control by using a macro.

4. Close the **ME_EmailSender Control Properties** window and the **99 Data Logging** display. Select **No** if prompted to save any changes.
Let's assign an alarm message to send via a text message.

1. Open **Alarm Setup** by double clicking it.

The **Alarm Setup** was configured to send an alarm message to a tag instead of a printer or a display. The alarm message is the body of the e-mail or text. Alarm message #1 was used in the demo. It is configured to send the text “Pelleter Speed is High!” to SMTP_EMAIL\Body when SMTP_EMAIL\Alarm_Digital2 goes to a 1.

The Optional trigger connections – **Message** and **Message Notification** were tied to HMI tags. **Message** is associated to the ActiveX control's Body property. This is the body of the e-mail or text message and in this case is alarm message #1. The trigger setting connection, **Message Notification**, is assigned a digital tag and it goes to a 1 when trigger setting connection, **Message**, receives a new string value. **Message Notification** is used to trigger the E-mail ActiveX control.
2. Click the **Messages** tab. Enter ‘Pelleter Speed is High!’ in the **Message** field for **Alarm message 1** and check the **Message to Tag** check box.

3. Click the **Triggers** tab and select **Alarm Digital2**.

4. Assign HMI tag, **SMTP_EMAIL\Body**, to **Message** and HMI tag **SMTP_EMAIL\Alarm_Message_Notification**, to **Message Notification** using the **Tag Browser**.

<table>
<thead>
<tr>
<th>Optional trigger connections:</th>
<th>Name</th>
<th>Tag or expression</th>
<th>Tag</th>
<th>Exprn</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Handshake</td>
<td></td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ack</td>
<td></td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remote Ack</td>
<td></td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remote Ack Handshake</td>
<td></td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message</td>
<td>SMTP_EMAIL\Body</td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Notification</td>
<td>SMTP_EMAIL\Alarm_Message_Notification</td>
<td>***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Handshake</td>
<td>SMTP_EMAIL\Alarm_Message_Notification</td>
<td>***</td>
<td></td>
</tr>
</tbody>
</table>
Shown below are the settings associated with the **Advanced** tab.

The **Advanced** tab contains a **Time settings** property – **Hold time**. If no handshake is used, the **Hold time** is used to reset **Message Notification** which stays at 1 for the hold time of 1000mS (or 1 second) and then is reset to 0. The ActiveX control writes a 1 to its **Handshake** property once it recognizes the e-mail or text message (the string contained in **Body**).

5. Click **OK** to exit the **Alarm Setup**.
Macros were used to perform the E-mail logic functions rather than a PLC. Let’s explore the macro configurations.

1. Open **Global Connections** by double clicking it.

2. Click on the **Macro** tab to explore its configuration.

   Remote Macro 2 is triggered once **SMTP_EMAIL\Alarm_Message_Notification** goes to a 1 (this is associated to Alarm Setup’s **Message Notification**).

   Remote Macro 3 is triggered once **SMTP_EMAIL\Handshake** goes to a 1.

3. Close the **Global Connections** window.
4. Open *Macro 2* by double clicking it to explore its configuration.

![Image showing tree view of files]

**Macro2** triggers the e-mail to be sent and adds the **Subject** to the e-mail.
5. Close the Macro2 window.

6. Open Macro3 by double clicking it to explore its configuration.

Macro3 resets the E-mail ActiveX Trigger and Handshake.

7. Close the Macro3 window.
Create the Runtime (.MER) file

1. Click Application and select Create Runtime Application.

2. Leave the defaults as shown and click Save.
3. Leave the defaults as shown and click *Finish.*
Download the runtime .MER file to a PanelView Plus 6 terminal

1. To download the runtime .MER to the PanelView Plus terminal at your workstation, select Transfer Utility on the Tools menu:

![Transfer Utility](image)

Note that you can also click the Transfer Utility icon on the toolbar.

The Transfer Utility is opened:

![Transfer Utility](image)
2. Click the **Source File** browse button to select the runtime .MER file to download. Select the **Block_Machine_Advanced_email.mer** file you previously created and click **Open**. Note that the MER file may already be selected for you if you have previously downloaded the same application.

3. Double-click the **EtherNet, Ethernet** driver to expand the item.
4. Select the PanelView Plus found at IP address 192.168.1.20 by clicking on it once. (It may look different from the screen shot below.)

5. Click Download to transfer the runtime file to your PanelView Plus terminal.

If you are asked if you want to overwrite the existing file, click Yes.

6. You will see a progress bar, and when the download process completes a confirmation dialog is shown. Click OK to acknowledge the dialog.

7. Click the Exit button to close the Transfer Utility.
Run the application on a PanelView Plus 6 terminal

In this section, you will work with a PanelView Plus terminal. Follow the steps below on your terminal to run the FactoryTalk View ME runtime application you just created.

1. If FactoryTalk View ME Station is not already running on the terminal, double-tap the **FTView ME Station** icon on the desktop.

![](image)

2. Press **Load Application [F1]**.

![](image)

3. Select the *Block_Machine_Advanced_email.mer* file from those available from the terminal's Internal Storage.
4. Press **Load [F2]** to load the runtime file into memory.

5. When you are prompted, press **Yes [F7]** to overwrite the terminal’s current communication configuration with the configuration contained within the Block_Machine_Advanced.mer file.

6. Once successfully loaded, press **Run Application [F2]** to start executing the runtime file.
1. Once the application starts, select the *E-mail* icon.

2. Select your cell phone provider from the list. If it's not shown and you know the SMS server name, you can select *Other* and enter it manually.
3. Click the String Input Enable field above the cell providers and enter your cell phone number, including the area code, using the on-screen keyboard.

The Adressee field will populate with the phone number & SMS server name.

4. Click the Generate Alarm button, 

The **Subject** & **Body** fields will populate as shown.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Pelletier Alarm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Body</strong></td>
<td>Pelletier Speed is High!</td>
</tr>
<tr>
<td><strong>Signature</strong></td>
<td></td>
</tr>
</tbody>
</table>

The **Mail Status** field will update and either show that the message was sent successfully or if any errors occurred.

| Mail Status | 0 | Mail Sent Successfully |

Congratulations! You have successfully created a FactoryTalk View Studio for Machine Edition application that sends alarm messages via text.
Auto Logout to designated screen (15 minutes)

Completing this section requires approximately 15 minutes.

As a controls engineer you want to tighten security on the PanelView Plus and control what screen is displayed when a user is automatically logged out after a period of inactivity. You also want to configure the application to navigate to different screens when the auto logout occurs depending on who is currently logged in.

Open the application and examine the configuration

In this section, you will open an existing Machine Edition application and examine the configuration used to custom configure automatic user account log put.

1. Using the Start menu select All Programs> FactoryTalk View Studio to open FactoryTalk View Studio.

After the FactoryTalk View Studio for Machine Edition, application opens you will see a screen similar to the screenshot shown below:
2. Select the **ScreenSwitch** application from the list and click **Open**.

![ScreenSwitch application selection](image1)

Several screens have already been created that contain graphic objects you will use in this lab.

3. Double-click the **Project Settings** option in the Application Explorer and navigate to the **Runtime** tab.

Note that the application is configured to auto-logout after 1 minute of inactivity and has nothing defined for the **Return to graphic on logout** option.

4. Click **OK**.

![Project Settings configuration](image2)
5. Double-click the **Runtime Security** option in the Application Explorer.

6. Highlight the **Engineer** user account and notice that a Logout Macro called **ADMINLOGOUT** has been assigned to this user. Also note that the **DEFAULT** user account has the **A** security code removed. The Display called **Secure** has been assigned the security code **A**, as a result the **DEFAULT** user account cannot access the **Secure** display.

7. Click **Close**.
The `ADMINLOGOUT` macro writes a value of 1 to the **Global Connection** called `RemoteDisplayNumber` which will call up that display when the user account **Engineer** is logged out.

8. Double-click the **Global Connections** option in the Application Explorer.

9. Select the **Display** tab

![Global Connections in Application Explorer](image)

Note that an HMI memory tag called `remoteDisplay` has been assigned to the **Remote Display Number** connection.

10. Click **OK**.
Test Run the Application

1. Click on the **Test Application** button. The application will load and run in test mode within Studio.

Note that this application has Diagnostics configured to help show how the application works, you will need to *Close* the Diagnostics window from time to time to close the popup Diagnostics dialog box.

![Application in studio](image)

When the application starts note that the **DEFAULT** user account is logged in.

2. Click on the **Secured Display** button, note that as the **Default** user you do not have access.

3. Click on the **Login** button.
4. Enter User: ‘Engineer’ and Password ‘1234’.

Note that the Engineer user account is now logged in.

5. Click the Secured Display button.

The Secured Display will open, remain on the Secured Display for one minute with no mouse or keyboard activity to allow the automatic logout to occur.
Note that after 1 minute of inactivity the Engineer user account will be logged out and the application will navigate to the Main Screen even though we did not define any screen for Return to graphic on logout in the Projects Settings Runtime tab.

**NOTE:** Any user defined within the application can have a unique logout macro. Using the method described in this lab any user account can be configured to navigate to a unique display when they are logged out.
The **AdminLogout** logout macro assigned to the **Engineer** user account allowed navigation back to the Main Display by writing a 1 to the **Remote Display Number** Global Connection.

6. Close the Test Application.
Let’s examine a couple of Macros related to Display Settings.

1. Double click **Displays** & then **Main**  
   Right click on a blank area of the **Main** display and select **Display Settings**. Note that the **Main Display** is configured as **Display Number 1**.

   ![Main Display](image1)

   ![Display Settings](image2)

   The Main Display has a macro assigned to it that writes a **0** to the **Remote Display Number** Global Connection which gives display navigation control back to the operator.

   ![Macro](image3)
Congratulations! You have successfully used the features of FactoryTalk View Studio for Machine Edition to customize the display behavior for a unique user account when an auto-logout condition occurs.

![Form](image)

<table>
<thead>
<tr>
<th>Tag</th>
<th>Tag</th>
<th>Expression</th>
<th>Exprn</th>
</tr>
</thead>
<tbody>
<tr>
<td>remoteDisplay</td>
<td>1</td>
<td>• • 0</td>
<td>• •</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>• •</td>
<td>• •</td>
</tr>
</tbody>
</table>
Web Browser ActiveX control (20 minutes)

Completing this section requires approximately 20 minutes.

FactoryTalk Machine Edition Web Browser Application

The Web Browser ActiveX allows one to view a web page or html file directly in the ME application. This method prevents the user from browsing to other websites as the specific URL information is one of the connection properties of the ActiveX. This lab section also uses a global object to demonstrate how a single display can be used to view multiple URL’s, HTML files, and third party applications.

Launch Application Manager to restore the runtime file.

1. Click on **Start > Programs > Rockwell Software > FactoryTalk View > Tools > Application Manager**

2. Select the **Restore runtime application** radio button and click **Next**.

3. Click on browse button to the right of the **Specify the runtime application to restore**: field.
4. Browse to and select the *C:Lab Files > ViewME - Advanced > ViewME > Block_Machine_Advanced_Begin.mer* and click *Open*.

5. Select the *Restore the runtime application and FactoryTalk Local Directory* radio button. Type in the password ‘*Data*’ (with capital D) in the password area and click *Next*. 
6. Click **OK** to accept the following dialog.

![Application Manager dialog]

You get this warning because you are going to convert the MER file back to design time and you chose to also restore the FactoryTalk Local Directory which contains all the users, groups and policies. The FactoryTalk Local Directory will be replaced with the one in the MER file.

7. Change the name to ‘Block_Machine_Advanced_web’ and click **Finish**.

![Application Manager window with 'Block_Machine_Advanced_web' entered]

Click **Finish**.
Since you are also restoring the FactoryTalk Directory you will get a popup to enter User Name and Password for the Local FactoryTalk Directory currently existing on the computer.

8. Enter ‘labuser’ for the User Name.

9. Enter ‘rockwell’ for the password

10. Click OK, the application will begin conversion and display a progress bar. The conversion may take a few minutes depending on the size of the application.
Open the restored application.

1. Click on **Start > Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio**

2. Select the **Block_Machine_Advanced_web** application file.

3. Select **English** as the default language.

4. Click **Open**.
Let’s examine the properties of the Goto Display Button Global Object.

1. Expand the *Graphics > Global Objects* folders. Open the *Screen Objects* display.

2. Find the Goto Display Button and right click on the button.

3. Select the *Global Object Parameter Definitions*.
Five parameters have been defined that the global object will use.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>Label</td>
</tr>
<tr>
<td>#2</td>
<td>URL</td>
</tr>
<tr>
<td>#3</td>
<td>program path</td>
</tr>
<tr>
<td>#4</td>
<td>program parameters (command line)</td>
</tr>
<tr>
<td>#5</td>
<td>trigger</td>
</tr>
</tbody>
</table>

4. Close the *Global Object Parameter Definitions* pop-up.

Now let's assign the display to open and the parameter list to pass to that display.

5. Open the Goto Display Button properties (Right click the Goto Display Button and select **properties**). On the **General** tab, assign the **Diag_Popup** display and enter the Parameter list as shown.

6. Select the **Label** tab, click **Insert Variable** and select **String**.
7. Enter ‘(#1)’ for the Tag and click **OK**.

8. Click **OK** to accept the changes.

9. Close the **Screen Objects** display and select **Yes** to save the changes.
Now let’s assign HMI tags to the specific instance of the Goto Display Button.

1. Open the *Diag. Selection* display.

2. Find *GotoDisplayButton1* using *Object Explorer*.

3. Right click on *GotoDisplayButton1* and select *Global Object Parameter Values*.
4. Assign the tags shown for parameters #1 & #2 using the tag browser.

5. Click OK to accept the changes.

6. Close the **Diag_Selection** display and select **Yes** to save the changes.

Now let’s assign the connections for the Web Browser and Program Launcher ActiveX controls.

1. Open the **Diag_Popup** display.
2. Using **Object Explorer**, find and double-click **RSViewWebBrowserControl1**.

![Object Explorer](image1)

3. Select the **Connections** tab and assign ‘#{1}’ to the **URL** property.

![Connections Tab](image2)

4. Click **OK** to accept the changes.
5. Using **Object Explorer**, find and open **MEProgramLauncher1**.

6. Select the **Connections** tab and assign `{#2}`, `{#3}`, and `{#4}` to the properties shown below.

7. Click **OK** to accept the changes.

The parameters `{#1}`, `{#2}`, `{#3}`, and `{#4}` actually correspond to #2, #3, #4, and #5 parameters that were added to the parameter list for the Goto Display Button on the global object screen, **Screen Objects**.

![Object Explorer with MomentaryPushButton1 highlighted](image)

9. Select the **Connections** tab and assign tag `{#4}` to the **Value** property.

![Momentary Push Button Properties dialog box with Value set to #4](image)

10. Click **OK** to accept the change.

11. Close the **Diag_Popup** display and click **Yes** to save the changes.
Create the Runtime (.MER) file

1. Click Application and select Create Runtime Application.

2. Leave the defaults as shown and click Save.
3. Verify that *English* is selected and click *Finish.*
Download the runtime .MER file to a PanelView Plus 6 terminal

1. To download the runtime .MER to the PanelView Plus terminal at your workstation, select **Transfer Utility** on the **Tools** menu:

![Transfer Utility](image)

Note that you can also click the **Transfer Utility** icon on the toolbar.

The **Transfer Utility** is opened:

![Transfer Utility](image)
2. Click the **Source File** browse button to select the runtime .MER file to download. Select the **Block_Machine_Advanced_web.mer** file you previously created and click **Open**. Note that the MER file may already be selected for you if you have previously downloaded the same application.

3. Double-click the **EtherNet, Ethernet** driver to expand the item.
4. Select the PanelView Plus found at IP address 192.168.1.20 by clicking on it once. (It may look different from the screen shot below.)

5. Click **Download** to transfer the runtime file to your PanelView Plus terminal.

   ![Download Button]

If you are asked if you want to overwrite the existing file, click **Yes**.

6. You will see a progress bar, and when the download process completes a confirmation dialog is shown. Click **OK** to acknowledge the dialog.

   ![Confirmation Dialog]

7. Click the **Exit** button to close the **Transfer Utility**.
Run the application on a PanelView Plus 6 terminal

In this section, you will work with a PanelView Plus terminal. Follow the steps below on your terminal to run the FactoryTalk View ME runtime application you just created.

1. If FactoryTalk View ME Station is not already running on the terminal, double-tap the **FTView ME Station** icon on the desktop.

2. Press **Load Application [F1]**.

3. Select the **Block_Machine_Advanced_web.mer** file from those available from the terminal’s Internal Storage.
4. Press **Load [F2]** to load the runtime file into memory.

5. When you are prompted, press **Yes [F7]** to overwrite the terminal’s current communication configuration with the configuration contained within the Block_Machine_Advanced.mer file.

6. Once successfully loaded, press **Run Application [F2]** to start executing the runtime file.

7. Once the application starts, click the **Web Browser** icon,
8. Click the **Launch AB.com** button.

The AB.com website will launch within the pop-up display. It may take a minute or two the first time the web page is launched.

9. Click the links on the webpage and explore the website. Use the **Back** button if there isn’t a link to return to the previous webpage. Notice that this is the only website that you can browse.
10. Click the **Launch Program** button.

The *PowerFlex 70* Technical document will launch within the Foxit PDF Reader.

11. Swipe your finger across the screen to turn the pages. Tap the screen to zoom into the page.

The Web Browser can be used to go to a specific website and the PDF reader can then be launched to provide specific navigation instructions.
12. Close the Foxit PDF Reader and the pop-up display.

13. Click the *Instant Fizz HTML* button.

The *Instant Fizz HTML* file will launch within the pop-up display. You may have to click the window in order to refresh the html page.
14. Click the **Launch Program** button.

The Media Player will launch and play a video.

The Web Browser can be used to show a ReadMe file or a snap shot of a process in HTML format. Then using the Program Launcher, a video can be shown that illustrates the steps involved in the process.

15. Close the Media Player and the pop-up display.

Since the Web Browser ActiveX itself isn’t supported as a global object, the demo used a Goto Display Button global object and parameter passing. In this manner, a single pop-up display was used to show two different uses of the Web Browser ActiveX and the Program Launcher ActiveX.
Working with security (15 minutes)

Completing this section requires approximately 15 minutes.

FactoryTalk View ME user accounts work with security codes to determine whether a user gets access to the system at run time. Each display in a project is assigned a security code. Users are also assigned security codes. At runtime, if a user is assigned access to the security code for a particular display they are allowed to access the display.

When you set up security in FactoryTalk View ME, user accounts are stored in the FactoryTalk Directory. This means that those same user accounts can be used not only with the FactoryTalk View ME application, but with any other FactoryTalk-enabled software running on the same computer. You can add a FactoryTalk user account that is separate from a user’s Windows account, or a Windows-linked user account that inherits all the operating system user settings.

FactoryTalk View ME security allows you to assign any one of 16 security codes (A-P), or an asterisk (*) to your displays. Users are also assigned security codes, and if a user’s assigned security codes match the code assigned to a display, they are allowed access to the display. All users are allowed access to displays with the asterisk for the security code. By default, new users have all 16 security codes assigned to them. The Default user also has all 16 security codes assigned. The Default user account is used when no user is logged on.

In this section you will:

- Add a User and grant permissions to the 3A Filling – Recipes display
- Create and apply security to a User Group.
Open the InstantFizz Advanced application

1. To open FactoryTalk View Studio, click Start > All Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio. If FactoryTalk View Studio is already open, then select File > Open Application… to switch applications.

2. Select the existing InstantFizz Advanced application, make sure English is selected for the Language, and click Open:

3. If you are switching applications you will be prompted to close the currently open application, click the Yes button to proceed.
View the security code for a display

1. In the application Explorer, expand Graphics and Displays, then double-click on the 3A Filling - Recipes display to open it:

   ![Image of the Explorer with the 3A Filling - Recipes display open]

The display is opened:

   ![Image of the 3A Filling - Recipes display]

The 3A Filling – Recipes display is where at run time you can upload and download recipes or lists of ingredients, view recipe contents, and select recipes to use.
2. Select **Display Settings** on the **Edit** menu. The Display Settings editor for the **3A Filling – Recipes** display is opened:

![Display Settings editor](image)

The Display Settings editor is where you can specify a security code for a display. Note that the **3A Filling - Recipes** display is assigned Security Code **A**. Click **Cancel** to close the dialog.

### Create a new user account and assign security codes

1. Since the display has a security code of **A**, you will now create a new user account called "**Engineer**" that is assigned the same security code. Double-click **Runtime Security** in the application Explorer to launch the Runtime Security editor:

![Runtime Security editor](image)
2. As mentioned earlier, the **Default** user has all 16 codes assigned by default. First remove security code A from the Default user so that only users assigned security code A can access the 3A Filling – Recipes display. Select the **Default** user and click to uncheck the checkbox for security code A as shown below. You then need to click **Accept** to apply the change.

![Screenshot of the security settings](image)

Note that on this screen and in the steps below, you may see different users or user groups from what is shown.

3. Click the **Add** button to add a new user account. The **Select User or Group** dialog is shown; select **Show users only**.
4. Click *Create New* and select *User…*

![Create New and select User](image)

5. On the *General* tab, enter ‘*Electrician*’ as the *User name*. Check *Password never expires*. Enter ‘*1234*’ as the *Password* and confirm it in the *Confirm* box.

![New User dialog with Electrician selected](image)
6. Click **OK** to add your new user. The **Electrician** user should now show up in the list on the **Select User or Group** dialog.

7. Select the **Electrician** user account and click **OK**. The user is added to the Runtime Security user account list. Because new users by default are granted access to all 16 security codes, the **Electrician** user is already assigned the security code A. This is what you want, so leave the security settings as they are.

8. Click **Close** to close the **Runtime Security** editor. Click **Yes** when you are asked if you want to save the changes:

You have now created a new user and configured security for it. Remember that you will need to login to the application as user **Engineer** from now on when running the application if you want to access the **3A Filling – Recipes** display.

**Exercise security**

The following steps show you how to log in as user **Electrician** when the application is running. Try accessing the **3A Filling – Recipes** display before and after you log in as **Electrician** to observe what happens when you do not have rights to a display. Remember that the password for the Operator user is ‘Operator’.
1. Double-click **Startup** on the application Explorer to configure startup options, check the *Initial graphic* checkbox if it isn’t selected already and verify that **1 Overview** is selected as the *Initial graphic*:

![Startup configuration dialog box]

2. Click **OK** to accept your changes.

3. Make sure that you don’t have the project running from any previous section. If you do, click **Exit** to close it.

4. On the Application menu, click **Test Application**.

![Test Application menu option]
5. Select *English* as the initial runtime language and click *Finish*.

![Runtime Application Wizard - Select Languages](image)

The runtime file is created. You will see progress bars while it is compiled, and when it is complete the application will run. It will take a couple minutes.

6. When the application starts, the Overview screen is shown, and you can see that the logged in user is *DEFAULT*:

![Overview Screen](image)

7. Because you are logged in as *Default*, you should not have security rights to the *3A - Filling Recipes* display that you configured security for. (Remember that the display is assigned security code “A”, and the Default user does not have access rights to “A”.) Click the *Recipes* button to see what happens:

![Recipes Button](image)

You should see an error saying access is denied because the currently logged in user does not have security access to the display. You can click *Close* to close the error display.

8. Now you will log in as the *Electrician* user you created, and who does have security access to the *3A – Filling Recipes* display. Click the *Security* button on the bottom toolbar.
9. A second toolbar is displayed. Click **Log In**:

![Login Screen]

The **Login** display is opened. Click the **User (F2)** button and use the keyboard to enter ‘Electrician’. Click the **Password (F3)** button and enter ‘1234’ for the password. It should look like this:

![Login Input]

Click the **enter** button to log in.

10. Click **Close** to close the Security menu.

![Close Button]

11. Now click the **Recipes** button to see what happens.

![Recipes Menu]

The recipes screen is opened, because as the **Engineer** user you have security access rights!

**NOTE:** You may see errors that an item is no longer available if the **Recipe Management** section has not yet been completed. You will notice that the **Engineer** user successfully opened this display however, the recipe objects are blank.

12. Click the **Close** button to close the recipes display.

13. Click **Exit** to close the running application.

**Using security groups**

Just as you assign security rights to a user account, you can also create user groups, include users in those groups, and assign security rights to the group as a whole. Any security rights that are applied to
the group will apply to each user account in the group.

In this section you will add user accounts to a group and configure security for the group.

1. Double-click **Runtime Security** in the application Explorer to launch the Runtime Security editor:

2. Click the **Add** button to create a new user group:

3. Click **Create New**, then **User group** to create a new group.
4. On the **New User Group** dialog, enter 'Managers' for the group **Name**.

   Click **Add** to add users to the new group:

   ![New User Group dialog](image)

5. Click **Show users only** to see the list of configured users. Hold down the **Ctrl** key and click the **Engineer** and **Supervisor** users to select them.

   ![Select User or Group dialog](image)
6. Click **OK** to add the users to your group. They should show up in the **New User Group** dialog in the **Members** list:

![New User Group dialog showing added users Engineer and Supervisor](image)

7. Click **OK** to create the new user group with group members **Engineer** and **Supervisor**. The **Select User or Group** dialog will list your new group:

![Select User or Group dialog showing Managers group](image)

8. Select the **Managers** group and click **OK**.
The Managers group is now ready for you to configure FactoryTalk View ME security:

You can tell a group from a user in the Runtime Security list by the square brackets around a group name.

You cannot manage users and groups, other than creating a new user or group, from the FactoryTalk View Studio Runtime Security editor. If you want to add users to an existing group or delete users you must use the FactoryTalk Administration Console or the Users and Groups editor found in the SYSTEM area of the FactoryTalk View Studio application Explorer.

9. Select the [Managers] group in the Account list. Because it is a new group, it has access rights to all of the A-Z security codes. Leave the account rights as they are.

User groups are highly recommended because they allow you to simplify the configuration of security rights. With user groups you can create groups for the different types of users (for example, “maintenance” or “operators”), add a particular user to the appropriate user group, and configure security rights for all users in that group at once, rather than having to configure rights individually.

Because you allowed the Managers group security rights to displays with security code A, both the Engineer and the Supervisor user will be able to access any displays assigned code A.
10. Click the **Close** button to close the runtime security editor. Click **Yes** to save your changes.

When you backup a FactoryTalk View ME project to an .APA file, the users you have created in the FactoryTalk Directory and configured security for are backed up along with the displays and other project data. When you restore the application, you are asked whether you want to restore the FactoryTalk Local Directory. If you answer “Yes”, the backed up FactoryTalk Local Directory is restored to the new FactoryTalk Directory, and overwrites the Local FactoryTalk Directory on the machine. If you answer “No” and do not restore the backed up FactoryTalk Directory to the new PC, you will see that in the Runtime Security editor, the usernames are listed in CLSID format, which is an alphanumeric display. If you have done this, refer to **Answer ID 38817** in the Rockwell Automation technical support database for an explanation and workaround.

**Testing the application on desktop**

1. Click **Test Application** on the **Application** menu to compile and run it.

2. Click on **Recipes** to access this display. Because you are automatically logged in as Default, you should not have security rights to the **3A - Filling Recipes** display that you configured security for. You should see an error saying access is denied because the currently logged in user does not have security access to the display. You can click **Close** to close the error display.

3. Click on **Security**, then **Log In** and enter the credentials of any of the Users that were added to the **Managers** User Group. Click **Close** to close the Security menu when you are finished.

   User: ‘**Engineer**’, Password: ‘1234’

   User: ‘**Supervisor**’, Password: ‘super’

4. Click on **Recipes** to access this display once again and you should now have access since the **Engineer** and **Supervisor** users were given security rights under the **Managers** group.

5. Close the Recipes display and click **Exit** to close the running application when you are done.

Congratulations! You have successfully created a User and User Group, assigned security codes to them and tested the functionality of security in your application.
Improving productivity with parameter passing (30 minutes)

Completing this section requires approximately 30 minutes.

FactoryTalk View ME allows the use of tag placeholders so that you can create one display that can represent a number of similar operations, simply by passing in different tag values.

A tag placeholder is used to mark where you want to insert a tag name or some part of a tag name at run time. A tag placeholder is a crosshatch character (#) followed by a number from 1 to 500.

Tag placeholders are used in FactoryTalk View ME by parameter files. A parameter file is a user-created text file that lists a number of tag placeholders that can be referenced in the project. The parameter file is part of an FactoryTalk View ME project.

FactoryTalk View ME also supports the use of parameter lists. A parameter list is a list of tag placeholders specified for use on a particular display.

Parameter lists can be used in the same components where parameter files can be used:

- The graphic display that opens when an application is first run. Specify the graphic display to open, and the parameter file or parameter list to use with it, in the Startup editor.
- Graphic displays that are opened using Goto Display buttons or Display List Selectors. Specify the graphic display to open, and the parameter file or parameter list to use with it, when you set up the button or selector.
- Global Connections. Parameter files or parameter lists can be used with Remote displays.

Parameters can be carried forward, or passed, to secondary displays that are linked to and open from the initial display, using either a parameter file or a parameter list. When a graphic display opens at run time, FactoryTalk View ME substitutes the string specified in the parameter file or the tag(s) specified in the parameter list for the tag placeholders in the graphic display. The same set of tag placeholders can then be passed on to a secondary display that is opened from the initial display.

In this section you will:

- Look at how Process Faceplates can take advantage of parameter lists to carry tags forward to subsequent displays.
Open the InstantFizz Advanced application

If you already have the InstantFizz Advanced application open from a previous section, you can move on to the next section, **Add the PIDE faceplates.**

1. To open FactoryTalk View Studio, click **Start > All Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio.** If FactoryTalk View Studio is already open, then select **File > Open Application...** to switch applications.

2. Select the existing **InstantFizz Advanced** application, make sure **English** is selected for the **Language**, and click **Open:**

3. If you are switching applications you will be prompted to close the currently open application, click the **Yes** button to proceed.
Add the PIDE faceplates

You will use the Enhanced PID Process Faceplates supplied by FactoryTalk View ME to work with carry forward parameters. These faceplates must be added to the project.

1. In the application Explorer, right-click the InstantFizz Advanced HMI Server and select Add Process Faceplates:

   ![Add Process Faceplates dialog]

   The Add Process Faceplates dialog is opened.

2. Click Clear All on the Add Process Faceplates dialog:

   ![Clear All on Add Process Faceplates dialog]

3. Click the Enhanced PID – PIDE check box and click OK.
When faceplates are added to a project, displays and global objects are imported that can be used in the project.

4. The **12 What is new** display in the **InstantFizz Advanced** project contains a Goto display button that is configured to open the **Logix_PIDE** display on the startup. Double-click the **12 What is new** display in the application Explorer under Displays to open it:

![Display Explorer](image)

5. On the display, double-click the **Process Faceplates with Carry Forward Parameters** Goto display button (as shown above). The button’s property dialog is opened. Note the **Display** and **Parameter list** properties:

![Property Dialog](image)

The display specified by the **Display** property is the display opened when the Goto display button is clicked at run time. In this case it is the **Logix_PIDE** display.
The value for the Parameter list property is:

`{::*[PLC1]Program:Fermenter_Temp.Tank_PIDE01}`

This value is a tag name that is “passed” to the Logix_PIDE display when it is opened. Passed tags can be referenced on the display.

Click **OK** to close the property dialog.

**Testing the PIDE faceplates**

Now you will run the application in test mode and observe how parameter passing works with the PIDE faceplates. First you must set the initial display graphic.

1. Double-click **Startup** in the application Explorer to open the startup utility.

2. For the Initial graphic, select **12 What is new** and click **OK**.

3. On the Application menu, click **Test Application**.

Verify that only **English** is checked to include in the runtime application, and that **English** is selected as the initial run time language.
4. Click **Finish** to create the runtime application. The runtime is created, then the application starts using the initial display you selected:

![FactoryTalk View Machine Edition](image)

5. Click the **Process Faceplates with Carry Forward Parameters** button to open the Logix_PIDE display.

6. Click the **Operator** button to put the PIDE into **Operator** mode. When it is in **Operator** mode, the button will appear disabled (grayed out).

![Operator Mode](image)
7. Click the **SP** (Set Point) numeric input, as shown above. The numeric input keypad is displayed:

![Numeric Input Keypad](image)

The keypad can be configured to include helpful information like the tag name and its minimum and maximum value. We will do this in the next section.

8. Click the **ESC** (Escape) key to close the keypad.

**Investigate the numeric input configuration**

You will now look at how the Process Faceplates are configured to use the parameter passing, numeric input keypad and global objects.

1. Click the **FactoryTalk View Studio – Machine Edition** item in the Windows taskbar to bring Studio to the front (you can leave the application running).
2. When you import Process Faceplates, both Global Objects and Displays are brought into the project. The numeric input object is configured on the Logix_PIDE Global Object.

In the application Explorer tree, double-click on Logix_PIDE under Global Objects.

3. Double-click on the SP numeric input object to open its properties. You may need to double-click a few times to get to the numeric input.

4. On the Numeric tab, click Insert Variable > String button under the Caption on keypad/scratchpad field:
5. Set the values on this dialog to those shown below. To configure the numeric input keypad caption to include the tag name of the passed parameter, you use the **Literal string** option. In this example, we will fix the number of characters to **11**, and choose the option **Right-most characters in string are displayed**, since the full tag name is too long to fit on the numeric input keypad.

![String Variable dialog](image)

Click **OK** to close the dialog.

6. On the **Numeric** tab you can also configure how the numeric input keypad displays the minimum and maximum for the tag value you are entering. You can either enter a value for minimum and maximum, or use a variable minimum and maximum. In this case, we want to use the variable minimum and maximum. Select the **Use variable minimum/maximum** checkbox to do so.

![Numeric Input Cursor Point Properties](image)

7. When using a variable minimum and maximum, you need to specify the tags to be used. Open the **Connections** tab.

8. Double-click in the **Tag / Expression** field for the **Minimum** property and enter ‘`{#1.PVEUMin}`’.

9. Now double-click in the **Tag / Expression** field for the **Maximum** property and enter ‘`{#1.PVEUMax}`’.
Your **Connections** tab should look like this:

![Connections Tab](image)

10. Click **OK** to close the numeric input properties dialog.

11. Close the PIDE global object display, save the changes.

**Investigate “carry forward” parameters**

You will now look at how parameter lists and parameter files support “carry forward” parameters.

1. Click the **FactoryTalk View ME Soda Demo**... item in the Windows taskbar to bring the running application to the front.

![Start Menu](image)

2. On the **Logix PIDE** faceplate, click the **Trend** icon as shown below to open another display, **Logix_PIDE_Trend**.

![Logix PIDE Trend](image)
When the Trend faceplate is opened, you may see a diagnostic error message. If you do, click the **Close [F5]** button. The display should show valid data.

3. Now open a secondary Ontop display from the **Trend (Logix_PIDE_Trend)** faceplate. Click the **AutoTune** button as shown below:

When the AutoTune faceplate is opened, note the errors shown in the diagnostic window, and that the AutoTune display does not have valid data.
4. Click the **Shutdown Demo** button to close the test application.

---

**Fixing the PIDE faceplate errors**

You will now explore the PIDE faceplates and determine why you are getting errors at run time. You will look at how parameters are passed from one display to secondary faceplates or displays and fix any problems you find.

1. Expand the Diagnostic list in **FactoryTalk View Studio** so you can see a number of entries, as shown below:

   ![Diagnostic List](image)

   Scroll down the list to see the errors that are there. Error messages display this icon: ☒.

   Note that most of the errors say that an “Expression contains a tag that has an error value.”, and display the expression with the error. You might notice that all of the expressions listed reference the #2 tag placeholder. This most likely indicates that there is a problem with the second tag reference being passed somewhere between the initial display and the AutoTune display. In the next steps you will look at the parameter passing configuration and try to find the problem.
2. The last diagnostic errors you saw were generated on the AutoTune, or Logix_PIDE_AutoTune display. This display is generated from a Global Object imported with the PID faceplates, so you will need to explore how the faceplate Global Object gets its data. Double-click Logix_PIDE_AutoTune under Global Objects in the application Explorer:

The Logix_PIDE_AutoTune Global Object is opened:
3. You can determine what tags are required on a display by using a few keyboard shortcuts. Press **CTRL + A** to select all components on the display. Press **Ctrl + R** to open the Tag Substitution dialog:

The Tag Substitution dialog is primarily used to make substitutions for tag references on a display. We are using it here because it shows all tags referenced in the display. Look at the tags listed in the **Search for** list, and note that two tag placeholders (#1 and #2) are referenced. This means that the display is expecting to get at least two tag placeholders passed to it when it is opened.

4. Click **Close** to close the tag Substitution dialog.

5. The Logix_PIDE_AutoTune display is opened from a Goto Display button on the Logix_PIDE_Trend display. This display is again generated from a Global Object, so double-click on Logix_PIDE_Trend under Global Objects in the application Explorer:
The Logix_PIDE_Trend global object display is opened:

6. Right-click on the Global Object display and select **Object Explorer**.

7. The View Studio Object Explorer is opened. Find and double-click **AutoTune_GotoDisplayButton** to open the Properties dialog for that object.
8. The Parameter list should include tag placeholders #1 and #2. Close the display.

These are the tag placeholders required by the AutoTune display, so this Goto Display button is configured properly.

9. The Logix_PIDE_Trend display is opened from a Goto Display button on the Logix_PIDE display. Again the Logix_PIDE is generated from a Global Object, so double-click on Logix_PIDE under Global Objects in the application Explorer.
10. When the global object display opens, right-click on it and select **Object Explorer** (if it is not already open). Double-click on the **Trend_GotoDisplayButton** object in the Object Explorer to open the Properties dialog for the button.

11. The Parameter list should have two tag placeholders, #1 and #2. Close the display.

12. Finally, the **Logix_PIDE** display is opened from a Goto Display button on the **12 What is new** display. Double-click **12 What is new** under Displays in the application Explorer to open it.
13. On the **What is new** display, double-click the **Process faceplates with Carry Forward Parameters** Goto Display button to open its Properties dialog.

Note: The **Automated Global Connection Features Demo** is not part of this lab, and therefore a display is not associated with the button.

14. The property we are interested in is the **Parameter list**. The property value is too long to see in the space provided, so click in the box to place the cursor there and use the right arrow key to scroll. Note that there is only one tag in the list!

The first tag in a parameter list will populate the #1 tag placeholder in a secondary display, the second the #2 tag, and so on. Although the **Logix_PIDE** and **Logix_PIDE_Trend** displays do not require a second placeholder tag, **Logix_PIDE_AutoTune** references placeholder #2. Therefore this placeholder must be defined in the “base” display, the display that initializes the passing of parameters. To fix the errors you were seeing from the AutoTune display, you must add a tag reference that will be passed through as the #2 tag placeholder.
15. Place the cursor at the end of the **Parameter list** property, then click the browse button to select a tag to pass through to the #2 tag placeholder. Browse to and select the online tag **PIDE01_Tune** in **Program:Fermenter_Temp**. (The tag is here: InstantFizz Advanced -> PLC1 -> Online -> Program: Fermenter_Temp -> PIDE01_Tune.)

Click **OK** to add the structure tag to the Parameter list.

![Tag Browser](image)

Click **OK** to close the Goto button properties dialog.

16. Close all open displays and global object displays. Save your changes if you are prompted to do so.
Testing the Application

1. Click *Test Application* on the *Application* menu.
   Make sure *English* is selected as the only language in the run time application, and *English* is selected as the initial run time language. Click *Finish* to compile and start the project.
   When the project starts, click the *Process Faceplates with Carry Forward Parameters* button on the initial display.

2. Now navigate through the nested displays to the *Logix_PIDE_AutoTune* display. On the *Tank_PIDE01* display, click the *Trend* button to open the *Trend* display:

   ![Trend Display](image1)

   Note that there are no errors when opening the Trend display.

3. On the *Trend* display, click the *AutoTune* button to open the *AutoTune* display:

   ![AutoTune Display](image2)

   The display should open with no errors! Your troubleshooting was successful. Feel free to navigate through other displays.
   Click the *Shutdown Demo* button to stop the test application:

   ![Shutdown Demo Button](image3)

   Congratulations! You have successfully explored the use of parameter files and learned how to carry tags forward using this functionality.
Logging data (20 minutes)

Completing this section requires approximately 20 minutes.

A commonly required feature in FactoryTalk View ME is the storage of application data. FactoryTalk View ME provides two choices for storing data, the Data Model and the DataStorePlus ActiveX control. The following table highlights the differences between the two:

<table>
<thead>
<tr>
<th>Data Model</th>
<th>DataStorePlus ActiveX control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native/built-in control</td>
<td>Add-on feature</td>
</tr>
<tr>
<td>Timed, data change, data % change triggers</td>
<td>Timed or tag trigger logging</td>
</tr>
<tr>
<td>Always active/running</td>
<td>Can control logging (on/off)</td>
</tr>
<tr>
<td>Only 1 data log model supported at run time</td>
<td>Multiple data log support</td>
</tr>
<tr>
<td>1 file only, first in-first out after maximum data points</td>
<td>Multiple file support</td>
</tr>
<tr>
<td>300,000 data point maximum</td>
<td>Only storage limit to data points</td>
</tr>
<tr>
<td>Binary proprietary file</td>
<td>Plain text .CSV file</td>
</tr>
<tr>
<td>Works with Trend display to show historical data</td>
<td>Does not work with trends</td>
</tr>
<tr>
<td>Supported in all PVP terminals</td>
<td>Requires PVP CE (700-1500) or any PVP v6.0</td>
</tr>
<tr>
<td>Good for Trending historical data and native functionality. Easy setup.</td>
<td>Excellent for plain text or high speed data archiving.</td>
</tr>
</tbody>
</table>

In this section, you will:

- Use the Data Model storage method to log product counts for the InstantFizz Advanced application
- Configure the logging of data for every one second, add the required tags to the data log model and add it to the project startup properties.
Open the InstantFizz Advanced application

If you already have the InstantFizz Advanced application open from a previous section, you can move on to the next section, Creating the Data Log Model.

1. To open FactoryTalk View Studio, click Start > All Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio. If FactoryTalk View Studio is already open, then select File > Open Application... to switch applications.

2. Select the existing InstantFizz Advanced application, make sure English is selected for the Language, and click Open:

3. If you are switching applications you will be prompted to close the currently open application, click the Yes button to proceed.
Creating the Data Log Model

1. Right-click the Data Log Models option located in the Data Log System container to open the dialog and select New.

   Data Log Models allow for historical trending. Multiple Data Log Models can be defined at design time. Only 1 Data Log Model can be enabled at runtime. The model will not log data for string tags, array tags, parameters or expressions, and you cannot use tag placeholders in your data log model. The Data Log file is saved in a format designed for runtime performance and file size reduction. The file cannot be saved/exported to CSV or other file format at runtime.

2. On the Setup tab, type 'Product Counts' in the Description field and enter '3600' in the Maximum data points field.
In this lab you will be logging 3 tags at a rate of 1 second, as a result a **Maximum data points** setting of 3600 will allow us to log approximately 20 minutes of historical data. 3600 data points / 3 data points every second = 1200 seconds = 20 minutes.

Note that the Description field is for your reference only and is not used by Machine Edition.

The Maximum data points are the total number of tag values to store. When this number is reached, the oldest tag values are deleted to make room for new values. The minimum is 100; the maximum is 300,000. The default is 1000. The higher you set the limit, the more space the data log file requires on the runtime computer, or, if you are logging to a custom path on a remote computer, on the remote computer. The data log file is created at startup of the application. 300,000 points will use approximately 4.7MB of space.

3. Select the **Paths** tab, here you can define where the Data Log is stored. Leave the path set to the default location.

If logging to \Storage Card2\Logs and the SD card is removed Data Logging will stop. Once the data log starts it cannot be stopped during runtime by the operator.

Machine Edition does not support dynamic log file creation during runtime.

Best practices for a Panel View Plus 400-600 terminals is to data log to an external CF card or remotely.

If using a network location the network must have domain name resolution, such as a DNS server, to resolve the UNC computer name defined in your path, you cannot use the IP address of a network location.
4. Select the **Log Triggers** tab; here you can define the method used to trigger the logging of data. Enter ‘1’ in the Interval field as shown below, leave the default at Seconds.

**Periodic:**
The fastest log rate possible is 100 ms however logging too fast can cause performance related issues in most cases this rate is not feasible with a typical application due to other loads on communications.
Best practice is logging at the slowest possible rate desired by the process being monitored.

**On Change:**
Use the On Change trigger to log tag values for any tag whose value has changed by a certain percentage. A Change Percentage of 0 will log all changes for a tag.
The Maximum update rate is used to specify the maximum rate at which data servers will send data to the tags in the data log model.
The Change percentage is the percentage a tag value has to change in order to trigger logging. To log all changes, enter 0.

**Heartbeat:**
Type a time and select a time unit to specify how often tag values are logged even if no change has occurred. The heartbeat is unique to each tag in the Data Log Model according to when the tag last changed value.

5. Select the **Tags in Model** tab, here you will define which tags will be included in the Data Log Model.
The maximum number of tags allowed in a model is 100
The model will not log data for string tags, array tags, parameters or expressions, and
you cannot use tag placeholders in your data log model.
Tags can be either HMI or Direct Reference

6. Click the Tag browse button to open the tag browser.

7. Expand the PLC1 shortcut, expand the Online, then ProdCounts folder. Use the mouse + Ctrl key to select the
Good, Scrap and Total tags.
8. Click Add Tags(s) to List, the 3 tags will now appear in the Selected tag(s) window as shown below.

9. Click OK to close the tag browser dialog.

At this point all 3 tags have been selected but have not yet been added to the Data Log Model.
10. Click **Add** to add the 3 tags to the Data Log Model. You should have 3 tags in the Data Log Model as shown below.

11. Click **Close** to save the Data Log Model changes.

12. Click **Yes** when prompted to save changes.

13. Enter ‘Product Counts’ in the **Component name** field when prompted.

14. Click **OK** to save the Data Log Model.
Configure application to start the Data Log Model on startup

1. Double-click the **Startup** option located in the System container to open the dialog.

2. Check the **Data Logging** checkbox. Since **Product Counts** is the only data log configured, it is automatically selected.

3. Check the **Initial graphic** checkbox if it isn’t selected already and ensure that **1 Overview** is selected as the Initial Graphic.

4. Click **OK** to save the Startup changes.
Examine the Trend object properties

1. Expand the Displays folder in the Project Explorer window and double-click on 4A Labeling – Trend to open the graphic.

2. Double-click on the Trend object to open the Trend Object Properties dialog and select the Connections tab. Here you will see that the same 3 tags that you selected for the Data Log Model in the previous section have been configured in the Trend.

3. Select the Pens tab. The pens from the Connections tab have been linked in this list and there currently is no Data Log Model defined so you will only see real time data in the Trend object. The data will start trending when you open the graphic display and will restart every time you open the graphic display containing the Trend object.
4. Use the drop down menu to select the **Product Counts** data log model so that the **Trend object** will now display historical data.

The Trend object doesn’t support dynamically making pens visible or invisible. Note that each pen has a default color, these can be changed. The default width of 1 should not be changed as performance issues can result. Only one Data Log Model can be loaded and available at runtime. The data log will read the data log model only once when opening a display with a Trend object. 300,000 point data logs can take several seconds to load data and allow the operator to interact with the Trend object. This will occur each time a display is opened that is accessing the Data Log.
4. Select the **X-Axis** tab.

Notice the default value of 2 in the Time Span field. Remember that the Data Log Model has been configured to log approximately 20 minutes of data so you will leave the default at 2 minutes to show some of the Data Logging capabilities within the timeframe of the lab.

Setting the X-Axis time span too high (hours, days) will also leave a severe memory footprint. The trend display should never be left open if the X-Axis is configured for hours or days because the HMI will run out of available memory.

5. Select the **General** tab. You will use the default Refresh Rate of 1 second.
Best practice is to set the Trend Refresh Rate no faster than the amount of time it takes to put a data point on a trend.

**Formula:** Fastest Possible Refresh Rate = (X-Axis Time (seconds)) / (Width of Trend Object in pixels)

6. Select the **Display** tab. Highlight the value of 200 in the **Buffer for extra data** field and enter the value ‘0’.

Note that in this lab we are using a Data Log Model to buffer historical data so we do not need a buffer. If this was a real time trend with no Data Log model and we did not define any buffer you would not be able to scroll back in time and see historical data. Once you attempted to scroll back in time or ahead in time all data on the screen would be lost. Using a Data Log Model also allows for data to be seen immediately when the Trend display is opened, depending on how long the application has been running. Without a Data Log Model you will have to wait for real time data to fill the Trend display. If the X-Axis time span is a long time span you will have to wait that long to see how the data is trending.

Trending memory is consumed over time and not allocated on startup of the application. Buffer extra data for Real Time Trending only.

- Max buffer size 32767 records.
- Buffering too much extra data will leave a severe memory footprint.
- If the application uses a data log you don’t need to buffer extra data (Data log becomes the buffer).
- The Trend object must be linked to the data log file for Historical Trending.
7. Click OK to the Trend Object Properties dialog.

8. Close the display and choose Yes when prompted to save changes.

**Testing the application**

1. Click Create Runtime Application on the Application menu to create the run time .MER file.

2. Make sure the file is named InstantFizz Advanced.mer and the Runtime 6.10 application type is selected, then click Save.

   Click Yes if asked if you want to overwrite the existing file. Select English as the initial runtime language and click Finish.

   The run time file is created. It will take a minute or two to compile the application and generate the .MER file.

3. When it is finished, click the Transfer Utility button on the FactoryTalk View Studio toolbar:
4. Click the browse button for the **Source file** and select: \Users\Public\Documents\RSView Enterprise\ME\Runtime\InstantFizz Advanced.mer.

5. In the bottom pane, select your PanelView Plus terminal as the destination. The **Transfer Utility** should look something like this:

![Transfer Utility](image)

Your terminal may look slightly different – look for the PanelView Plus terminal with the IP address of 192.168.1.20.

6. Click **Download** to download the project to the terminal. If you are asked if you want to overwrite the destination file, click **Yes**. You will see a progress bar, and when the download is complete a message box will tell you the download completed successfully. Click **OK**.

7. Click **Exit** to close the Transfer Utility.

8. Move over to your PanelView Plus terminal. If **FTView ME Station** is not already running, start it by double-tapping the icon on the desktop:

![FTView ME Station](image)

9. Tap **Load Application [F1]**. Select **InstantFizz Advanced.mer** in the file list, then tap **Load [F2]**.
10. Tap **Yes** to replace the communications setup. When the application is loaded successfully, you will see the file name in the **Current application** box. Tap **Run Application [F2]** to run your project.

11. Take about 1 minute and **navigate** throughout the application **without** viewing the Trend display. This will allow some time for data to be stored in the data logs.

12. Now, switch to the Trend display using the **Trend** button on the navigation bar across the bottom of your display.

   The data log model started running once the project had started. Since we are using historical data from the data logs, this data is populated in the trend object every time you view this display.

   Congratulations! You have successfully created a data log model and used a trend object to view historical data.
Additional Information A: How does data logging work?

The datalogging that is used here is the DataStore Plus ActiveX control which must be loaded all the time in order for it to log data. Because we do not want it to be visible we place the OnTop Cannot Be Replaced display with the ActiveX at X-Y coordinates that place it off the visible screen on the PanelView Plus 6 terminal.

**99 Data Logging** is the Display that hosts the DataStore Plus ActiveX and also the email ActiveX. The Position of the Display is set to -250 pixels X and -250 pixels Y axis, this moves this screen outside of the visible area. The Display is configured as OnTop Cannot Be Replaced so it will always be open and running the ActiveX control.

Display **99 Data Logging** is configured as the **Initial graphic** for the application. The Display is loaded first when the application starts however the operator never sees it.
The **99 Data Logging** Display is configured with a **Startup** macro that sets the value of the Global Connections Remote Display Number to 1 which then calls Display **1 Main** which is the first Display the operator will see.
Note that this behavior changes slightly when Scenario 2 using the ME_IPInfo active is completed in the lab.
Additional Information B: Differences in Data Models

A commonly required feature in FTView ME is the storage of application data. FTView ME provides two choices for storing data, the Data Model and the DataStorePlus ActiveX control. The following table highlights the differences between the two:

<table>
<thead>
<tr>
<th>Data Model</th>
<th>DataStorePlus ActiveX control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native/built-in control</td>
<td>Add-on feature</td>
</tr>
<tr>
<td>Timed, data change, data % change triggers</td>
<td>Timed or tag trigger logging</td>
</tr>
<tr>
<td>Always active/running</td>
<td>Can control logging (on/off)</td>
</tr>
<tr>
<td>Only 1 data log model supported at run time</td>
<td>Multiple data log support</td>
</tr>
<tr>
<td>1 file only, first in-first out after maximum data points</td>
<td>Multiple file support</td>
</tr>
<tr>
<td>300,000 data point maximum</td>
<td>Only storage limit to data points</td>
</tr>
<tr>
<td>Binary proprietary file</td>
<td>Plain text .CSV file</td>
</tr>
<tr>
<td>Works with Trend display to show historical data</td>
<td>Does not work with trends</td>
</tr>
<tr>
<td>Supported in all PVP terminals</td>
<td>Requires PVP CE (700-1500), with exception of PVP v6.0</td>
</tr>
<tr>
<td>Good for Trending historical data and native functionality. Easy setup.</td>
<td>Excellent for plain text or high speed data archiving.</td>
</tr>
</tbody>
</table>

The Data Model data storage method is covered in-depth in the FTView ME Fundamentals lab. In this section you will work with the DataStorePlus ActiveX control.

The DataStorePlus ActiveX control can be used to log tag data to a file and store the file dynamically. It is a great utility for logging vital information from temperature tags, pumps, valves, and other items for later historical analysis. The DataStorePlus ActiveX control provides functionality very similar to that in FTView 32 and the SE data log model, such as:

- Allowing users to specify when new files are created and old files are deleted
- Optimization of logging to log assigned connections only
- Overwriting the oldest data in a single file when the file has reached a user-specified size limit
- Automatically adding a date and time stamp at the beginning of each record
- Allowing users to change the headers of each column in the file
- Logging data periodically at a user-specified interval

The DataStorePlus ActiveX control has the same functionality on both the Windows 2000/XP and WinCE platforms.

For more information on the DataStorePlus ActiveX control, please refer to answer ID 35155 in the Rockwell Automation Knowledgebase.
Additional Information C: Application Documenter

In the first scenario you don’t know if the tag Maintenance\pdf1 is used in the application. You could go through all the screens and check however, there is a tool available called Application Documenter. The Application Documenter is a stand-alone utility that provides detailed information on FactoryTalk applications. This section assumes that the runtime file Block_Machine_Advanced.mer was restored.

1. Navigate to Start > Program Files > Rockwell Software > FactoryTalk View > Tools and select Application Documenter.

2. When prompted, select Machine edition and then select your application Block_Machine_Advanced.

3. Select English as the language and click OK.

4. On the left Panel, click on + sign before the Tag Database to expand it and Check Tags Cross Reference

5. You will see that the tool is going through all the screens and generating a cross reference to the HMI tags. Once it’s finished, you will see a screen where on the top you see all the tags that are used in screen and on the bottom part you see all the unused tags. Your tag Maintenance\pdf1 is in the Unused Tag Info section.
6. Select **File > Exit** to close the Application Documenter.

With the **Application Documenter** you could also do a cross reference of the Direct Reference tags or controller tags.
Additional Information D: Studio overview

Exploring FactoryTalk View Studio for Machine Edition Interface

The FactoryTalk View Studio for Machine Edition Application Windows is divided into several key elements:

**Application Menu**
- Used to interact with the application;
- Open/Close/Create new applications;
- Import/Export information.
- The menu changes context based on what project object is open in the **Work Pane**.

**Explorer Pane**
- Contains all objects related to an application project.
- Application objects are opened in the **Work Pane**.
- The Explorer’s content is described in more detail on the next page.

**Diagnostic List**
- Contains status and error messages related to system, application and project.

**Work Pane**
- Open project object contents are display in this area.
- In this example, an untitled display has added to the application.

**Graphics Toolbar**
- Provides easy access to tools that are used to manipulate objects on a display.
- Ex. Rotate, Group, Ugroup, etc.

**Objects Toolbar**
- Provides easy access to objects that are used on displays to create the user interaction.
- Ex. Numeric Input, String Display, Ramp button, etc.
Now let us take a close look at the objects in the Explorer Pane.

- **Local (CORE)** contains computer-scoped components, FactoryTalk View Data Server and FactoryTalk Services Platform objects.

- **L15** contains application-scoped components like the FactoryTalk View HMI Server and RSLinx Enterprise configurations. The FactoryTalk View Data Server object is used to add additional data servers (Ex. OPC) to an application project. This is an advanced operation and not covered in the manual.

- **System** contains all information scoped to the HMI server such as, Project Settings, HMI Tags, Graphic files, Alarms, Information Messages, Macros, Data Logs, and Recipes.

- **HMI Tags** contains all tags resident in the memory of the HMI Server. Applications for Logix Controllers use direct tag referencing eliminating the need to create HMI tags to communicate with the PLC.

- **Graphics** contains all graphic images used in the application as displays, or contained within displays and parameter files. Parameter files are one way to enable graphic reuse between applications.

  Symbol Factory provides access to thousands of stock images organized in an easy to use library.

- **Alarms** contains the application’s alarm configuration including triggers and messages.

- **Information** contains the application’s information message configuration.

- **Logic and Control** contains the application’s macros.

- **Data Log** contains any Data Log Model configured for the application.

- **RecipePlus** contains the Recipe management system configuration and any configured recipes.

- **RSLinx Enterprise** contains communication shortcuts used by the application.

- **System** contains FactoryTalk Services Platform related configuration. This is primarily used to manage users and groups in a FactoryTalk Machine Edition application.
Lab VZ02 – FactoryTalk View Machine Edition and PanelView Plus Advanced Lab
**Lab Setup and Configuration Information**

### Lab Information

<table>
<thead>
<tr>
<th>Lab Name</th>
<th>VZ02 - FactoryTalk View Machine Edition and PanelView Plus Advanced Lab</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab Description</td>
<td>This hands-on lab focuses on advanced FactoryTalk Machine Edition concepts for experienced users. Topics include data collection, trending, and the Program Launcher and Email ActiveX controls. FactoryTalk ViewPoint for FactoryTalk View Machine Edition v2.0, including tag write functionality, is also covered.</td>
</tr>
<tr>
<td>Lab Creator</td>
<td>Bob Syms</td>
</tr>
<tr>
<td>Date Created</td>
<td>02/27/2012</td>
</tr>
<tr>
<td>Updates:</td>
<td>04/18/2012 Wil Mattheis</td>
</tr>
</tbody>
</table>

### Hardware Configuration per Student

<table>
<thead>
<tr>
<th>Qty</th>
<th>Demo Cat.# / Description</th>
<th>Communication</th>
<th>Location</th>
<th>Firmware</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Computer workstation consisting of</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2 GHz CPU</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4 GB RAM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Hard drive. Minimum 40GB capacity</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>USB Keyboard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>USB Mouse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Ethernet adapter.</td>
<td>Configured for DHCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Ethernet adapter.</td>
<td>IP Address: 192.168.1.1, Subnet: 255.255.255.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Video adapter. Minimum resolution: 1280 x 1024 pixels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>LCD Display. Minimum resolution: 1280 x 1024 pixels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Ethernet CAT5E cross-over cable 5ft (used with PanelView Plus terminal)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Ethernet CAT5E cable 5ft (used with Event classroom network)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>PanelView Plus 6 1250</td>
<td>IP Address: 192.168.1.20, Subnet 255.255.255.0</td>
<td></td>
<td>Firmware revision must support ViewPoint 2.10</td>
</tr>
</tbody>
</table>
Computer/Host Settings
Location

Files
Varies by machine
DHCP – connected to classroom network & internet
Host computer TCP/IP setting: DHCP
VMware image TCP/IP setting:
IP Addr: 192.168.1.1
Subnet: 255.255.255.0
Windows 7 Professional

Computer Name
IP Address (NIC 1)
IP Address (NIC 2)

Operating System

Basic Setup Diagram

Student Station
PanelView Plus 6 1250

Second NIC set for DHCP and
connected to the Internet

High level Tank 02
ON

ON

Level H234

PLANT J45

Temp.:
123deg
Press.:
12Pa
Level:
4,5m

Tank
02

Start/Stop
Auto
Manual

ON
ON
ON
OFF
ON
ON

P
P
P
P
P
P

Na
H2O
Ca
Li
Pb
Cu

Force
345
34
LL

Setpoint
Output
L

H

HH

React. A
Output:

ON

M

Temp.: 123 gr
Speed: 56 m/s

React. B

Tank 01
Tank 02

Waste

React. C

Converter
Tank 03

React. D

Tank 04
React. E

F1
F7

F2
F8

F3
F9

F4
F10

F5
F11

F6
F12

12

102

31

IP Addr: 192.168.1.20
Subnet: 255.255.255.0

CAT5E cross-over
cable

NIC 2
Host computer TCP/IP setting: DHCP
VMWare image TCP/IP setting: IP Addr 192.168.1.1
Subnet 255.255.255.0
NIC 1
Host computer TCP/IP setting: DHCP

CAT5E cable

To Classroom network
DHCP assigned IP

289 of 294


### Application/Programming

<table>
<thead>
<tr>
<th>Location</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Lab Files\ViewME – Advanced\PVP</td>
<td>Copy all files to the root of a USB storage or SD memory card. Then insert into the PanelView Plus 6 terminal to run the batch that will configure and copy all necessary lab files.</td>
</tr>
<tr>
<td>C:\Lab Files\ViewME – Advanced\SoftLogix</td>
<td>InstantFizz.acd (slot 2) ME_Intro.acd (slot 3) Block_Machine_Advanced.acd (slot 6)</td>
</tr>
<tr>
<td>C:\Lab Files\ViewME – Advanced\ViewME</td>
<td>Configure Internet Explorer on the Image computer to open BlockMachine.html as the default web site. Also set Internet Explorer to be every visit to page in the Options.</td>
</tr>
<tr>
<td>C:\Lab Files\ViewME – Advanced\ViewME\Faceplates</td>
<td>Copy all files to C:\Users\Public\Public Documents\RSView Enterprise\ME\Faceplates and overwrite the existing.</td>
</tr>
<tr>
<td>C:\Lab Files\ViewME – Advanced\ViewME\Pick and Choose\ME Projects</td>
<td>Restore this application; InstantFizz Advanced.APA Restore the application, Language_Translate.apa Restore the application ScreenSwitch.apa</td>
</tr>
<tr>
<td>C:\Lab Files\ViewME – Advanced\ViewME\Pick and Choose</td>
<td>Register the ME_EmailSender.ocx on the Desktop computer “regsvr32 ME_EmailSender.ocx”</td>
</tr>
</tbody>
</table>

### Additional Equipment Required per Workstation

<table>
<thead>
<tr>
<th>Qty</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ethernet CAT5E cross-over cable 5ft (used with PanelView Plus terminal)</td>
</tr>
<tr>
<td>1</td>
<td>Ethernet CAT5E cable 5ft (used with Event classroom network)</td>
</tr>
<tr>
<td>1</td>
<td>PanelView Plus 6 1250, NOTE: ftp must be configured and running on the terminal with the default directory configured as blank and NOT \Temp.</td>
</tr>
</tbody>
</table>

### RSLinx - DDE/OPC Topic Configuration

<table>
<thead>
<tr>
<th>Topic Name</th>
<th>Path to Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>
RSLinx - Driver Configuration

<table>
<thead>
<tr>
<th>Topic Name</th>
<th>Path to Hardware</th>
</tr>
</thead>
</table>

Note: Only InstanFizz (Node2) and Block_Machine (Node6) are needed for the Advanced lab.

RSLinx Enterprise - Shortcut Configuration

<table>
<thead>
<tr>
<th>Shortcut Name</th>
<th>Path to Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlockComms</td>
<td>Ethernet, Ethernet &gt; 192.168.1.1 &gt; Backplane &gt; 6, 1789-L60 v19</td>
</tr>
<tr>
<td>PLC1</td>
<td>Ethernet, Ethernet &gt; 192.168.1.1 &gt; Backplane &gt; 2, 1789-L60 v19</td>
</tr>
</tbody>
</table>

Application Versions

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Software</th>
<th>Version</th>
<th>Service Pack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rockwell</td>
<td>FactoryTalk Activation Manager</td>
<td>3.40</td>
<td>CPR9 SR4</td>
</tr>
<tr>
<td>Rockwell</td>
<td>FactoryTalk Diagnostics</td>
<td>2.50.00</td>
<td>CPR9 SR5</td>
</tr>
<tr>
<td>Rockwell</td>
<td>FactoryTalk Service Platform</td>
<td>2.50.00</td>
<td>CPR9 SR5</td>
</tr>
<tr>
<td>Rockwell</td>
<td>FactoryTalk View Machine Edition</td>
<td>6.10.00</td>
<td>CPR9 SR4</td>
</tr>
<tr>
<td>Rockwell</td>
<td>FactoryTalk ViewPoint ME</td>
<td>2.10.07</td>
<td>CPR9 SR4</td>
</tr>
<tr>
<td>Rockwell</td>
<td>RSLinx Enterprise</td>
<td>5.40.00000</td>
<td>CPR9 SR4</td>
</tr>
<tr>
<td>Rockwell</td>
<td>RSLinx Classic</td>
<td>2.59.00</td>
<td>CPR9 SR5</td>
</tr>
<tr>
<td>Rockwell</td>
<td>RSLogix 5000</td>
<td>19.01.00</td>
<td>CPR9 SR3</td>
</tr>
<tr>
<td>Rockwell</td>
<td>SoftLogix 5800</td>
<td>19.01.00</td>
<td></td>
</tr>
<tr>
<td>Rockwell</td>
<td>Silverlight</td>
<td>4.0.60129.0</td>
<td></td>
</tr>
<tr>
<td>Rockwell</td>
<td>.NET Framework</td>
<td>1.1, 2.0 SP2, 3.0 SP2, 3.5 SP1</td>
<td></td>
</tr>
<tr>
<td>Microsoft</td>
<td>Internet Explorer</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Rockwell</td>
<td>PanelView Plus 6</td>
<td>6.10.17 or greater</td>
<td>O/S 2.0 release</td>
</tr>
<tr>
<td>Rockwell</td>
<td>Translation Utility</td>
<td>1.3.3</td>
<td></td>
</tr>
</tbody>
</table>
Required Pre-Lab Configuration

Configure the Virtual Network

1. Select Virtual Network Editor from the Edit menu to open the Virtual Network Editor dialog.

2. The dialog below will appear.
3. Select the Host Virtual Network Mapping tab

4. For VMnet0 select the physical ethernet adapter that will be connected to the PanelView Plus terminal.

5. Click the OK button to commit the changes that have been made.