

Proficiency^{*} Machine Edition

GETTING STARTED

Version 7.50

2012

GFK-1868Q





All rights reserved. No part of this publication may be reproduced in any form or by any electronic or mechanical means, including photocopying and recording, without permission in writing from GE Intelligent Platforms, Inc.

Disclaimer of Warranties and Liability

The information contained in this manual is believed to be accurate and reliable. However, GE Intelligent Platforms, Inc. assumes no responsibilities for any errors, omissions or inaccuracies whatsoever. Without limiting the foregoing, GE Intelligent Platforms, Inc. disclaims any and all warranties, expressed or implied, including the warranty of merchantability and fitness for a particular purpose, with respect to the information contained in this manual and the equipment or software described herein. The entire risk as to the quality and performance of such information, equipment and software, is upon the buyer or user. GE Intelligent Platforms, Inc. shall not be liable for any damages, including special or consequential damages, arising out of the use of such information, equipment and software, even if GE Intelligent Platforms, Inc. has been advised in advance of the possibility of such damages. The use of the information contained in the manual and the software described herein is subject to GE Intelligent Platforms, Inc. standard license agreement, which must be executed by the buyer or user before the use of such information, equipment or software.

Notice

GE Intelligent Platforms, Inc. reserves the right to make improvements to the products described in this publication at any time and without notice.

© 2012 GE Intelligent Platforms, Inc. All rights reserved. * Trademark of GE Intelligent Platforms, Inc. Any other trademarks referenced herein are used solely for purposes of identifying compatibility with the products of GE Intelligent Platforms, Inc.

We want to hear from you. If you have any comments, questions, or suggestions about our documentation, send them to the following email address:

doc.ip@ge.com



Contents

1 Contents	i
2 Welcome	1
System Requirements	2
Installation	4
Product Authorization	5
Hardware Key Authorization	5
Software Key Authorization	6
Contact GE INTELLIGENT PLATFORMS	10
General Contact Information	10
Technical Support	10
Americas	10
Europe, the Middle East, and Africa (EMEA)	10
Asia Pacific	11
3 Proficy* Machine Edition*	13
Quick Start	14
Machine Edition Environment	16
Getting to Know Machine Edition	17
Right-click, right-click, right-click	17
Getting Help	17
Accessing the Right Tool	19
Using docking markers	19
Projects and the Navigator	20
Properties and the Inspector	22
Data Watch Lists	23
Smart Lists	24
The Toolchest	25
The Feedback Zone	26
Managing Variables	27
Machine Edition Projects	29
Sharing Projects between Machine Edition Workstations	29
Running a Sample Project (View/Logic Developer - PC)	30
Developing a Machine Edition Project	32
Validating and Downloading a Project	35
Testing a View Project	37

4 Logic Developer - PC	39
SFC Editor	41
Sequential Function Chart	41
Working with the SFC editor - Offline	42
Working with the SFC editor - Online	44
Ladder Editor	45
Ladder Program	45
Working with the ladder editor - Offline	46
Working with the ladder editor - Online	47
Instruction List Editor	49
Instruction List	49
Working with the IL editor - Offline	50
Working with the IL editor - Online	51
Structured Text Editor	52
Structured Text	52
Working with the ST editor - Offline	53
Working with the ST editor - Online	54
Function Block Diagram Editor	55
Function Block Diagram	55
Working with the FBD editor - Offline	56
Working with the FBD editor - Online	58
Logic Developer - PC Web Access	59
Control I/O Drivers	62
I/O Drivers	62
Control I/O Tool	63
Working with Control I/O	64
Controller	65
Working with the Controller	66
Warm Standby	67
Working with Warm Standby	68
Hot Standby	68
Working with Hot Standby	68
OPC Servers, Warm Standby, and Hot Standby	69
5 View*	71
Panel Editor	72
Working with the Panel Editor	73

- Message Displays 74
- Script Editor 75
 - Scripts 75
 - Scripting Languages 76
 - VBScript Syntax 78
 - Active Scripting Objects 78
 - Working with the Script Editor 78
- Grid Editors 79
- Alarms 80
 - Alarm Groups 80
 - Alarm Display Objects 81
- Logging data 82
- PLC Access I/O 83
 - Drivers 83
- OPC 85
 - OPC Client 86
 - Machine Edition OPC Servers 86
- View Web Access 87
- Languages folder 90
 - Working with the Languages editor 91
- View Runtime 93
 - Networking 93
- QuickPanel Applications 95
 - External Keypad Assignment 95
 - Scripts on QuickPanel Targets 96
 - QPScript Language 96

6 Motion Developer* 99

- About Motion Developer 100
 - Supported motion control devices 100
- Motion Applications 101
 - Motion targets 101
 - Motion configuration 102
 - Motion programs 103
 - Motion blocks 104
- Motion Scripts 105
- Motion wizards 106
- Special Wizards 108

Application Builder flowchart editor	108
Motion Calculator	110
CAM Profile Editor	112

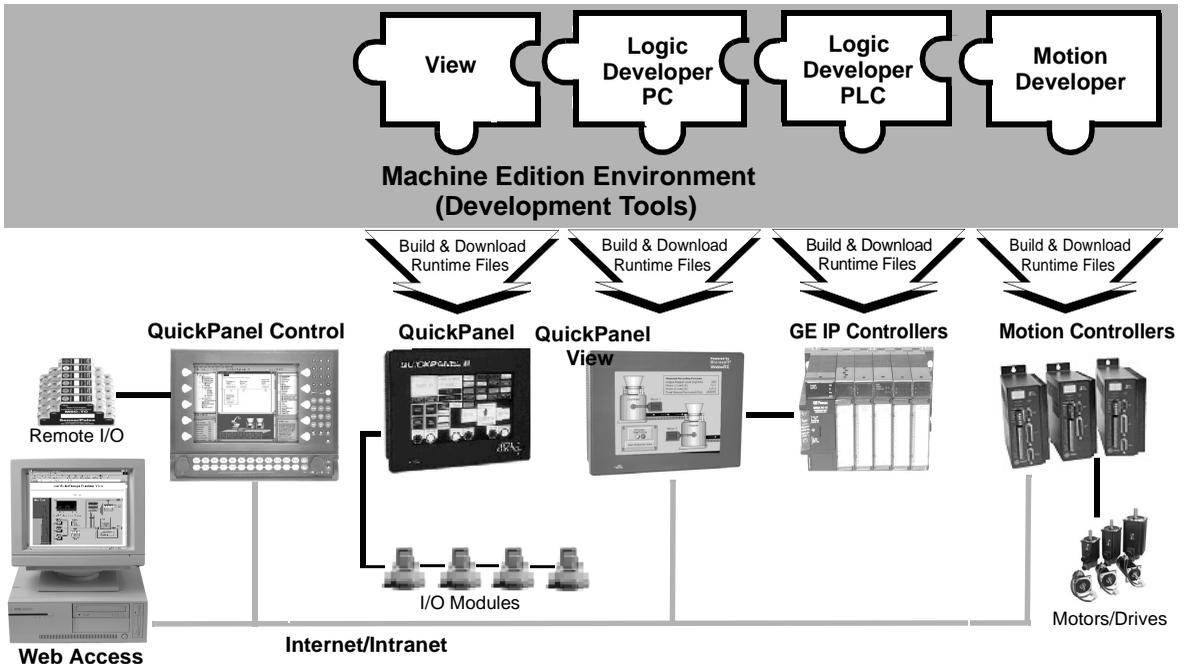
Index	115
--------------	------------

1

Welcome

Congratulations on your purchase of a Proficy* Machine Edition* 7.50 product. This package provides all the tools necessary to create powerful control and HMI (human machine interface) applications for a variety of run-time targets; a truly scalable solution. Machine Edition supports the following products and features:

- A fully-integrated environment. Every tool and editor works with the others.
- Logic Developer - PC*, scalable, computer based control software with a complete set of IEC standard editors (available with QuickPanel Control products).
- Logic Developer - PLC*, a software tool for programming the full line of GE IP Controllers. (See GFK-1918 Getting Started Logic Developer - PLC.)
- View*, for creating HMIs for Windows 7, Windows XP, QuickPanel View/Control, and QuickPanel units.
- Motion Developer*, for developing Whedco motion control applications.
- Web Access features that access real-time HMI data any time, from anywhere.



SYSTEM REQUIREMENTS

To use Machine Edition and its tools, you require the following:

Development Environment Operating System

Machine Edition product	Windows 7 ^a	Windows XP ^b
Logic Developer - PC	yes	yes
View	yes	yes
Logic Developer - PLC	yes ^e	yes
Logic Developer - State		yes
Motion Developer		yes

a. 32-bit or 64-bit variations of the following: Windows 7 Ultimate, Windows 7 Enterprise, or Windows 7 Professional.

 **Want to know more?** In the Help Index, look up “Windows 7: Caveats”.

b. **Supported:** Windows XP Professional SP2 and SP3. You must be part of the PowerUsers group or the Administrators group. **Not supported:** Windows XP Tablet PC Edition and Windows XP Embedded.

c. Windows regional settings must be set to English

Processor speed and memory

- Minimum: Intel Core Duo or AMD Athlon 64 X2 with 4 GB RAM
- Recommended: 2.6 GHz Intel Core i series with 4 GB RAM

Other requirements

- Internet Explorer version 8.0, 7.0, version 6.0 SP1 or later, or version 5.5 SP2 Q810847 or later.

Note: You must install Internet Explorer before installing Machine Edition.

- TCP/IP Network protocol (if you use an Ethernet connection).
- .NET Framework 4.0 Client Profile. If the Microsoft .NET Framework is not yet installed, it is automatically included during the installation of Machine Edition, and a reboot may be required to complete the installation.
- 2 GB hard disk space for all the Machine Edition products and sample projects. Additional space is required for your projects and temporary files.

The Development Environment is supported on VMWare Player 2.0 or greater or VMWare Workstation 6.0 or greater

Note: The APM Motion Programmer does not function if Machine Edition runs within VMWare.

Windows PC Runtime

- If you install only the Runtime for Logic Developer - PC and View Developer, the run-time executables are automatically updated from the development machine when the project is downloaded.
- 32-bit or 64-bit variations of Windows 7 Ultimate, Windows 7 Enterprise, or Windows 7 Professional; Windows XP Professional SP2 or SP3; or Windows XP Embedded.

Notes

- Windows regional settings must be set to English.
- Windows XP Tablet PC Edition are not supported.
- 233 MHz Pentium-based workstation (1.6 GHz recommended)
Note: 300 MHz is required for Windows XP Professional.
- 200 MB free hard disk space.
- 128 MB RAM (1 GB recommended; minimum 256 MB on XP).
- Minimum requirements for the range of variables defined in your projects:

Requirements	Variables		
	<500	500 - 10000	>10000
CPU	233 MHz	1 GHz	2 GHz
Disk Space	200 MB	200 MB	200 MB
RAM	128 MB	512 MB	1 GB

QuickPanel View/Control Runtime

- *For HMI and Logic components:* QuickPanel Control.
- *For HMI only:* QuickPanel View.

INSTALLATION

For last-minute information, release notes, and supported hardware lists for Machine Edition products, see the Important Product Information (IPI) document on the DVD. There are several ways to view this document:

- When installing Machine Edition, select **Important Product Information** on the initial Launcher screen.
- On the Machine Edition Help menu, choose **Important Product Information**.
- When running Machine Edition, click the InfoView tab in the Navigator, then double-click the Important Product Information page under Getting Started in the Table of Contents.
- When running Machine Edition, click the Home button on the InfoViewer toolbar, and click the What's New link under Get Started on the left hand side.

If you have any problems installing Machine Edition, please contact Technical Support (see page 9).

To install the full Machine Edition development environment from a DVD

1. Insert the Machine Edition DVD into your computer.
Windows automatically starts the setup program. If the setup program does not automatically start, run *Setup.exe* in the root directory of the DVD.
2. Click **Install** to start the install process.
3. Follow the instructions that appear.

If a previous version of Machine Edition is installed on your workstation, you will be prompted to uninstall Machine Edition during the upgrade process. You should do so only when asked to by the installation process. Do not delete files left behind during uninstallation; these will be used by the new version.

To install only the View and Logic Developer - PC Run-times from a DVD

1. Insert the Machine Edition DVD into your computer.
Windows automatically starts the setup program. If the setup program does not automatically start, run *Setup.exe* in the root directory of the DVD.
2. Click **Install** to start the install process.
3. Follow the instructions that appear until you reach the Proficy Machine Edition Setup page.
4. On that page, expand the View node. Under that node, select Run-time and deselect everything else. Expand the Logic Developer - PC node. Here too, select Run-time and deselect the rest. At the bottom of the page, click **Next**.
5. Follow the instructions that appear.

PRODUCT AUTHORIZATION

A new installation of Proficy Machine Edition provides a 4-day trial license with full access to Machine Edition features. This license overrides all other licensing and cannot be removed. Any licensing added will be apparent on the fifth day. View Runtime can be run for 2-hour demo periods at any time. Using View OPC drivers with View Runtime on a PC requires a separate View license on the PC.

To continue using Machine Edition beyond the 4-day trial period, you must authorize the software.

Two types of authorization are available: *software key* authorization and *hardware key* authorization. On a single computer, it is not recommended to use both types of authorization.

Hardware Key Authorization

A M4 or MAX USB hardware key contains authorization files that are read by the Proficy Common License Viewer, which enables Proficy products, including one or more Machine Edition products, to execute.

To see what products and target types are authorized by a hardware key

- On the Windows **Start** menu, point to **Programs**, then to **Proficy Common**, and choose **License Viewer**.

To authorize Machine Edition products with a hardware key

1. Find out which version of the Proficy Licensing Software is installed on your computer: On the operating system's **Start** menu, point to **Programs**, then to **Proficy Common**, and choose **License Viewer**. In the top right corner of the License Viewer window that appears, click **About**. Take note of the version number and click **OK**.
2. If this is the first time you install a Proficy product, access <http://support.ge-ip.com>, and in the top right corner, click **Register**. In the Single Sign On page that appears, enter your email address and click **Submit**. When you receive a GE Intelligent Platforms Web Access Registration email, follow the instructions. In the Single Sign On -- SSO Registration application window that appears, fill out the information and click **Submit**.
3. Access http://support.ge-ip.com/support/index?page=securitykey&id=ST15&Iclicked=Licensing_Key_Updates ([login req](#)). Enter your SSO User ID and Password. In the middle of the

page that appears, follow the Updating the Max/M4 or Pro/M1 key procedure, keeping in mind the following:

- Machine Edition does not support the insertion of a hardware key into a parallel port, the insertion of hardware keys in multiple USB ports, or the combination of a hardware key with a software key on the same computer.
- If the web site refers to a more recent version than the one on your computer, uninstall your current version, download the more recent version, and install it.
- The USB port is usually located on the front or back of your tower case, or in the side of a laptop. We do not recommend using a USB port on your monitor.
- If this is a new hardware key, double-click the license file <serialnumber>.plic (supplied by your local distributor). In the wizard that appears, accept all the defaults. The license file will be flashed onto your key.
- The first time you plug the key into a USB port, you see a progress dialog box that indicates that Windows has detected new hardware and is updating its system settings. After the dialog box disappears, the settings stored in the hardware key are active and you can use the products that it authorizes.

You must leave the key in the port while using the products. If you remove the key from the port, authorization for those products is removed and replaced with read-only access to your projects or a 2-hour demo mode for Runtime.

To move the hardware-key authorization to another computer

- Remove the hardware key from the source computer and follow the previous procedure on the destination computer.

After the authorization file <serialnumber>.plic has been flashed to the M4 or MAX hardware key, it can be used on any computer on which the current version of the Proficy Common License Viewer is installed.

Software Key Authorization

When authorizing Machine Edition products with a software key, you need to contact us by telephone, fax, or e-mail. Authorization personnel are available on the telephone between 6 a.m. and 6 p.m. MST.

Software key authorization is specific to a single computer or workstation. If you want to work with Machine Edition on a different workstation, you must move the authorization to that second workstation (see page 8).

To see what products and target types are authorized by a software key

- On the operating system **Start** menu, point to **Programs**, then to **Proficy** (or **GE Fanuc** or **CIMPLICITY Machine Edition**, depending on when Machine Edition was first installed on the computer), then to **Proficy Machine Edition**, and then choose **Product Authorization**.

To authorize a copy of Machine Edition with a software key

1. Have your serial number(s) ready. The serial numbers can be found on the Certificate of Authentication that came with your product.
2. Run the Product Authorization program from the Start menu | Programs | Proficy | Proficy Machine Edition | Product Authorization.
3. In the Product Authorization dialog box that appears, click **Add**.
4. Select the means by which you are authorizing: Internet, Phone/Fax/E-mail or Floppy Disk Transfer. Click **Next**.

If you choose the Internet option, follow the instructions on the web site.

If you choose the Phone/ Fax/ Email option, proceed to step 5.

If you choose the Floppy Disk Transfer option, ensure that you have an authorization removable media to proceed.

5. Fill in the fields in the dialog box. Fields identified with an asterisk (*) must be filled in.

If authorizing online click Submit Authorization once the form is completed.

If authorizing via phone/fax, click the Phone/Fax button once the form is completed and call the number on the screen to receive a new key code(s).

If authorizing via e-mail, click the Send E-mail button once the form is complete.

- **Phone.** Phone the number listed on the screen to receive a new key code(s).
- **Fax.** Click Print FAX and fax the Product Authorization Request to us (our fax number will be on the print out). We will then reply by fax with your new key code(s).
- **Internet.** On the Authorization web page, click Submit Authorization. We will then reply by e-mail with your new key code(s).
- **E-mail.** Click Authorize to e-mail us. We will then reply by e-mail with your new key code(s).

Product Authorization is complete after you enter the new key code and it has been accepted. Depending on the product you have purchased, you may need to run the Product Authorization program a number of times.

To move the software-key authorization to another computer

You can run a Machine Edition development product or Runtime only on the computer on which the authorization process was completed. If you want to develop your projects or execute Runtime on a different computer, you need to complete the following steps to move the authorization from one computer to another. (For a more detailed procedure with screen shots, access www.ge-ip.com/support and look up Knowledge Base article KB5659.)

On the computer that the authorization will be moved to, do as follows:

1. Install Machine Edition.
2. Run the Product Authorization program as follows: on the operating system's **Start** menu, point to **Programs**, then to **Proficy** (or **GE Fanuc** or **CIMPLICITY Machine Edition**, depending on when Machine Edition was first installed on the computer), then to **Proficy Machine Edition**, and then choose **Product Authorization**.
3. In the top right corner of the Product Authorization dialog box that appears, take note of the Site Code.
This code is required to generate the license file on the source computer.
4. Leave the dialog box open.

On the source computer, do as follows:

1. Run the Product Authorization program. Select the authorized products you want to move and click **Move**.
2. In the Before Moving message box that appears, read the note and click **OK**.
3. Enter the Site Code that you wrote down when you were on the destination computer.
If the Site Code is entered incorrectly, the authorization will be lost and cannot be recovered; you will need to contact technical support (see page 10).
4. Click **Next**.
5. In the Confirm dialog box that appears, verify that the site code is correct. If it is, click **OK**.
6. Insert a blank formatted removable medium in the appropriate location.
7. In the refreshed Product Authorization dialog box, select the drive letter that corresponds to the medium you inserted.
Do not select the C:\ drive; otherwise, your source license files will be overwritten.
8. Click **Next**.
9. In the window that appears, confirm the accuracy of the Target Site Code. If it is correct, click **Finish**.

The authorization code is moved to a file named `authoriz.dat` on the removable medium you selected.

10. In the Authorization disk successfully created message box, read the message and click **OK**.

On the destination computer, do as follows:

1. Insert the removable medium.
2. In the Product Authorization window that you left open earlier, click **Add**.
3. In the refreshed Product Authorization dialog box, select **Floppy Disk Transfer** and click **Next**.
4. In the refreshed dialog box, to the right of the Please insert your Authorization Disk field, click the button.
5. In the dialog box that appears, select the drive letter that corresponds to the medium you inserted and navigate to the location of the `authoriz.dat` file. Click **Open**.
6. In the Product Authorization dialog box, click **Next**.
7. In the refreshed dialog box, verify the product and contact information. If all is correct, click **Finish**.
8. In the message box that indicates that the move was successful, click **OK**.
The authorization has now been moved to the new computer, as you can see in the refreshed Product Authorization window.
9. Click **Exit**.

To change a software key to a hardware key or vice-versa

- Contact your local distributor.

CONTACT GE INTELLIGENT PLATFORMS

If you purchased this product through a GE Intelligent Platforms Authorized Channel Partner, please contact the seller directly.

General Contact Information

Online Technical Support & GlobalCare: www.ge-ip.com/support

Comments about our manuals or online help: doc.ip@ge.com

Additional information: www.ge-ip.com

Solution Provider: solutionprovider.ip@ge.com

Authorization: authorization.ip@ge.com

Technical Support

If you have technical problems that cannot be resolved with the information in this guide, please contact us by telephone or email, or on the web at www.ge-ip.com/support.

Americas

Online Technical Support: www.ge-ip.com/support

Phone: 1-800-433-2682

International Americas Direct Dial: 1-434-978-5100

Technical Support Email: support.ip@ge.com

Customer Care Email: customercare.ip@ge.com

Inside Sales: insidesales.ip@ge.com

Primary language of support: English

Europe, the Middle East, and Africa (EMEA)

Online Technical Support: www.ge-ip.com/support

Phone: +1 800 433-2682 or +1 780-401-7717 if the 800 option is not available in your country, or if dialing from a mobile telephone

Technical Support Email: support.emea.ip@ge.com

Customer Care Email: customercare.emea.ip@ge.com

Inside Sales: insidesales.emea.ip@ge.com

Primary languages of support: English, French, German, Italian, Czech, Spanish

Asia Pacific

Online Technical Support: www.ge-ip.com/support

Phone: +86-400-820-8208

+86-21-3217-4826 (India, Indonesia, and Pakistan)

Technical Support Email: support.cn.ip@ge.com (China)

support.jp.ip@ge.com (Japan)

support.in.ip@ge.com (other Asias locales)

Customer Care Email: customercare.apo.ip@ge.com

customercare.cn.ip@ge.com (China)

Welcome

Contact GE INTELLIGENT PLATFORMS

2

Proficiency* Machine Edition*

Proficiency* Machine Edition* offers you a complete solution for the development of automation applications under one roof. With Machine Edition's integrated development environment and tools, you will spend more time building applications and less time learning the software.

Machine Edition products are fully integrated with the environment and with each other:

- They share the same project database. No more wasted time synchronizing data points between applications!
- They share the same set of tools, providing a consistent user interface throughout the development process.
- They feature full drag-and-drop capabilities between tools and editors.
- They feature a true scalable solution. You can choose what type of machine your projects are downloaded to and will run on.

The first part of this chapter provides an overview of the Machine Edition environment. The second part provides some key Machine Edition concepts you need to know before you begin. The third part shows how to open and explore some of the sample projects included with a new Machine Edition installation. When you have finished, you will have a solid foundation for building your own automation projects.

- For more help getting started, in the  InfoView tab of the Navigator, expand the **Getting Started** library.

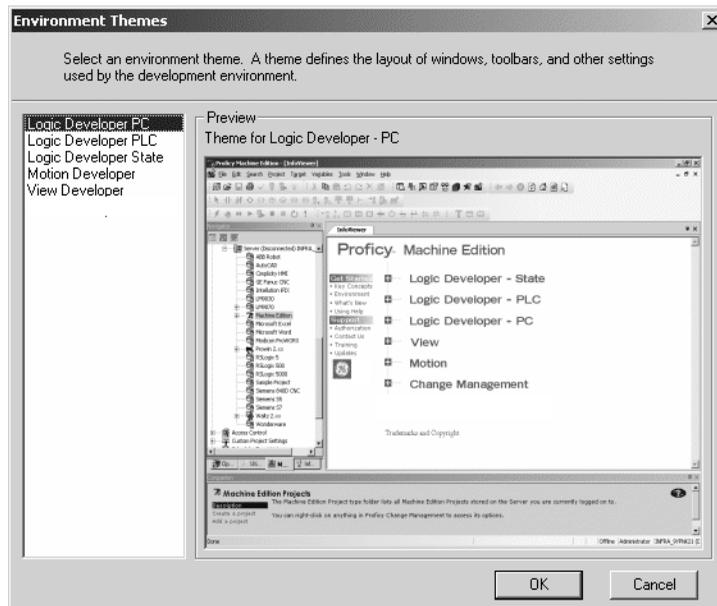
QUICK START

Machine Edition makes it easy to develop a project.

To start Machine Edition

1. On the Windows toolbar, click  **Start**, point to **All Programs**, then **Proficy**, then **Proficy Machine Edition**, and then click **Proficy Machine Edition**. (See also page 30.)

When Machine Edition initializes for the first time following installation, the Environment Themes dialog box appears.

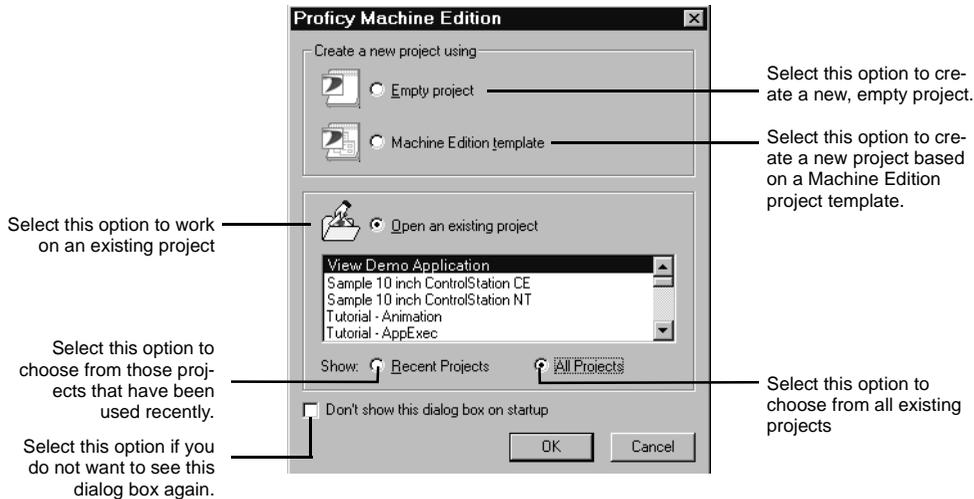


You can change the default theme later by clicking the Window menu and choosing **Apply Theme**.

2. Choose the environment theme you want to work in.
3. Click **OK**.

When you open a project, the appearance of your Machine Edition screen matches the preview in the Environment Themes dialog box. As you work, you can modify the environment – opening and closing windows, changing the selected toolbars, and so on. Any changes are retained in your default environment setup.

The Machine Edition dialog box appears.



4. Select the appropriate option to open a project. The **Open an existing project** option is selected by default.

Notes

- If you select either Empty project or Machine Edition template, the New Project dialog box appears, in which you can create a new project.
 - If you select Open an existing project, you can choose from Recent Projects (the default) or All Projects. You may need to also specify the location of the project: whether it is located on the local computer (My Computer), in a shared folder (Shared Projects), or on Change Management Server (Server).
5. If you selected the **Open an existing project** option, select the project that you want to open from the list.
Existing projects include samples and tutorials that you can use to familiarize yourself with Machine Edition.
 6. If desired, select the **do not show this dialog box on startup** option.
 7. Click **OK**.

Your project opens in the Machine Edition environment that you specified in the Environment Themes dialog box. For more information, see “Machine Edition Projects” on page -29.

MACHINE EDITION ENVIRONMENT

All Machine Edition tools and editors appear in the Machine Edition window. The run-time programs are separate Windows applications, so you can run a finished application without starting Machine Edition.

The following illustration shows a possible layout of the tools and a couple of the editors available to you. Most of the time, you will be using only a few of these at once—you can open and close tools and editors as you need them. Many tools are also available only when editing a project. We will look at some of the tools in the following section; for more details on the others, see the online Help.

Navigator:
Organizes and displays project information in a tree structure.

Navigator tabs:
Conveniently layers information about your projects.

Inspector:
Displays the properties of a selected object.

Companion:
A dynamic help window that provides information based on what you are doing.

The screenshot shows the Proficy Machine Edition interface for editing a ladder logic program. The main window displays a ladder logic diagram with rungs and logic instructions. Surrounding the main editor are several toolbars and panels:

- Navigator:** A tree view on the left showing the project structure.
- Inspector:** A panel below the Navigator showing properties for the selected object.
- Companion:** A help window at the bottom left providing context-sensitive information.
- Toolchest:** A panel on the right containing pre-configured objects for drag-and-drop.
- InfoViewer:** A panel on the right displaying reports and help.
- Data Watch:** A panel at the bottom right showing the current status of project variables.
- Feedback Zone:** A panel at the bottom right displaying output information.

The ladder logic diagram shows a sequence of instructions: START, calculate total flow, MOV (Dumpey IN, Dmummy 0), EN (Dmummy A), ADD (Dmummy A, Dmummy C), and Zone2.Dampey B. It also includes components like Blower, Damper, and \$Blower.

Tabbed Editor window:

One of many windows where you create and edit your application. Tabs at the top let you switch between active Tabbed Editor windows with the click of a mouse.

Common Machine Edition Tools

Data Watch:
Displays the current status of your project's variables.

Toolchest:
Contains pre-configured objects that you can drag into Machine Edition projects.

InfoViewer:
An embedded browser used to display reports and comprehensive help.

Feedback Zone:
Displays output information generated by Machine Edition components.

GETTING TO KNOW MACHINE EDITION

The following are some key features of the Machine Edition environment. Knowing them will make your first few hours with Machine Edition a breeze.

Right-click, right-click, right-click

No matter what object appears on your screen while using Machine Edition, you can right-click it to perform operations on it. In fact, this is probably the most common way you will get things done. Machine Edition tailors the menu of commands depending on the current status of your project.

Getting Help

There are many ways to access the Machine Edition online help system.

- Press F1 on any selected item for context-sensitive help.
- Browse through the table of contents in the ? InfoView tab of the Navigator.
- Perform a full-text search of the help in a separate HTML Viewer window: on the Help menu, choose **Search**.
- Search for keywords using the index: on the Help menu, choose **Index**.
- Use the ? Companion to dynamically display a brief description about whatever item you have selected.

You can also access additional help on the web. In Machine Edition on the Help menu, point to **Proficiency on the Web**, and then choose:

- **Proficiency Support**
- **GE IP Home Page**

Machine Edition provides two kinds of Help windows to display help information: the **Companion** and the **InfoViewer**.

The screenshot shows the Proficy Machine Edition software interface. The Help menu is open, showing options like Finding Information, Index, Product Authorization, and Important Product Information. The Companion window is open, displaying a table of contents for the 'Buttons' topic. The InfoViewer window is open, displaying detailed HTML-based help for the 'Buttons' topic, including a list of steps for changing the size of a button object.

To access the online Help index, on the Help menu, choose **Index**.

To access the full-text search viewer, on the Help menu, choose **Search**.

The **InfoView** Tab contains the table of contents for the online help. Double-click a page to view it.

The **Companion** is a dynamic help window that displays snippets of information on any selected item.

For detailed context-sensitive help, select any item (in this case, a button on a graphical panel) and press F1.

Use the **Info-Viewer** toolbar to navigate the HTML help.

The **InfoViewer** window is a browser that displays the comprehensive HTML-based online help.

Click this button to open the Info-Viewer with more detailed information on the topic in the Companion.

Getting Help

Companion

The Companion window displays brief information about whatever item you are working with, anywhere in Machine Edition. Whether you need the meaning of a property, the purpose of a node in the Navigator, or an explanation of errors generated during validation, the Companion is always there for you.

By default, the Companion is open while editing projects. If it has been turned off, click  on the Tools toolbar to open it.

InfoViewer

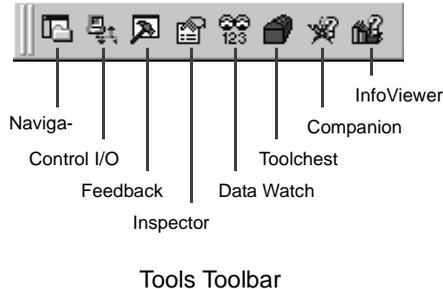
The InfoViewer window is an embedded Web browser that provides more detailed and procedural help. It is launched whenever you access help topics from the InfoView Tab, the help index, or by pressing F1 on a selected item.

 **Want to know more?** In the Help Index, look up "Help" and choose "Finding information in the Help".

Accessing the Right Tool

Often, Machine Edition Help will direct you to a specific tool window. If the tool isn't visible, there are two ways to open it:

- On the **Tools** menu, choose the tool name.
- On the Tools toolbar, click the tool icon. To display the Tools toolbar, on the **Tools** menu, choose **Toolbars** and select a menu item.



If you are not sure of the name of a tool within a toolbar, hover the mouse pointer over the tool's button to display its name in a tool tip.

 **Want to know more?** In the Help Index, look up "Toolbars: an Overview".

Using docking markers

If the Show Docking Markers option is enabled, you can use docking markers to help dock a tool window to a desired location.

As you drag a dockable window, a series of docking markers appear, indicating valid docking locations. Docking markers appear as a series of large blue arrows. Move the mouse over a docking marker to preview how the window would be docked when using that marker. Release the mouse button while hovering over a marker to use that location.



These markers dock tools to the edge of an existing tool or to the Machine Edition environment window.



This marker "stacks" the tool overtop an existing tool window, forming a series of tab-docked windows. You can then switch between tab-docked tools by clicking the tabs that appear at the bottom of the tab-docked windows.

Projects and the Navigator

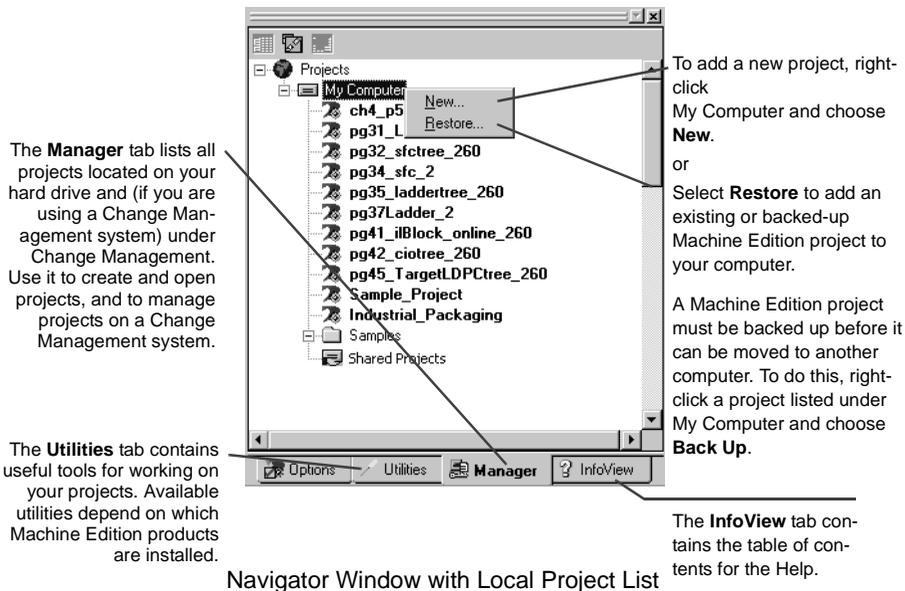
Navigator

The Navigator window organizes and manages your projects.

- Use the Navigator to create and manage projects, add targets and components to your project, set your environment preferences, create scripts, open editors, create variables, and more.
- The Navigator is organized into several tabs. The available tabs depend on which Machine Edition products are installed and whether a Machine Edition project is open. For example, the Project and Variables tabs appear only when a Machine Edition project is open.
- Within each tab, items are displayed in a tree of nodes or folders. You can expand and collapse the tree, just like folders in Windows Explorer.

The following picture illustrates the Navigator prior to any projects being opened. All the files listed under My Computer are projects that you can access on your computer. The **Samples** folder contains sample projects and tutorials.

? **Want to know more?** In the Help Index, look up "Navigator" and choose "Navigator Tool Window".

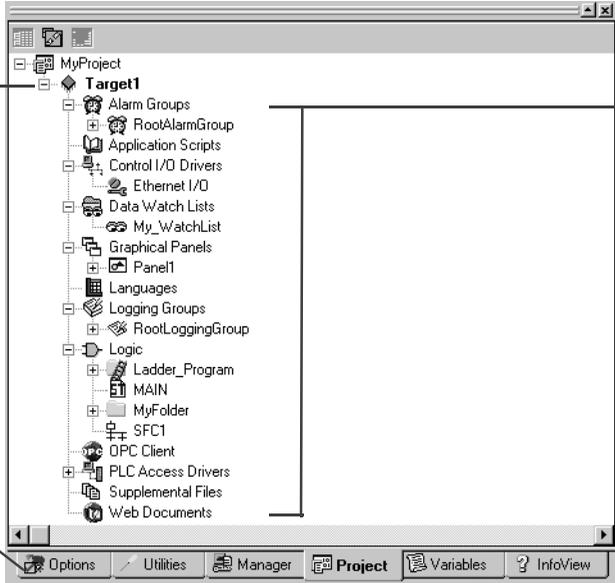


Navigator Window with Local Project List

To open a project in the Navigator

1. In the  Manager tab of the Navigator, right-click an existing project listed under  My Computer and choose **Open**.

The project is opened in Machine Edition. The Variables and Project tabs are added to the Navigator and the nodes change to reflect those components that make up the project you just opened.



The screenshot shows the Navigator window with a tree view under 'MyProject' containing a 'Target1' folder. The tree includes sub-folders like Alarm Groups, Application Scripts, Control I/O Drivers, Data Watch Lists, Graphical Panels, Languages, Logging Groups, Logic, Ladder_Program, MyFolder, SFC1, OPC Client, PLC Access Drivers, Supplemental Files, and Web Documents. The bottom tab bar shows 'Options', 'Utilities', 'Manager', 'Project', 'Variables', and 'InfoView'.

Targets are the hardware devices and computers that your finished project will run on. A project can have multiple targets.

These nodes represent the different parts of your project. They appear when you add components to a target.

For example, the Control I/O Drivers, Ladder_Program, Logic and SFC folders appear when a Logic component was added to Target1 (when using Logic Developer - PC).

The **Options** tab contains option and preference settings for the Machine Edition environment.

The **Project** tab organizes your project. Use it to navigate around your project, and add, delete, and configure the items that make up your project.

The **Variables** tab contains all the variables in your project. Use it to add, delete, move, and otherwise manipulate variables.

Navigator Window with an Open Project

Properties and the Inspector



Inspector

In Machine Edition, practically every object has *properties*. Properties are attributes and information about that object. For example, the properties of a box on an HMI graphical panel include Height, Width, Fill Color, and Outline Color.

The properties of an object are edited in the Inspector window:

- To edit an object's properties, select it with the Inspector open. The Inspector displays all those properties associated with the selected object.
- When a property changes an object's appearance, you will see the results of the change immediately in the object's editor.
- To see more properties for an object, select a tab at the bottom of the Inspector.

This is the kind of object you selected.

If you are ever unsure of the valid range for a property, hover the mouse over its current value and look at the status bar at the bottom of the Machine Edition window.
Maximum length is [255]. All characters are valid

To edit this property, type the new value in the box.

To edit this property, select the new value

To edit this property, click the **...** button.

Properties are organized much like Windows folders.
Click **+** to expand the list of sub-properties.

Select a tab for more properties.

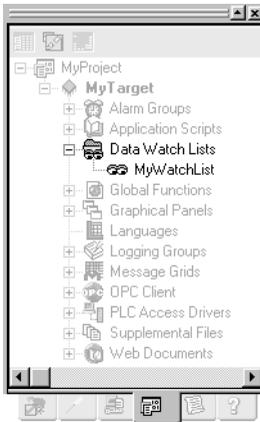
Variable [Target1]	
Name	myDINT
Description	Filler status
Publish	True
Mark As Used	False
Array Dimension 1	0
Data Source	PLC Access
Device	<Unassigned>
I/O Address	A13
Data Type	DINT
Current Value	0
Initial Value	0
Default Display Format	Decimal
Retentive	False
Engineering Units	
Range Limits	

Inspector Window

As you select the various properties within the Inspector, the Companion displays a brief description of the selected property.

? Want to know more? In the Help Index, look up "Inspector" and choose "Inspector Tool Window".

Data Watch Lists



Navigator: Project tab
Data Watch nodes

The  Data Watch (available only while editing a project) is a debugging tool that you can use to monitor and edit real-time values of variables defined in your project. This is useful while working online with a run-time target. With the Data Watch tool, you can monitor individual variables or user-defined watch lists of variables. You can change variable values and force the state of discrete (BOOL) variables. Watch lists can be imported, exported, or saved with a project.

There are three tabs in the Data Watch tool:

- The **Static** tab shows variables added to the Data Watch tool.
- The **Auto** tab contains variables selected in the variable list or associated with the currently-selected instruction in ladder logic.
- The **Watch List** tab contains all variables in the currently selected watch list. A watch list lets you create and save a separate list of variables to monitor. You can define more than one watch list, but only one watch list can be monitored in the Data Watch tool at a time.

 **Want to know more?** In the Help Index, choose “Data Watch List”.

The structure variable **Robot1** has been expanded to show all of its elements.

The **Static** tab contains variables that you add to the Data Watch tool.

The **Auto** tab displays variables associated with the item currently selected in an editor or tool.

	Address	Value
Robot1		Done: On, Active: Off, Begin: Off, photoeye: Off
Done		On
Active	T1	Off
Begin	Q2	Off
photoeye	I5	Off
Height	A15	77

The IEC or Reference address of variables mapped to I/O terminals or Controller memory.

The name and value of each element of the structure variable **Robot1** is shown here, even when not expanded.

Double-click a value to change it.

A **Watch List** tab contains all variables in your project that have been added to the currently selected Watch list.

Data Watch Window

To define a Watch List:

1. Right-click the Data Watch Lists folder and choose **New**.
2. Double-click the new Watch List to open it in the Data Watch tool.
3. Add variables to the Data Watch as desired.

The changes to the watch list are automatically saved for later use.

Smart Lists

While working in Machine Edition editors, you are frequently required to input variable names, instruction mnemonics, and other data items. The Smart List can speed text entry of these items.

This text changes color based on whether the entry is valid or invalid. For example, invalid entries are red. Valid entries in the list are black. Valid typed entries that are not in the list appear green; accepting this entry creates a new item (in this case, a new variable).

This tells you what you need to enter or select.

As you type in the input box, the list selection jumps to the item that is closest to what you typed. Press ENTER to create a new item based on what is typed, or use the Down arrow to use the selected item.

For example, if you typed "fill" in the Smart List, the FillLevel variable would be highlighted. If you want to use an existing variable (in this case, "FillLevel"), use the Down arrow to select the highlighted item, then press ENTER. To create a new variable named "fill", press ENTER without using the Down arrow.

Smart List Display

The Toolchest



Toolchest

The Toolchest (available only while editing a project) is a repository of preconfigured object templates you can drag into your project. These objects can be as simple as a single ladder logic instruction, or as complex as a robotic arm with fully-configured ladder logic and HMI animation.

While Machine Edition comes with a set of preconfigured object templates, you can create your own. Because you can add as many copies of them as you want, this can save you hours of development time.

? **Want to know more?** In the Help Index, look up "Toolchest: an Overview".

Within a drawer, Toolchest items are organized into folders.

Toolchest items are grouped into drawers. Select the drawer you want with this list box.

You can drag these logic instructions directly into your ladder program.

The Ladder, HMI, and SFC drawers contain simple instructions and functions.

Other drawers contain fxClasses—definitions for fully configured objects that you can also drag into your project.

Toolchest Window

The screenshot shows a window titled 'Ladder' with a list of drawers. The drawers are: Advanced Math, Allen-Bradley RIO, ASCII Communications, Basic Math, Bit Shift/Rotate, Bitwise Logic, Coils, Comparison, Contacts, Conversion, Copy, Counters, DeviceNet, Increment/Decrement, InterBus-S, Process Control, Program Flow, Sequencer, and Timers. The Bitwise Logic drawer is expanded, showing sub-items: Bitwise AND, Bitwise NOT, Bitwise OR, and Bitwise XOR.

Definitions of most preconfigured objects within the Toolchest are displayed in the Companion.

The Feedback Zone



Feedback Zone

The Feedback Zone (available only while editing a project) is an interactive window that displays output information generated by Machine Edition components. With the Feedback Zone, you can keep track of project information, locate validity errors within your project, display generated reports, and more. It is organized into several tabs. Click a message to display more information about the error in the Companion Window.

? Want to know more? In the Help Index, look up “Feedback” and choose “Feedback Zone Tool Window”.

Press F4 to cycle through entries in the Feedback Zone.

In the Build Tab (shown), Reports Tab, and References Tab, pressing F4 also opens the project

```

Error 2861: 'Active' is invalid [Rectangle, Rectangle257: Fill Color Expr]
Error 2862: 'boxpresent' is invalid [Rectangle, Rectangle258: Fill Color Expr]
Error 2603: Expecting variable left of := [Script, Conveyor_On: Text]
Error 2603: Expecting variable left of := [Script, Conveyor_Off: Text]
Validating HMI Application Scripts...
Validating HMI Alarm Groups...
RootAlarmGroup
Validating HMI Logging Groups...
RootLoggingGroup
Validating HMI Web Documents...
Target1 - SFC

Validating Complete - 5 error(s), 0 warning(s)

Tip: Press F4 to cycle through warnings and errors.
  
```

The **Build** tab displays the status and results of a validate or download operation. Use this tab to discover and fix any errors you may have in your project.

The **Import** tab displays a list of errors and warnings from an import operation (for example, errors/warnings are listed when you import variables from a Controller into a Machine Edition project).

The **Messages** tab tracks and displays operations that have been completed in Machine Edition. For example, a message is added every time you open a project.

The **References** tab displays a list of all the places a selected variable is used in a project. Click the Reference tab and then select a variable from the Variable List. You will see a list of all the places the variable is referenced in your project.

The **Reports** tab displays a list of all reports generated during the current session. Double-click a report in the list to redisplay it in the InfoViewer.

Feedback Zone

Managing Variables

Variables are named storage spaces for data values defined in Machine Edition projects. A variable could store the current velocity of a motion Controller motor, the height of a robotic arm, or any other value that an application needs to keep track of. Most variables in a project can be shared among various components and targets, such as View panels and Logic Developer - PC's ladder logic.

You manage variables in the  Variables tab of the Navigator, also named the *Variable List*. Like most items in Machine Edition, you configure variables by editing their properties in the Inspector.

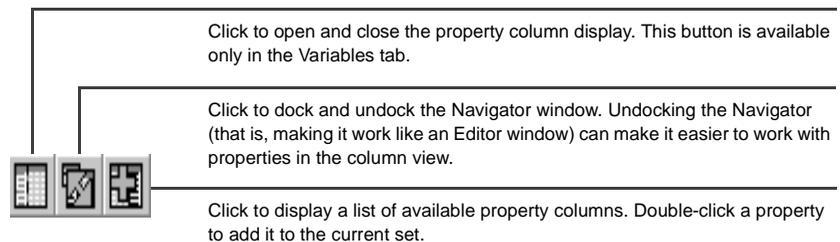
The values a variable can store depends on its *data type*. For example, a DINT data type indicates that the variable can store “Double Integers”, 32-bit values. The location where a variable’s value is stored is indicated by its *data source*. Typically, a variable’s value is either stored internally in the target’s memory or is retrieved (and sent) to external Controller hardware, via an I/O terminal or other connection. Available data types and data sources depend on the target type and (if applicable) components added to the target.

You can also use *arrays* and *structure data types* in Machine Edition projects. An array is a series of variable elements with identical data types, referenced by a 0-based index (as in “MyArray[3]”). A structure data type is a group of variable elements that may or may not have the same data type, referenced by the name of each element (as in “MyStructure.MyElement”). Custom structure data types are created with fxClasses in the Toolchest.

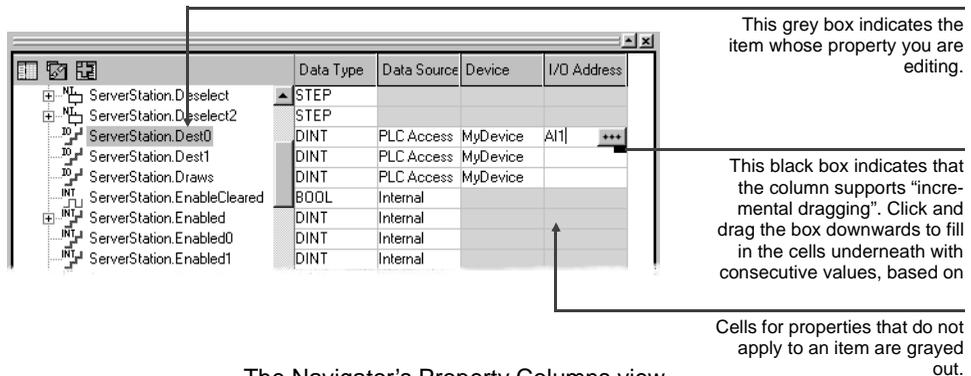
 **Want to know more?** In the Help Index, look up “STRUCTURE data types”.

Property Columns

In addition to the Inspector, with the three buttons at the top of the Navigator, you can edit variable properties in a spreadsheet-like column view.



When opened, the Navigator's property columns view consists of a grid of cells:



The Navigator's Property Columns view

Variable Reports

While you can generate reports on many things in Machine Edition, they are especially useful for the Variable List. Among the available reports are:

- All variables by name, filtered or unfiltered.
- Cross-references to variables (that is, the places they are used in the project).
- All forced variables (Logic Developer - PC and Logic Developer - PLC only).
- Unused variables (that is, variables with no references anywhere in the project). The unused variables report also has a link that lets you delete all unused variables from the project at once.

Reports are displayed as HTML pages in the InfoViewer. To generate a report, click anywhere in the Variables tab then, on the **File** menu, select **Report**.

? Want to know more about Variables? Look up the following in the Help Index:

- For information on variables in general, choose "Variables: an Overview".
- For information on variables in View, choose "Variable support in View Developer".
- For information on variables in QuickPanel projects, choose "Variable support in QuickPanel applications".
- For information on variables in Logic Developer - PC, choose "Variable support in Logic Developer - PC".
- For information on variables in Logic Developer - PLC, choose "Variables in Logic Developer - PLC vs. Memory in GE IP Controllers: an Overview".
- For information on variables in Motion Developer, choose "Variable support in Motion Developer".
- For information on importing variables from other applications or database files, choose one of the "Importing Variables..." topics.

MACHINE EDITION PROJECTS

During development, your automation application is named a *project*. Each Machine Edition project is made up of *targets* and (sometimes) *components*. A target represents the hardware platform where the finished project runs, such as a Windows XP computer, a QuickPanel View/Control unit, a motion Controller, or a QuickPanel unit. Different Machine Edition products support different target types. Targets are often further subdivided into models. Components add specific capabilities to a target. Available components depend on the target type, model, and what Machine Edition products you have installed.

For example, with View, you can add an HMI component to Windows PC and QuickPanel View/Control targets, used for creating human-machine interfaces. With Logic Developer - PC, you can add a Logic component to Windows PC and QuickPanel View/Control targets. You can then create ladder logic to make a PC act as a Controller. QuickPanel View targets do not support logic; therefore, you cannot download a project that contains logic to one.

With Motion Developer, you can program motion Controllers from your PC. Each motion device programmed by Motion Developer is represented by a separate Motion target. Because motion targets have no additional capabilities, they do not require adding components.

 **Want to know more?** In the Help Index, look up “projects” and choose “Machine Edition Projects: an Overview”.

Sharing Projects between Machine Edition Workstations

If your site has multiple Machine Edition workstations connected through a network, you can use the  Shared Projects folder to work on the same set of projects. Projects under the Shared Projects folder are stored in a directory you specify, typically a shared directory on the network.

No access or version control is applied to the Shared Projects folder. If multiple users open the same project at the same time, some changes may be lost.

Note: Shared Projects and Shared Variables are two completely different and separate features. For information on Shared Variables (which implement proxy variables between targets in different projects), see “PLC Access I/O” on page -83.

 **Want to know more?** In the Help Index, look up “Shared Projects”.

Running a Sample Project (View/Logic Developer - PC)

If you are using View or Logic Developer - PC, we've provided some sample projects that show basic operations of Machine Edition. Take a few minutes to complete the following steps and learn the basics of project development in Machine Edition.

- For a more detailed example of creating projects, in the Help Index, look up “Tutorial” and choose one of the topics listed there.

To run a sample project

1. Run Machine Edition from the Start menu | All Programs | Proficy | Proficy Machine Edition | Proficy Machine Edition.
2. In the  Manager tab of the Navigator, double-click one of the sample applications listed under the  My Computer folder.
Sample applications include the following:
 - **Alarm Trends:** an HMI application that demonstrates alarms and charts.
 - **Animation Features:** an HMI application that depicts the various types of animation available in View.
 - **Brewery:** an HMI and logic brewery application that runs on a Windows PC target.
 - **Car Wash:** an HMI and logic application that runs on a Windows PC target. This application is built in an SFC document.
 - **ControlStation HMI Features:** an HMI and logic application that runs on a Windows PC target. This application also shows some web documents.
 - **QP Control 6 inch Brewery:** an HMI and logic brewery application that runs on a QuickPanel Control unit. (Projects containing a logic component cannot be downloaded to a QuickPanel View unit, because QuickPanel View units do not support logic.)
 - **QP Control 6 inch Ventilation:** an HMI and logic application that runs on a QuickPanel Control unit. This application runs a four-zone ventilation logic system. It is to be used with the QuickPanel Control unit Tutorial. (Because QuickPanel View units do not support logic, projects with a logic component cannot be downloaded to a QuickPanel View unit.)
 - **ST - Lunar Lander:** an HMI and logic application that runs on a Windows PC target. This project demonstrates the Structured Text (ST) language and its interaction with User Defined Function Blocks (UDFBs).

Tutorials include

- **Animation:** an HMI application that illustrates and provides details on the various animations that you can use in your HMI.
- **AppExec:** an HMI application that demonstrates the AppExec Script function.
- **Keystrokes:** an HMI application that displays the analog values associated with the keys on your keyboard.
- **Logging:** an HMI application that demonstrates how to log production data to an ASCII text file.
- **Recipes:** an HMI application that shows how one would load and modify recipes.
- **Scripting:** an HMI application that demonstrates some of View's scripting capabilities.

3. Browse through the project in Machine Edition.

Open the project's graphical panels, ladder logic, and SFCs (if they exist) in their respective editors. To open an editor, right-click the appropriate node in the  Project tab of the Navigator and choose **Open**.

Also, take a look at the properties of the various nodes and objects. To view an object's properties, open the Inspector window, and then select an object.

4. In the newly opened project, press F9 to validate, download, and run the sample application.

The F9 key is a shortcut for the Run command which automatically starts View Runtime and the Controller.

The sample application should now be running in both View Runtime and the Controller. View Runtime appears, displaying the first panel of the project's HMI. You can see the real-time status of the logic by going online to the Controller.

To go online to the Controller

If you have chosen a project that contains logic and/or an SFC document, you can *go online* to the application and watch the logic being executed.

1. Minimize the Runtime window (but do not close it) and return to Machine Edition.
2. In the  Project tab of the Navigator, right-click the  Target node and choose **Go Online**.

You are now online to the Controller; that is, you are working with the application while it is running. The Controller is the part of the Runtime that solves logic and SFCs.

3. In the  Project tab of the Navigator, open the Ladder Editor by double-clicking the  Ladder Program node or open the SFC Editor by double-clicking the  SFC node.

In the editor, you can watch the logic being solved as the Controller operates. To start and stop Runtime and the Controller, right-click the  target, point to **Online Commands**, and then click **Start/Stop Runtimes**. To go offline from the Controller, right-click the target, and then click **Go Offline**.

Developing a Machine Edition Project

The first thing to decide when you create a Machine Edition project is where the project will run after it is developed. That is, on what type of target hardware it will run.

For View and Logic Developer - PC, targets can be a Windows PC computer (either the one you are developing the project on, or a remote one that you connect to through a network), or a QuickPanel View/Control unit.

For View alone, your target can also be a conventional QuickPanel unit.

For Logic Developer - PLC and Logic Developer - State, your target is a PACSystems, Series 90, or VersaMax Controller. You can also configure remote I/O targets that represent a variety of remote I/O adapters and their associated I/O modules. See GFK-1918 Getting Started Logic Developer - PLC.

If you are using Motion Developer, the choice is easy—determine whether your motion hardware is a motion Controller/drive or a drive-only device (see page 101).

After you've decided on the target, you need to determine which components your project will include: HMI (with View) and/or Logic (with Logic Developer - PC). Note that ViewStations do not support Logic components. A project can have multiple targets of different types with various components running on each target. In some cases, targets can be converted from one type to another. For example, you can convert a Windows XP target to and from a QuickPanel View target.

 **Want to know more?** In the Help Index, look up "Targets: an Overview".

To create and develop a project

The following procedure introduces the general steps involved in creating a project using a template, and downloading a project to a target computer.

1. Start Machine Edition from the Start menu | All Programs | Proficy | Proficy Machine Edition | Proficy Machine Edition.

2. Create a project using a template.

In the  Manager tab of the Navigator, right-click  My Computer and choose **New**. The New Project dialog box appears.

If you are using shared projects, you can also add new projects under the Shared Projects folder. Or, if you are working on a Change Management system, you can add new projects to the Machine Edition folder under the Server.

3. In the New Project dialog box, enter a name for the new project, select a template, and then click **OK**.

A description of the template appears below your selection. You can click hypertext links in the description for details on the template components.

At this point, you need to know the type of target hardware to which your project will download, as this will determine which template you choose. In some project templates (such as the “View/Control” template), you can select this as one of several parameters within an HTML page on the template dialog box.

The project is opened in Machine Edition and the Navigator changes to reflect those components that make up the project you just created.

 **Want to know more?** In the Help Index, look up “templates” and choose “Creating a New Machine Edition Project” or “Creating a new project under a Change Management system”.

4. In the Variable List, create variables for your application.

In the  Variables tab of the Navigator, right-click the Variable List node, point to New Variable, and choose the type of variable you want to create.

By default, the  Variable List node filters out all system variables. System variables are created automatically when you add components to Windows PC, QuickPanel View/Control, QuickPanel, or GE IP Controller targets. To display all variables including system variables, right-click the Variable List node, point to **Filter By**, and choose **No Filter**.

 **Want to know more?** In the Help Index, look up “Variables: an Overview”.

5. Create your application.

With Logic Developer - PC, add logic (Ladder, FBD, IL, ST, SFC) to your project, and then configure a control I/O driver to model your hardware. If the template you chose did not include a Logic component, add one now—right-click the target, point to **Add Component**, and choose **Logic**. (View-only targets do not support logic.)

- In the  Project tab of the Navigator, under the Logic folder, open the Ladder Editor by double-clicking the  Ladder Program node.

- Drag ladder instructions from the  Toolchest into the editor. You can find ladder instructions in the PC Ladder Instructions drawer of the Toolchest.
- Assign variables to instructions. You can do this with the Smart List, which appears when you insert or double-click an instruction. Or, drag a variable from the  Variables tab of the Navigator and drop it on the instruction you want it mapped to.
- You can add additional logic blocks (ladder subroutines, FBD, IL, ST) and organize your logic with SFC.

 **Want to know more?** In the Help Index, look up “Ladder Instructions: an Overview (PC)”.

With View, you can create the graphical panels and animation for your project. If the template you chose did not include an HMI component, then add one now—right-click the target, point to **Add Component**, and choose **HMI**. (If a target has both an HMI and a Logic component, you will typically create the Logic component first.)

- In the  Project tab of the Navigator, open the Panel Editor by double-clicking a  Panel node.
- Use the Graphical Panel toolbar to create your HMI and/or drag graphical objects from the Toolchest onto a panel. You can find a set of fully-configured objects (complete with animation) in various Toolchest View Expert Objects drawers.

With Motion Developer, begin by running the Motion Expert wizard.

- If necessary, open the “Main Wizard” home page by right-clicking the Motion target and choosing **Main Wizard**.
 - In the wizard page, click Motion Expert and follow the instructions on screen. For more information on using wizards, see page 106.
6. If necessary, configure the I/O hardware connections for your project.
- For Logic Developer - PC, set up Control I/O in the  Control I/O Drivers folder (in the  Project tab of the Navigator). To add a driver, right-click the Control I/O Drivers folder and choose **New Driver**. Use the Control I/O tool to configure your drivers. Map variables to I/O terminals.

 **Want to know more?** In the Help Index, look up “Control I/O Tool: an Overview”.

- For View, set up PLC Access I/O in the  PLC Access Drivers folder. To add a driver, expand the PLC Access Drivers folder, right-click View Native Drivers, and click **New Driver**. Configure the drivers in the Inspector window. On Windows PC targets, you may want to set up OPC I/O in the  OPC Client folder.

7. When your project is complete; validate, download, and run your project by pressing the F9 function key.

Machine Edition saves the project, performs a validation, builds the run-time files, and attempts to establish a connection to the target computer.

Validating and Downloading a Project

During project development, you will go through the validation and download processes several times. The validation process checks the project for errors. If the project contains errors, they are listed in the Build tab of the  Feedback Zone.

- All errors *must* be fixed before the download process can proceed. Warnings are also listed in the Feedback Zone, but they do not prevent the download from starting.

When fixing project errors, use the F4 function key to scroll through errors in the  Feedback Zone, jumping to their locations in the project.

The download process involves two steps. The first step creates (or “builds”) all the run-time files necessary for a target to perform its role in a completed project. The second step downloads those files to the target devices or computers.

- For Motion Developer, the easiest way to set up the motion control device for downloading is with the Set Communication Parameters wizard. Open the Main Wizard page for the target (right-click the target and select **Main Wizard**). Then, point to **Target Configuration** and click **Communication Parameters**. Click **Finish** when you are done.
- To download a View or Logic Developer - PC project to the Windows PC computer on which you are developing (a *local* target), ensure that the target property Computer Address is set to “.” (without the quotation marks). Otherwise—for *remote* targets—enter the IP address or computer name of the remote computer you want to download to, in the Computer Address property.
- To download a QuickPanel project, ensure that the target property Computer Download Port is set to the applicable COM port.

Additional basic information on setting up Windows PC and QuickPanel View/Control targets follows.

Preparing a Remote Windows PC target for Downloading

The following steps need to be completed on a remote Windows PC target computer before you can download a project to it.

1. If you are downloading to another Windows PC computer, ensure that the run-time files are installed on that device.

QuickPanel View/Control targets are shipped with the run-time files preinstalled. If you are using your own Windows PC computer, you will have to install the run-time files yourself from the Machine Edition installation DVD (see page 4).

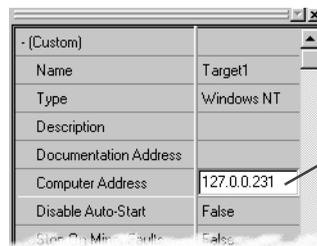
2. Share the Machine Edition installation directory with the development computer that contains the project files.

In the Windows PC Explorer, navigate to the Proficy Machine Edition folder, right-click it and choose **Sharing**. Click **Shared As** and in the Share Name field, enter Proficy Machine Edition.

3. Ensure you have the proper permissions to download files to the target computer.

In Windows PC, open the Control Panel, click **Administrative Tools**, and then click **Computer Management**. Expand "System Tools" and then expand "Local Users and Groups". Select the Users folder. In the right pane, double-click the Guest user account. In the Guest Properties dialog box that appears, clear the Account is Disabled check box.

4. Within your Machine Edition project, specify the IP address or computer name of the remote Windows PC target in the Computer Address target property. This enables Machine Edition to locate and connect to the Windows PC target computer.



Type the IP address or the computer name of the remote Windows PC target to which you want to download your project files.

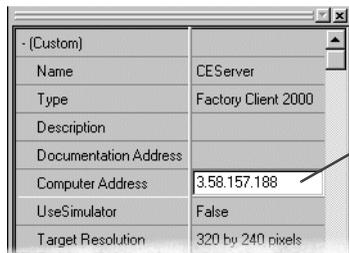
? **Want to know more?** In the Help Index, look up "Preparing Windows PC targets for downloading".

Preparing a QuickPanel View/Control target for downloading

The following steps need to be completed on a QuickPanel View/Control target before you can download a project to it. If you are not physically connected to the QuickPanel View/Control unit but want to test your project, you can download the project to your local computer by setting the Use Simulator target property to **True**. You cannot download a project that contains a logic component to a QuickPanel View target, because they do not support logic.

1. Physically connect the QuickPanel View unit to the Ethernet network. This step may require a network administrator.
2. Within your Machine Edition project, specify the IP address of the QuickPanel View/Control unit in the Computer Address target property.
To find the IP address of a QuickPanel View/Control unit, double-tap the System Information icon located on the unit's desktop.
By default, the QuickPanel View/Control unit uses a DHCP (Dynamic Host Configuration Protocol) IP address. This means that the IP address is dynamically assigned by the Ethernet network server using DHCP. If you want to use a fixed IP address, you must manually set the IP address on the unit.

? **Want to know more?** In the Help Index, look up "Preparing QuickPanel View/Control targets for downloading".



Type the IP address of the QuickPanel View/Control unit to which you want to download your project files.

Testing a View Project

The following steps introduce you to the Quick Test feature. Quick Test is similar to a preview: with Quick Test, you can quickly see the results of any changes you've made to a graphical panel without having to download your entire project.

NOTE The Quick Test feature is not available on QuickPanel targets.

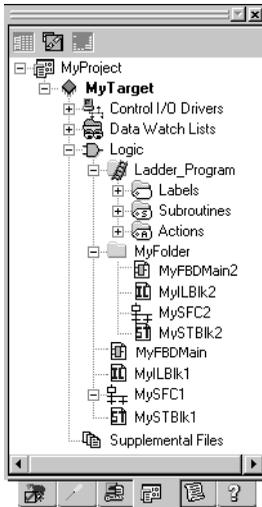
Before you can use Quick Test, your project must have been downloaded at least once to the target computer. If you want to Quick Test a graphical panel from a QuickPanel View/Control project, you must set the target's Use Simulator property to **True** and perform at least one download after changing the setting.

To Quick Test changes made to a graphical panel

1. Ensure that you've downloaded the HMI to the target computer at least once.
2. If you have not already done so, open the graphical panel you want to test.
3. Right-click in the panel and choose **Quick Test**.
View Runtime appears with a preview of the graphical panel.

3

Logic Developer - PC



Navigator: Project tab

Logic Developer - PC is Machine Edition's PC Control product, where you edit a project's logic component. With Logic Developer - PC you create control programs using standard IEC 61131-3 editors in a graphical environment. The programs you create are downloaded to and executed on the PC Logic Controller. You interface to a plant or process through the I/O system of your choice; Logic Developer - PC supports a wide range of industry standard hardware.

Logic Developer - PC supports Windows 7 and Windows XP ("Windows PC" for short), and QuickPanel Control/View targets. All user-defined logic programs are contained in the  Logic folder, including the following:

-  Sequential Function Chart (SFC) editor (see page 41)
-  Ladder Program (see page 45)
-  Instruction List Blocks (see page 49)
-  Structured Text Blocks (see page 52)
-  Function Block Diagram (see page 55)

The following run-time elements are included with Logic Developer - PC:

- Logic Developer - PC Web Access (see page 59)
-  Control I/O Drivers (see page 62)
-  PC Logic Controller (on target computer) (see page 65)

TIP User-defined  folders can be added to the Logic folder or to another user-defined folder to organize your logic. A user-defined folder can contain any type of logic block contained in the logic folder, except ladder (of which there can be only one).

To create a project that includes a logic component

1. In the  Manager tab of the Navigator, right-click  My Computer and choose **New**.

The New Project dialog box appears.

2. In the Project Name box, enter a name for your project.
3. From the Project Template list, choose a template that includes a logic component.
A preview of what each template includes is displayed as you select different templates.
4. Click **OK**.
Your new project is created and the Project tab opens in the Navigator.

To add a logic component to an existing project

- In the  Project tab of the Navigator, right-click a  target, point to **Add Component** and click **Logic**.
The  Logic folder with an empty ladder program and SFC component is added to your project.

To add a user-defined folder to your project

1. Ensure that your project contains a logic component (see above).
2. Right-click the  Logic folder or a  user-defined folder, point to **New**, and click **Folder**.
The new folder appears in the  Project tab of the Navigator in alphabetical order.
3. (Optional.) Enter a name for the new folder.

SFC EDITOR

SFC logic is a graphical language for organizing the sequential execution of control logic. The SFC editor is an easy-to-use graphical tool for editing sequential function charts in the Machine Edition environment. With the SFC editor, you can work on a disk copy of a sequential function chart (offline) or monitor the execution of an SFC running in the Controller (online). You can also configure the appearance and behavior of the SFC editor.

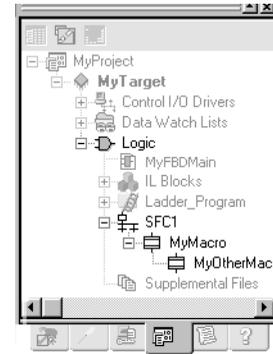
Sequential Function Chart

An SFC is much like a flow chart; it is a high-level control program that organizes detailed blocks of logic into a state-like flow. In general, an SFC consists of steps alternating with transitions. A transition can be a BOOL variable or a Structured Text statement, which resolves to a BOOL value. Each step represents a state that the program is in until the transition following it allows execution to proceed to the next step. Each step contains actions that are user-defined sections of ladder, IL, ST, or FBD logic. The actions in a step are executed in the order they are listed and are repeated each Controller scan while the step is active. Each action in a step has a qualifier associated with it that determines when and how the action is executed.

An SFC document is a grid of cells. Each cell can contain an instruction. The instruction set includes steps, transitions, branches, parallels, and jumps.

An SFC can be expanded by adding macros. Each macro is a special SFC document that can be referenced any number of times from its SFC document or other macros.

? Want to know more? In the Help Index, look up “macro” and choose “SFC Macro”. You can also look up “SFC Tutorial” in the online help Index. This tutorial takes you step-by-step through the creation of a Sequential Function Chart that controls traffic signals.



Navigator: Project tab
SFC nodes

To add a new SFC

1. In the Project tab of the Navigator, right-click the Logic folder or a user-defined folder, point to **New**, and click **SFC**.
The new SFC appears under the Logic folder or a user-defined folder in alphabetical order.
2. (Optional.) Enter a unique name for the new SFC.

To open an SFC or macro for editing

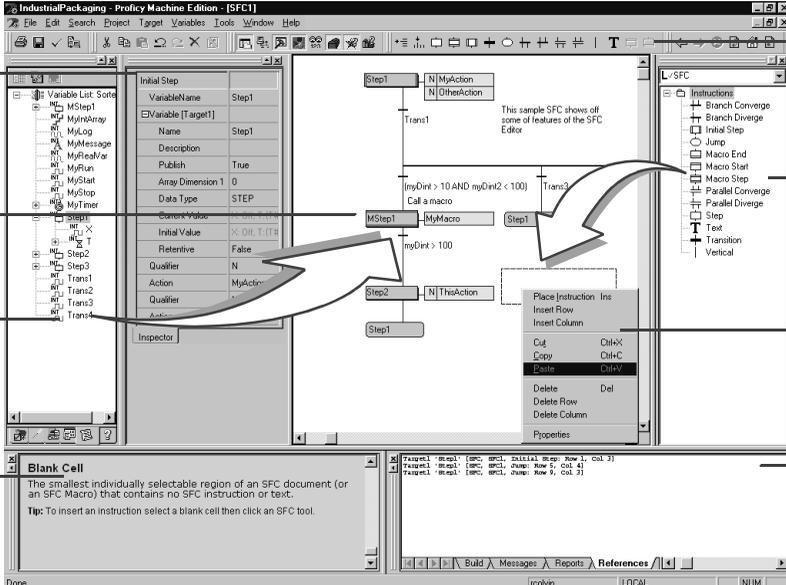
- In the  Project tab of the Navigator, under the  Logic folder or under a  user-defined folder, double-click an  SFC or  macro node. The SFC editor opens with the document ready for editing.

To create a new SFC macro

- In the  Project tab of the Navigator, under the  Logic folder or under a  user-defined folder, right-click an  SFC or  macro node and click **New**.
- (Optional.) Enter a unique name for the new macro.
Your new macro appears as a child node of the SFC or macro node you started with.

Working with the SFC editor - Offline

The SFC editor interacts with the Machine Edition tools to provide maximum flexibility when editing a program. The following picture illustrates some of the operations you can perform.



The screenshot shows the Proficy Machine Edition SFC editor interface. The main window displays a Stateflow chart with steps (Step1, Step2, Step3) and transitions (Trans1, Trans2, Trans3). A context menu is open over a blank cell, showing options like 'Place Instruction', 'Insert Flow', 'Insert Column', 'Cut', 'Copy', 'Paste', 'Delete', 'Delete Flow', and 'Delete Column'. The left sidebar shows a Variable List and an Inspector. The right sidebar shows an L/SFC drawer with instructions like Branch Converge, Initial Step, Jump, Macro End, Macro Start, Parallel Converge, Parallel Diverge, Step, Text, Transition, and Vertical. The bottom status bar shows 'Done' and 'LOCAL'.

Configure instructions and their associated variables in the Inspector.	Insert instructions, rows, and columns with the SFC toolbar.
Call a macro with a Macro Step instruction.	Drag instructions from the SFC drawer of the Toolchest to the SFC editor.
Use the Variable List to create variables then drag them to instructions in the SFC editor.	Right-click a blank cell to start editing.
The Companion provides help on anything you click.	Locate all instances of a variable (References) or syntax errors (Build).

To insert an SFC instruction

1. In the SFC editor, right-click a blank cell and click **Place Instruction**.
2. Choose an instruction mnemonic from the smart list that appears and press **ENTER** to finish.

You can set the SFC editor options to automatically assign default variables to instructions or to let you choose and create your own.

To Quick Edit an SFC

- In the SFC editor, click a blank cell and type instruction mnemonics separated by operators (see the following table). Press **ENTER** to finish.
Each mnemonic specifies one or more instructions. Each operator moves the location of the currently selected cell. Default variables are assigned when Quick Editing an SFC.

SFC Operators

To ...	Use ...
move right one cell	+
move left one cell	-
move down one cell	;
stay on current cell	/

Working with the SFC editor - Online

In the SFC editor, you can view the execution of an SFC as it happens. The following picture illustrates this capability.

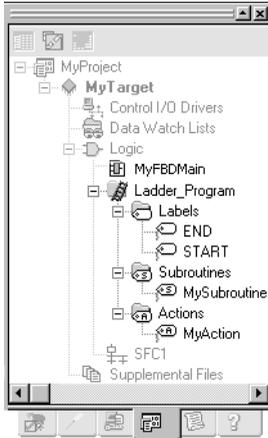
The screenshot shows the SFC editor interface with several windows and annotations:

- Active steps and transitions are highlighted in bold.** (Annotation pointing to the SFC diagram)
- Power flow in the associated ladder program (actions) is indicated in bold while actions are executed.** (Annotation pointing to the Ladder Program window)
- Many actions can be viewed at once.** (Annotation pointing to the Ladder Program window)
- Right-click any BOOL to turn or force it** (Annotation pointing to the right-click context menu)
- The References tab of the Feedback Zone lists all instances of a variable. Click an entry to locate it in an editor.** (Annotation pointing to the References window)
- Use the Data Watch to monitor or change data values in real time.** (Annotation pointing to the Data Watch window)

To go online to the Controller's SFC

- In the Project tab of the Navigator, right-click the target and click **Go Online**. (Before you can go online to the Controller, you must first download the project.)
The Controller's SFC appears in the SFC editor. When the Controller is running, active steps and transitions are graphically indicated.

LADDER EDITOR



Navigator: Project tab
ladder logic nodes

Ladder logic may be the most popular language in use today for creating control programs. The ladder editor is a graphical tool for editing ladder programs in Machine Edition. With the ladder editor, you can work on a disk copy of a ladder program (offline) or with a live program as it runs in the Controller (online). You can also configure the appearance and behavior of the editor under the  Options tab of the Navigator.

Ladder Program

A ladder program is composed of rungs of ladder logic instructions that execute sequentially from left to right, top to bottom. The instruction set includes standard IEC 61131-3 functions supplemented by a large library of advanced and I/O specific functions. Along with rungs and instructions, a ladder program contains:

-  **Labels:** Each default (START and END) or user-defined  label is a rung providing an entry point for program execution.
-  **Subroutines:** Each user-defined  subroutine is a separate, reusable section of logic that can be called from anywhere in a ladder program.
-  **Actions:** Each  action is a separate block of logic that can be referenced in a Sequential Function Chart.

Tip: You can also create User-Defined Function Blocks (UDFBs). Each UDFB is a callable subroutine to and from which you can pass parameters.

 **Want to know more?** In the Help Index, look up "ladder" and choose "Ladder Program: an Overview". Also look up "UDFB" and choose "User-defined Function Blocks: an Overview".

To open a ladder program for editing

- In the  Project tab of the Navigator, under the  Logic folder, double-click the  Ladder_Program node.
Your ladder program opens in the ladder editor.

To locate a label

- In the  Project tab of the Navigator, under the  Logic folder, under the  Ladder Program folder, double-click a  label node.
Your ladder program, subroutine or action opens in a ladder editor window with the label selected.

To create a new subroutine or action

1. In the Project tab of the Navigator, under the Logic folder, under the Ladder Program folder, right-click the Subroutines or Actions node and click **New**.
2. (Optional.) Enter a unique name for the new subroutine or action.
An empty subroutine or action, labeled with the name you entered, is created. Double-click the node to open it for editing. You can add ladder logic to the subroutine or action as required.

Working with the ladder editor - Offline

The ladder editor interacts with all of the Machine Edition tools to provide maximum flexibility when editing a program. The following picture illustrates some of the operations you can perform.

The screenshot shows the Ladder Editor interface with several callouts:

- Click a shunt and type mnemonics to insert instructions (QuickEdit).** Points to the 'MOV' instruction in rung 1.
- Drag variables to instruction parameters to configure them.** Points to the 'MyMessage' variable in rung 1.
- Drag branches around instructions to create parallel execution paths.** Points to the 'TON' instruction in rung 3.
- Display subroutines and actions in separate editor windows.** Points to the 'SUB START MySubroutine' window.
- The Companion automatically provides information on what you click.** Points to the 'Timer Off Delay (TOF)' tooltip.
- Insert rungs and common instructions with the ladder Instruction toolbar.** Points to the 'PC Ladder Instructions' toolbar.
- Edit descriptions in place for rungs, variables, and your ladder program.** Points to the 'Hello' text in rung 1.
- Drag instructions from the PC Ladder Instructions drawer of the Toolchest.** Points to the 'Timer Off Delay' instruction in the toolchest.
- Locate all instances of a variable (References) or syntax errors (Build).** Points to the 'References' and 'Build' buttons in the bottom right.

To insert a rung

- In the ladder logic, right-click a rung, shunt, label or description and click **Insert Rung**.
An empty rung is inserted. You can set the editing options for the ladder editor so that rungs are inserted either before or after the current selection.

To Quick Edit ladder logic

- In the ladder logic, click a rung and type instruction mnemonics separated by semicolons, and then press **ENTER**.

The instructions specified by the mnemonic string you entered are inserted, in order, on the rung. You can enter mnemonics followed by variable names if you want to assign instruction parameters while Quick Editing.

Example: Add a Normally Open Contact (NO) instruction followed by a Coil (OUT) instruction and assign the variables MySwitch and MyCoil to these instructions respectively.

NO MySwitch; OUT MyCoil

Working with the ladder editor - Online

With the ladder editor you can go online to a ladder program being run by the Controller (page 65). While online, you can view the power flow through the ladder logic and watch data values change in real time. You can also edit the ladder program, just as you would while offline, with the following restrictions:

- Labels, subroutines, and actions cannot be deleted while online.
- Only one rung can be changed before writing the changes to the Controller.

The following picture illustrates the ladder editor while online to the Controller.

The screenshot shows the 'IndustrialPackaging - Proficy Machine Edition - [Ladder Program Main View]' window. The main area displays a ladder logic program with four rungs. Rung 1 is labeled 'Initialize the message buffer' and contains a 'MOV' instruction. Rung 2 is labeled 'Begin the main program' and contains a 'START' instruction. Rung 3 contains a 'TOF' (Timer Off Delay) instruction. Rung 4 contains a 'TON' (Timer On Delay) instruction. The interface includes a project tree on the left, a parameter box at the bottom left, and a data watch table at the bottom right. Annotations with arrows point to various features: 'Online status is indicated by the color of the target icon.' (pointing to a lightning bolt icon), 'New or edited rungs are indicated with alternately colored and bolded lines until you write changes.' (pointing to rung 3), 'Power flow is indicated with bold colored lines.' (pointing to the bolded line of rung 4), 'Lightning bolts in each corner indicate the Controller's status and change to reflect fault conditions.' (pointing to lightning bolts in the corners), 'Data values are displayed in real time as they change each scan on the Controller.' (pointing to the data watch table), and 'Use the Data Watch to view or set data values for any variable in your project.' (pointing to the data watch table).

Parameter Box
A Parameter Box is where you assign a variable or expression to an instruction or I/O terminal.
Tip: A variable or expression can be dragged from anywhere it exists and dropped onto a Parameter Box.

Variable	Value
MyTimer	PT: 5000; ET: 5000; Q: On; TI: Off
PT	5000
ET	5000
Q	On
TI	Off

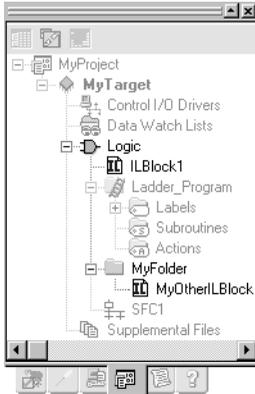
To go online to the Controller's ladder program

- In the  Project tab of the Navigator, right-click the  target and click **Go Online**. (Before you can go online to the Controller, you must first download the project.)
The Controller's ladder program appears in the ladder editor. When the Controller is running, power or state flow is graphically indicated and data values updated.

To write a changed rung to the Controller

- From the Controller menu, choose **Write Changes**. (The Controller menu is available only when the ladder editor is selected.)
The Controller's ladder program is updated with the single altered rung.

INSTRUCTION LIST EDITOR



Navigator: Project tab
IL block nodes

IL logic is one of four languages specified by the IEC 61131-3 standard. The IL editor is an easy-to-use, intelligent, free-form text editor for editing IL blocks in the Machine Edition environment. With the IL editor, you can work on a disk copy of an instruction list (offline) or monitor the execution of an IL block running in the Controller (online). You can also configure the appearance and behavior of the editor under the  Options tab of the Navigator.

Instruction List

IL is a low-level language composed of basic and advanced Math instructions that execute sequentially from top to bottom. IL blocks can also contain labels providing entry points for program execution. In its simplest form an IL can:

1. Load a data value (operand) into the accumulator.
2. Perform an operation on the accumulator, saving the result in the accumulator.
3. Store the accumulator's value to memory (a variable).
4. Do another accumulator load or operation, and so on.

 **Want to know more?** In the Help Index, look up "IL" and choose "IL Editor".

To open an IL block for editing

- In the  Project tab of the Navigator, under the  Logic folder or under a  user-defined folder, double-click an  IL block node.
Your IL block opens in the IL editor.

To create a new IL block

1. In the  Project tab of the Navigator, right-click the  Logic folder or a  user-defined folder, point to **New**, and click **Instruction Language**.
2. (Optional.) Enter a name for the new IL block.
Your new IL block appears as a child node of the Logic folder or a user-defined folder.

NOTE IL blocks can be executed only when called as Actions from an SFC.

Working with the IL editor - Offline

The IL editor interacts with the Machine Edition tools to provide maximum flexibility when editing a program. The following picture illustrates some of the operations you can perform.

The screenshot shows the Proficy Machine Edition IL editor window. The main text area contains assembly code with comments. A variable list on the left shows variables A, B, C, D, and mydint. A smart list is open at the bottom, showing mnemonics like LD, STN, SUB, TAN, etc. The bottom right shows a Feedback Zone with a References tab.

Annotations and their descriptions:

- Drag variables to the IL editor to complete IL statements.** (Points to the Variable List)
- Copy and paste one or more IL statements from the IL editor to other sections or IL blocks.** (Points to the main code area)
- Begin typing for a Smart List to appear (see page 24).** (Points to the start of the ST instruction)
- The Companion automatically provides information on what you click.** (Points to the LD (operand) block)
- Multiple-line block comments to document your IL block.** (Points to the multi-line comment at the top of the code)
- In-line comments for individual IL statements.** (Points to the single-line comments within the code)
- Locate all instances of a variable with the References tab of the Feedback Zone.** (Points to the References tab in the Feedback Zone)

To insert an IL instruction

1. In the IL editor, begin typing.
A smart list appears containing any items (labels, mnemonics, or variables) that are applicable at the current cursor position.
2. Enter or choose from the list the mnemonic you want.

Working with the IL editor - Online

In the IL editor, you can view the execution of an IL block as it runs. This is illustrated in the following diagram.

The screenshot shows the Proficy Machine Edition IL editor interface. On the left, a project tree shows a target named 'Target1' with a sub-tree containing 'ILBK2'. The main editor window displays IL code with a grey background, indicating it is online and read-only. A context menu is open over the code, showing options like 'Cut', 'Copy', 'Paste', 'Delete', 'Undo', 'Redo', 'Find/Replace...', 'Watch', 'Turn On', 'Turn Off', 'Force On', 'Force Off', 'Remove Forces', 'Insert Keyword...', 'Insert Variable...', 'Reset/Source', and 'Properties'. At the bottom, a 'Data Watch' table is visible, showing variables 'MyBOOL' and 'MyDINT' with their current values.

Annotations in the diagram explain the interface elements:

- Online status is indicated by the color of the target icon.** (Points to the 'Target1' icon in the project tree)
- Variable types and their data values are displayed in real time as they change each scan on the Controller.** (Points to the 'Data Watch' table)
- Right-click any BOOL variable to force it On/Off or to turn it On/Off.** (Points to the context menu over the IL code)
- The grey background indicates that the editor is online to the Controller and is read-only.** (Points to the grey background of the IL editor)
- Use the Data Watch tool to view or set data values for any variable in your project.** (Points to the 'Data Watch' table)

To go online to the Controller's IL

1. In the Project tab of the Navigator, right-click the target and click **Validate (F7)**.
2. Right-click the target and click **Download (F8)**.
3. Right-click the target and click **Run (F9)**.
4. Right-click the target and click **Go Online (CTRL+F11)**.

The Controller's IL block appears in the IL editor as read-only and with a grey background. In the IL editor, you can monitor the values of a variable, by clicking the variable. You can also change or force BOOL variable states.

STRUCTURED TEXT EDITOR

ST logic is one of four languages specified by the IEC 61131-3 standard. The ST editor is an easy-to-use, intelligent, free-form text editor for editing ST blocks in the Machine Edition environment. With the ST editor, you can work on a disk copy of a structured text block (offline) or monitor the execution of an ST block running in the Controller (online). You cannot edit an ST block online.

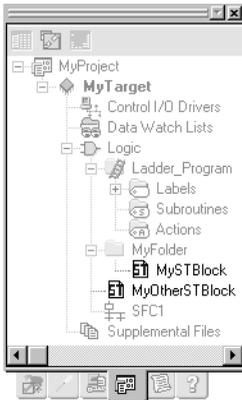
Structured Text

ST is a high-level language composed of basic and advanced Math instructions. ST blocks can be executed as Actions called from SFC Steps, or when called as a subroutine from other ST blocks or ladder programs.

If there is an ST block named 'MAIN', it executes after the SFC programs. That is, after downloading a Logic Developer - PC application, the execution of every scan begins with the ladder program, then any SFC programs, and then the ST block named MAIN.

ST supports a timer, arrays, bitwise access, strings, and User Defined Function Blocks (UDFBs—callable subroutines to and from which you can pass parameters). The Return keyword causes an early exit from a subroutine.

? **Want to know more?** In the Help Index, look up "ST Editor (PC)" or look up "UDFB" and choose "User Defined Function Blocks: an Overview".



Navigator: Project tab
ST block nodes

To open an ST block for editing

- In the Project tab of the Navigator, under the Logic folder or under a user-defined folder, double-click an ST block node. Your ST block opens in the ST editor.

To create a new ST block

1. In the Project tab of the Navigator, right-click the Logic folder or a user-defined folder, point to **New**, and click **ST Block**.
2. (Optional.) Enter a name for the new ST block. Your new ST block appears under the Logic folder or under a user-defined folder.

Working with the ST editor - Offline

The ST editor interacts with the Machine Edition tools to provide maximum flexibility when editing a program. The following picture illustrates some of the operations you can perform.

The screenshot shows the ST editor interface with several annotations:

- Drag variables to the ST editor to complete ST statements.**: An arrow points from the Variable List (containing variables like Accel_Current, Accel_Initial, etc.) to the main editor area.
- Copy and paste ST statements from the ST editor to other sections or ST blocks.**: An arrow points from the main editor area to the Block Properties panel.
- Multiple-line block comments to document your ST block.**: An arrow points to a multi-line comment block at the top of the editor.
- Hover over a variable with the mouse pointer to see its data type.**: An arrow points to the variable 'Status' in the code.
- In-line comments for individual ST statements.**: An arrow points to a single-line comment within the code block.
- The Companion automatically provides information on what you click.**: An arrow points to the Companion window showing details for the 'FOR' instruction.
- Locate all instances of a variable in the References tab of the Feedback Zone.**: An arrow points to the References tab in the Feedback Zone.

To insert an ST instruction

1. In the ST editor, begin typing. To see a list of valid variables, right-click anywhere and click **Insert Variable**. Choose a variable from the list that appears, then press **ENTER**.
2. To obtain a list of valid mnemonics, right-click anywhere and click **Insert Keyword**. Choose a keyword from the list that appears, then press **ENTER**.

Working with the ST editor - Online

In the ST editor, you can view the execution of an ST block as it runs. This is illustrated in the following diagram.

The screenshot shows the Logic Developer interface. On the left, the Project Navigator shows a tree structure with 'MySTBlock' selected. A callout points to the 'MySTBlock' icon, stating: 'Online status is indicated by the color of the target icon.' The main editor window displays a code block with a grey background, indicating it is read-only. The code includes a 'WHILE' loop and a 'MyBOOL' variable. A callout points to the grey background, stating: 'The grey background indicates that the editor is online to the Controller and is read-only.' Below the code editor, a 'Watch' table is visible, showing the current values of variables: 'DintArray' (1, 2, 3, 4, 5, 6, 7, 8, 9, 10), 'Index' (10), 'Sum' (55), and 'MyBOOL' (On). A callout points to this table, stating: 'Variable types and their data values are displayed in real time as they change each scan by the Controller.' At the bottom, a 'Data Watch' tool is shown, with a callout stating: 'Use the Data Watch tool to view or set data values for any variable in your project.'

Right-click any BOOL variable to force it On/Off or to turn it On/Off.

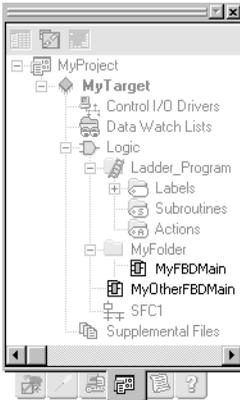
Use the Data Watch tool to view or set data values for any variable in your project.

To go online to the Controller's ST

1. In the Project tab of the Navigator, right-click the target and click **Validate (F7)**.
2. Right-click the target and click **Download (F8)**.
3. Right-click the target and click **Run (F9)**.
4. Right-click the target and click **Go Online (CTRL+F11)**.

The Controller's ST block appears in the ST editor as read-only and with a grey background. In the ST editor, you can monitor the values of a variable, by clicking the variable. You can also change and/or force BOOL variable states.

FUNCTION BLOCK DIAGRAM EDITOR



Navigator: Project tab
FBD nodes

Function Block Diagram (FBD) logic is one of four languages specified by the IEC 61131-3 standard. The FBD editor is a graphical, free-form editor for editing FBD logic in the Machine Edition environment. With the FBD editor you can work on a disk copy of a function block diagram (offline) or monitor the execution of an FBD running in the Controller (online). You cannot edit an FBD online.

Function Block Diagram

A Logic Developer - PC FBD is a named section of Function Block Diagram logic that is compiled and downloaded to the PC represented by the parent target. The graphical FBD language can be used where there is a flow of signals between control blocks.

FBDs can be executed as Actions called from SFC Steps or called from other FBDs, ladder programs, or ST blocks. An FBD can contain instructions, wires, and variables, as well as ladder user-defined function blocks (UDFBs), and ST UDFBs.

Want to know more? In the Help Index, look up "FBD: an Overview (PC)".

To open an FBD for editing

- In the Project tab of the Navigator, under the Logic folder, or under a user-defined folder, double-click an FBD node.

Your FBD opens in the FBD editor.

To create a new FBD

1. In the Project tab of the Navigator, right-click the Logic folder or a user-defined folder, point to **New**, and click **Function Block Diagram**.

A new, empty FBD with the default name "FBDBlk*n*" appears in the Navigator in alphabetical order under its folder.

2. (Optional.) Enter a unique name for the FBD.

Your new FBD appears under the Logic folder or under a user-defined folder.

Working with the FBD editor - Offline

The FBD editor interacts with the Machine Edition tools to provide maximum flexibility when editing a program. The following picture illustrates some of the operations you can perform.

The screenshot displays the Profinity Machine Edition FBD editor interface. The main workspace shows a ladder logic diagram with several rungs. The first rung contains a NOT instruction (A) connected to a coil (C). The second rung contains a coil (Off) connected to a timer block (Charging_Delay) with a time value of 10001. The third rung contains a coil (Off) connected to a timer block (Timer_TotalOff) with a time value of 0. The fourth rung contains a coil (A) connected to a BOOL_TO_INT instruction (B), which is then connected to a coil (A) of an EQ instruction (C).

Annotations and callouts:

- Drag variables to the FBD editor to complete FBD instructions.** An arrow points from the Variable List on the left to the NOT instruction in the diagram.
- Drag instructions from the PC FBD Instructions drawer of the Toolchest.** An arrow points from the PC FBD Instructions drawer on the right to the EQ instruction in the diagram.
- Locate all instances of a variable in the References tab of the Feedback Zone.** An arrow points from the References tab in the Feedback Zone at the bottom to the variable 'A' in the diagram.
- The Companion automatically provides information on what you click.** An arrow points from the Companion window at the bottom to the EQ instruction.

The Variable List on the left includes: Charging_Delay, Enable_Drive, gw, High_Voltage_Cabinet_Far, High_Voltage_Relay, HV_Fans_Timer, Start_Drive, Step1, Timer_Total, Trans1, Wire_587_10, and Wire_587_13.

The PC FBD Instructions drawer includes: Advanced Math, Bitwise Logic, Comment Block, Comparison, Copy, Counters, Math, Program Flow, Timers, and Type Conversion (with sub-items: BOOL_TO_INT, INT_TO_BOOL, INT_TO_REAL, REAL_TO_INT).

The Companion window shows the following text for the EQ instruction:

```

Timer On Delay (TON)
The TON instruction times a delay of PT milliseconds from when an event occurs; that is, from when IN is set to ON(1) and the TON instruction is being solved. Q is initially not solved (Q is set to OFF(0)) when IN is being solved (IN is set to ON(1)). PT milliseconds later, if IN is still being solved (IN remains set to ON(1), the instruction is being solved (Q remains set to ON(1)).
  
```

To insert an FBD text box (Toolchest method)

1. If you have not already done so, open the **Toolchest**.
2. In the Toolchest, choose the **PC FBD Instructions** drawer.
3. Expand the **Comment Block** folder.
4. Drag the text box from the Toolchest onto a blank area of the FBD editor.
You can now enter text in the FBD editor.

To insert an FBD text box (FBD editor method)

1. In the FBD editor, right-click a blank area and click **Insert Instruction**.
2. In the smart list that appears, do one of the following:
Enter the word “text” (without quotes), and then press **ENTER**.
- or -
Scroll down the list, choose **Text**, and then press **ENTER**.
3. You can now enter text in the FBD text box.

To insert an FBD instruction (Toolchest method)

1. If you have not already done so, open the  Toolchest.
2. In the Toolchest, choose the **PC FBD Instructions** drawer
3. Expand an FBD instruction group.
4. Drag an  FBD instruction from the Toolchest onto a blank area of the FBD editor.
5. Assign parameters above (for Call, Counter, and Timer instructions) and beside the FBD instruction, or draw wires to or from the FBD instruction.

To insert an FBD instruction (FBD editor method)

1. In the FBD editor, right-click a blank area and click **Insert Instruction**.
2. In the smart list that appears, enter or choose an  FBD instruction.
3. Assign parameters above (for Call, Counter, and Timer instructions) and beside the FBD instruction, or draw wires to or from the FBD instruction.

Working with the FBD editor - Online

In the FBD editor, you can view the execution of an FBD as it runs. This is illustrated in the following diagram.

Online status is indicated by the color of the target icon.

Right-click any BOOL variable to force it On/Off or to turn it On/Off.

The grey background indicates that the editor is online to the Controller and is read-only.

Use the Data Watch tool to view or set data values for any variable in your project.

Variable Name	Address	Value
Timer_Total	10001	
Wire_704_10		On

Online FBD editing is not supported. However, you can change or force BOOL variable states. While online, you can also watch data variables in the Data Watch tool.

To watch a variable's value

- In the FBD editor, right-click a variable whose value you want to monitor and click **Watch**.

The variable is added to the  Data Watch tool for monitoring.

LOGIC DEVELOPER - PC WEB ACCESS

Machine Edition enables you to view a target's PC logic data with a web browser, such as Microsoft Internet Explorer (version 4.0 or later is required). When a Machine Edition project is downloaded, it automatically launches an integrated web server. Users who connect to the target computer over the web are presented with the Site Index, from which they can select the *Logic Navigator*. (If the target has an HMI, Web Access options for the HMI appear as well.)

The Logic Navigator presents a tree diagram of PC logic stored in the Controller, where you can browse to specific programs and blocks:

- Published SFCs and SFC macros.
- Ladder program.
- Ladder subroutines.
- Actions.
- ST blocks.
- IL blocks

Click any logic block in the tree to view it.

? Want to know more? In the InfoView tab of the Navigator, expand the Web Access library, and then select a topic under the "PC Web Access features" book.

To allow remote access to your project

1. For each target that you want to allow access to, set the  target's Publish Logic property to **True**.
This enables all PC logic except SFCs and SFC macros to be remotely viewable.
2. For each SFC and SFC macro in those targets that you want users to access, set its Publish property to **True**.
For example, to allow users to view an SFC remotely, set the SFC's Publish property to **True**.

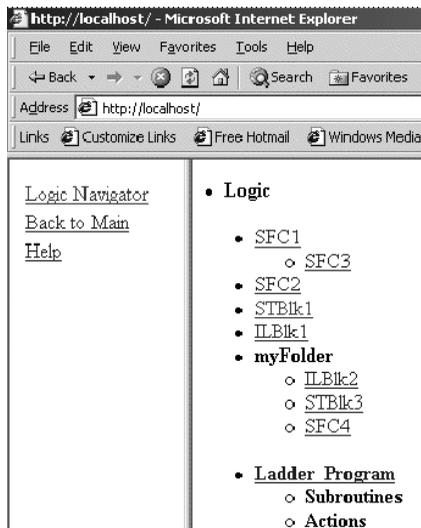
NOTE The target containing the PC logic to be published must be set as the active target and a successful project download must occur.

To access the PC logic Site Index

1. Start Internet Explorer.
2. In the Address box in Internet Explorer, enter the computer name or IP address of the run-time computer.
3. If you are not sure what the run-time computer's name or IP address is, check the properties of the TCP/IP driver in the Control Panel Network setup, or contact your network administrator.
4. If you are testing the project on the local computer, enter "**localhost**" (without the quotation marks) as the Address. This *connects* Internet Explorer with the local computer.

When you connect to the run-time computer, the Site Index appears with a list of links to possible ways to view your process data.

To view PC logic remotely



1. Start Internet Explorer.
2. In the Address box in Internet Explorer, enter the computer name or IP address of the run-time computer.
The Site Index appears.
3. Click **Logic**, and then click **Logic Navigator**.
An index (tree diagram) appears, which contains all published PC logic.

- Click the SFC, SFC macro, ladder program, ST block, or IL block you want to view.

The selected PC logic block appears. See **To allow remote access to your project** above for information on publishing PC logic.

While viewing ladder logic remotely, you can view the values of variables by hovering the mouse pointer over the variable. The value of a BOOL variable on a Coil or Contact instruction is identified by its color:

- If it is green, the variable is set to True (1).
- If it is black, the variable is set to False (0).

You can view the values of variables in SFC Transitions (including Structured Text Transition statements) by hovering the mouse pointer over the Transition. The color of an SFC Transition indicates its result:

- If it is green, its result is True (1).
- If it is black, its result is False (0).

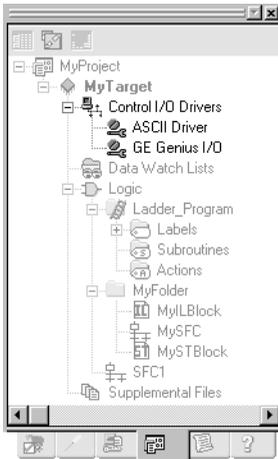
The screenshot shows a Microsoft Internet Explorer browser window displaying a web page with an SFC diagram. The browser's address bar shows 'http://localhost/'. The page has a 'Logic Navigator' sidebar on the left with links for 'Back to Main' and 'Help'. The main content area displays an SFC diagram with three steps: Step1, Step2, and Step1. Step1 contains three actions: N STBIk1, N ILBIk2, and N ILBIk1. Step2 contains two actions: N STBIk3 and N MyLadderAction. A transition between Step1 and Step2 is labeled 'myDint > myDint2 and myDint > myDint3'. A tooltip is visible over this transition, showing the values: myDint : 10, myDint2 : 1, myDint : 10, and myDint3 : 1. Annotations with arrows point to the actions and the transition, explaining how to view their values remotely.

Click to go back to the Logic Developer - PC main menu.

Click an SFC action to view it remotely.

Hover over an SFC transition with the mouse pointer to view its value or values.

CONTROL I/O DRIVERS



Navigator: Project tab
Control I/O nodes

A control program (SFC, ladder, IL, ST, or FBD) generally interfaces with the process it is controlling through some type of physical I/O equipment. For PC-based control systems this usually involves installing an adapter card in the target computer that connects to an industry standard I/O network. In this way, data generated by a process can be acquired and acted upon by the Controller as it executes its program(s). Control I/O drivers provide the software required to connect the Controller with an adapter card and the I/O network of your choice.

? **Want to know more?** In the Help Index, look up "Control I/O Drivers".

I/O Drivers

Logic Developer - PC comes complete with a growing library of drivers. Drivers that require no additional equipment on the target computer are the following:

- **ASCII:** Enables you to use an existing communications port on the target computer to send and receive ASCII formatted data to and from other equipment.
- **Control Peer:** Enables Controllers running on two or more computers to share process data. Messages are passed via ethernet when variable values change.

Other drivers require third-party adapters on the target computer. Logic Developer - PC supports most of today's most popular I/O networks including the following:

- | | |
|--|---|
| ▪ Allen-Bradley RIO | ▪ Honeywell SDS I/O |
| ▪ ASCII^a Driver | ▪ Interbus-S I/O |
| ▪ Control Peer Driver | ▪ Modbus Slave^{1a} |
| ▪ DeviceNet I/O^a | ▪ Modicon MB + Distributed I/O |
| ▪ Ethernet I/O^a | ▪ Modicon Quantum-800 Series I/O |
| ▪ Fanuc CNC I/O | ▪ PROFIBUS I/O^a |
| ▪ GE Series 90-30 I/O^a | ▪ VersaMax Expansion I/O^a |
| ▪ GE Genius I/O² | |

a. Also supported on QuickPanel Control targets

To add a Control I/O driver to a target

- Right-click the  Control I/O Drivers node, point to **New Driver**, and choose a driver from the list.

The new driver appears as a  child node under the  Control I/O node. A default configuration of the driver is also added to the Control I/O tool. You can add up to nine drivers to a  target.

Control I/O Tool

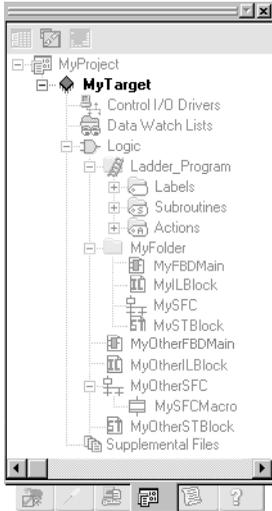


Control I/O

The Control I/O tool is a special editor used to graphically model the I/O equipment driven by Control I/O drivers. Each I/O system is displayed in a tree structure starting with a  driver node branching down to  terminals that represent physical field connections. Between the driver and terminals are nodes representing any  cards,  racks, or  modules included in the system. Field data is shared with the components of your project by mapping variables to each I/O terminal.

 **Want to know more?** In the Help Index, look up “Control I/O Tool”.

CONTROLLER

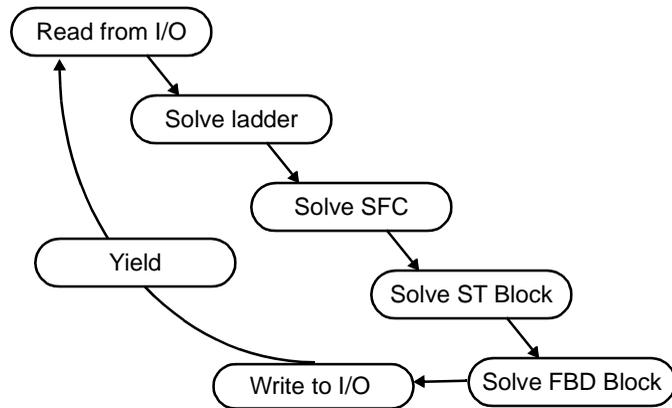


Navigator: Project tab
Target node

The Controller is a run-time engine, hosted on a  target, that performs the following operations in a repeated scan cycle:

- Solves logic in a ladder program, including any called subroutines.
- Solves logic in an SFC, including any called actions.
- Solves ST block named 'MAIN' (if it exists), including any called subroutines.
- Solves FBD block named 'MAIN' (if it exists), including any called subroutines.
- Interfaces with Control I/O drivers.
- Updates the target's run-time database, which is shared with the View Runtime (see page 93).

The following diagram illustrates the Controller's scan cycle.



Logic Developer - PC Run-time Controller Scan Cycle

The Controller can be hosted on a:

- *Windows PC target.* The Controller appears as a Logic Developer - PC system service.
- *QuickPanel View/Control target.* The Windows QuickPanel Control appears as a Logic Developer - PC Controller program.

You specify a target when creating a project. The Controller automatically starts when the target operating system is started.

 **Want to know more?** In the Help Index, look up "PC Logic Controller: an Overview".

Working with the Controller

The Controller's location (IP address or computer name), performance (scan rate or tuning), and behavior (response to faults, watchdog timer, and shutdown procedure) are configured by setting the properties of the target it resides on.

To configure the Controller

1. In the  Project tab of the Navigator, right-click a  target and click **Properties**.
2. In the  Inspector, adjust the properties.

If your project includes an HMI component, properties of the View Runtime can also be set at this time. You can also configure the Controller's properties by selecting **Properties** from the Controller menu. In this case, the Controller Properties dialog box appears. The information in the dialog box is the same as in the Inspector—just represented differently.

The logic component of a project must be downloaded to the Controller before the project can be run.

To download a logic component to the Controller

- In the  Project tab of the Navigator, right-click a  target and click **Download**. All components on the selected target are validated and, if no errors exist, they are downloaded to the specified computer. Errors listed in the Build tab of the Feedback Zone must be corrected before a download can be completed. Logic components are downloaded to the Controller, while HMI components are downloaded to the View Runtime.

Although the Controller service is always running, you still have control over the execution of your control programs. Offline, you can start, stop, and reset execution. When online to the Controller, additional commands are available to pause, single-scan, or enable forces.

Run Mode Store

With the Run Mode Store (RMS) functionality, you can update a target Controller's program without stopping the Controller. It is supported on all Logic Developer - PC targets.

When you attempt to download your project to the Controller, the Download to Controller dialog box appears with the following options.

- **Attempt a Run Mode Store:** Machine Edition attempts a Run Mode Store operation. That is, it tries to download your logic to the Controller while the Controller is running.

- **Stop the Controller, then download:** Machine Edition does not attempt to use the Run Mode Store feature. Instead, it stops the Controller before initiating the download operation.
- **Download Retentive Initial Values:** Choose whether to download initial values of retentive variables, which may have changed since the previous download.

Several system tests and checks are performed when attempting a Run Mode Store operation. If these tests fail, the logic program may not be downloaded to the Controller.

? **Want to know more?** In the Help Index, look up “Run Mode Store (RMS) (PC)”.

To run the Controller

- In the  Project tab of the Navigator, right-click a  target, point to **Online Commands** or **Offline Commands**, and click **Start Runtime**.
The ladder program, SFC program(s), the ST block named ‘MAIN’, and the FBD block named ‘MAIN’ begin executing on the target computer or unit. If the target also has an HMI component (see page 93), the View Runtime is started at the same time as the Controller.
To validate, download and run a project, right-click a  target and click **Download and Start “*targetname*”**.

Warm Standby

The Warm Standby feature provides an extra level of reliability for critical Windows PC and QuickPanel Control applications. It consists of two Controllers running the Logic Developer - PC Controller (a *primary* and *secondary* or *warm standby* Controller), and an I/O switch that determines which Controller is currently connected to the physical I/O.

When a system using warm standby is started, the primary Controller is connected to the I/O, solves logic, and updates the secondary Controller with critical data. The secondary Controller receives the data and monitors the status of the primary Controller. When a fault is detected in the primary Controller, or when the secondary Controller cannot communicate with the primary Controller, the I/O connection is automatically switched to the secondary Controller and the secondary Controller takes over operations.

Working with Warm Standby

You must set up the warm standby hardware before you can configure the warm standby system.

? **Want to know more?** In the Help Index, look up “warm” and choose “Warm Standby”.

To configure Warm Standby

1. In the  Project tab of the Navigator, right-click a  Windows PC or QuickPanel Control target and click **Properties**.
2. In the  Inspector, scroll to the **+Redundant System** group of properties and double-click it to expand the group.
3. Click the Redundant Mode property and click **Warm Standby**.
4. Click the Controller Mode property and choose a value from the list.

Hot Standby

The Hot Standby feature provides a high level of reliability for critical Windows PC applications. It consists of two Controllers running the Logic Developer - PC Controller (a *primary* and *secondary* or *hot standby* Controller) over a GE Genius I/O network. The GE Genius driver automatically handles the switching between the Controllers.

To download to a target with hot standby, the primary and secondary Controllers must be synchronized, that is, they must both be downloaded.

By default, the primary Controller is active and the secondary Controller is the backup. The backup Controller sets itself to active when one of the following takes place:

- A fault of a specified type occurs. (See To configure a Hot Standby, below.)
- A loss of communication occurs with the active Controller via the VMIPCI-5565 reflective memory cards.
- The target's Secondary Switch Time has elapsed.

Both Controllers can read from the network; however, only the active Controller can write to the network.

Working with Hot Standby

You must set up the hot standby hardware before you can configure the hot standby system.

? **Want to know more?** In the Help Index, look up “hot” and choose “Hot Standby”.

To configure Hot Standby

1. In the  Project tab of the Navigator, right-click a  Windows PC target and click **Properties**.
2. In the  Inspector, scroll to the **+Redundant System** group of properties and double-click it to expand the group.
3. Click the Redundant Mode property and click **Hot Standby**.
4. Click the Secondary Address property and enter the IP address (Controller name) of the secondary Controller.
5. Click the Secondary Switch Fault property and choose the type of fault the secondary Controller will become active on.
6. Click the Secondary Node ID property and assign a unique node ID to the VMIPCI-5565 card of the secondary Controller.
7. Click the Secondary Switch Time (ms) property and enter the number of milliseconds the secondary Controller is to wait for communication from the primary Controller before setting itself up as the active Controller.
8. Click the Primary Node ID property and assign a unique node ID to the VMIPCI-5565 card of the primary Controller.
9. Click the Primary Wait Time (ms) property and enter the number of milliseconds the primary Controller is to wait for communication from the secondary Controller before setting its synchronization bit to False.

OPC Servers, Warm Standby, and Hot Standby

Warm standby and hot standby are fully supported by the Logic Developer - PC OPC server for Machine Edition. When a configuration file is detected, the OPC server does the following:

- It generates internal objects so that it recognizes all warm standby or hot standby configurations as logical Controllers to OPC clients.
- It creates predefined OPC items for each logical Controller. These items indicate which physical Controller is active and the statuses of the primary and secondary Controllers.

To configure an OPC server for Hot Standby and Warm Standby

- Edit the “OPCWarmStandby.HTML” file, located in the same folder as LogicPC_OPC.EXE on the computer hosting the OPC Server. This XML file is used for both the Warm Standby and Hot Standby features. By default, this file is located in the “C:\Program Files\Proficy\Proficy Machine Edition\OPC” directory.

4

View*



Navigator: Project tab

HMI component nodes

View is the human-machine interface (HMI) component of Machine Edition, providing the means to organize, track, display, and affect data from your plant or process. With View you create graphical panels, write scripts, configure alarm and logging schemes, and launch the project in View Runtime. View supports HMI creation for Windows 7 and Windows XP (“Windows PC” targets for short), QuickPanel View/Control* targets (“QPV/C” targets for short), and QuickPanel targets.

For Windows PC and QPV/C targets, Web Access features publish data from Machine Edition HMIs for remote viewing over the Internet. Most View tools and editors are common to all targets, but a few are QuickPanel-specific; for details, see “QuickPanel Applications” on page 95.

The following are included with a View HMI component:

-  Panel Editor
-  Script Editor
-  Alarms
-  PLC Access Drivers
-  Language translation support
-  Message groups
-  Logging (Windows PC and QPV/C targets only)
-  OPC Client & Servers (Windows PC and QPV/C targets only)
-  Web Documents (Windows PC and QPV/C targets only)
-  Data Logging Windows (Windows PC targets only)
-  View Runtime (Windows PC and QPV/C targets only)
-  Password security (QuickPanel targets only)

To create a project that includes an HMI component

1. In the  Manager tab of the Navigator, right-click  My Computer and click **New**.
2. In the New Project dialog box that appears: in the Project Name box, enter a name for your project.
3. From the Project Template list, choose “**Configuration Wizard (View / Control)**”.

4. In the preview pane, select the desired parameters for your new application (such as the target type and model).
5. Click **OK**.
Your new project is created and the Project tab opens in the Navigator.

PANEL EDITOR

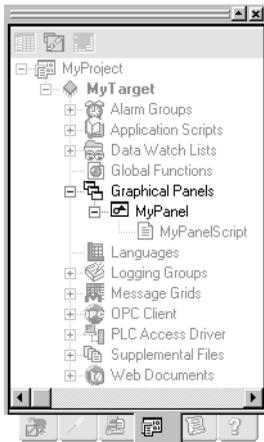
The Panel Editor is used to develop the layout of each *panel* in your HMI application. Each panel is a single screen or window of the HMI. You draw and configure *graphical objects* on the panel to create each screen.

Graphical objects are the various graphical elements of the HMI screen, such as rectangles, polygons, buttons, alarm displays, real-time graphs, and historical trend graphs. All objects are available from the *Drawing Tools* toolbar. Many items perform actions when you tap them at run time.

The Panel Editor displays the current panel as it will appear on the target unit's screen. In addition, the Panel Editor adds a drawing grid of dots to assist you in lining up a series of graphical objects with each other.

- On Windows PC and QuickPanel View/Control targets, many graphical objects can be configured with several types of *Animation* to move, change color, display data, trigger scripts, and receive input during Runtime.
- On QuickPanel targets, the Panel Editor also has a tiling grid. The tiles let you align graphical objects with the touch-sensitive cells on the surface of a QuickPanel unit's display.

? **Want to know more?** In the Help Index, look up "panels" and choose "Graphical Panels".



Navigator: Project tab
Panel Nodes

To add a new panel to an HMI

- In the Project tab of the Navigator, right-click the Graphical Panels folder and click **New Panel**.

A new panel is added beneath the Graphical Panels folder in the Navigator. If you have created a project by using a template, the Graphical Panels folder likely already contains some default panels.

To open a panel for editing

- In the Project tab of the Navigator, double-click a panel node.
The panel appears in the Panel Editor.

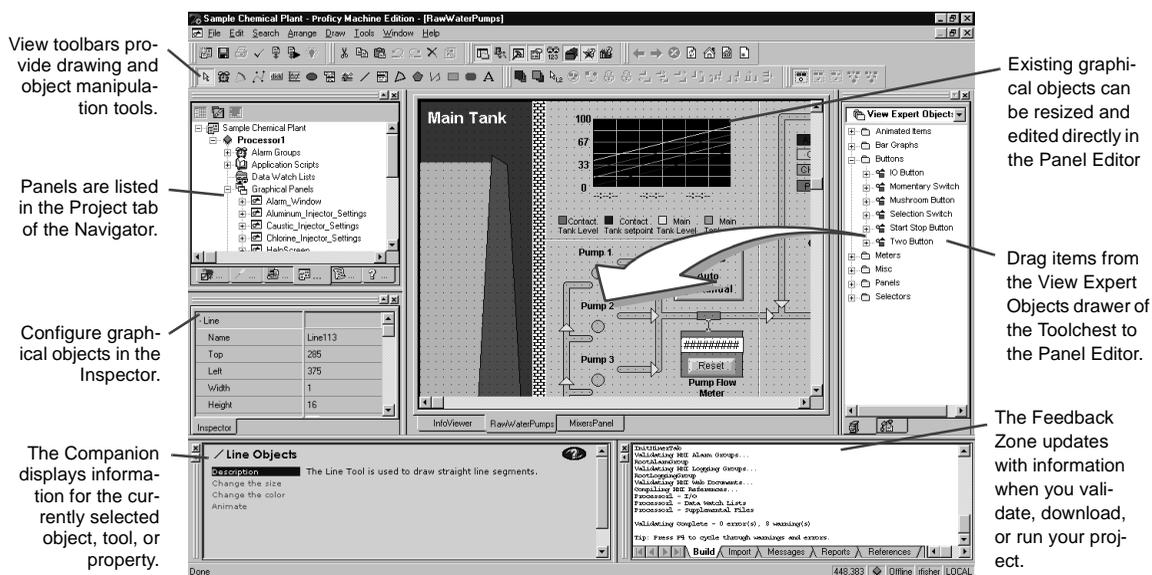
To configure a panel

1. In the  Project tab of the Navigator, right-click a  panel and click **Properties**. The Inspector displays the panel's properties.
2. Set the panel's properties in the Inspector.
For a description of the panel properties, select a property and look in the Companion. Click  in the Tools toolbar to open the Companion window if it is not already open.

Working with the Panel Editor

The Panel Editor interacts with all of the Machine Edition tools to assist you in the construction of graphical panels. The following picture illustrates some of the operations you can perform.

 **Want to know more?** In the Help Index, look up "panels" and choose "Working with Panels".



To draw a graphical object

1. If the Drawing toolbar is open, select the tool you want to use. Otherwise, right-click in the Panel Editor and choose a drawing tool. The Companion displays instructions on how to use the selected tool.
2. Follow the instructions in the Companion to draw your object. After drawing the object, set its properties in the Inspector.

To animate an object (*Windows PC and QuickPanel View/Control targets only*)

- In the Panel Editor, double-click a graphical object. The Animation Properties dialog box appears. Settings for each type of animation are grouped on its own tab. Animations become active when the project is launched with View Runtime.

Message Displays



Navigator: Project tab
Message Group Nodes

The Local Message Display, Triggered Message Display, and Local Image Display graphical objects display different messages or images based on the values or states of associated variables. Each of these three objects is associated with a message group of an appropriate type under the Message Groups folder—an IMD Group, TMD Group, or IID Group.

Each graphical object can be associated with only a single message group, but that message group may be associated with many graphical objects. (On QuickPanel targets, the size of all graphical objects associated with the same message group must be the same; this size is set in properties of that message group.) You can open the grid for an associated message group by clicking the button in the object's "IMD/TMD/IID Group" property.

Entries in a message group are edited in an appropriate grid editor.

- An **Indexed Message Display** graphical object shows text messages that change depending on the value or state of a single associated variable. The colors and appearance of the display can also change. The values and related appearances are set by the display's associated *IMD Group*.

? **Want to know more?** In the Help Index, look up "IMD" and choose "IMD (Local Message Display) group" and "IMD (Local Message Display) grid".

- A **Triggered Message Display** graphical object shows text messages that change depending on the results of boolean expressions. The colors and appearance of the display can also change. The values and related appearances are set by the display's associated *TMD Group*.

Unlike a Local Message Display, you do not associate a variable with a Triggered Message Display. Instead, you specify a series of boolean expressions in the TMD Group grid editor.

? **Want to know more?** In the Help Index, look up "TMD" and choose "TMD (Triggered Message Display) group", "TMD (Triggered Message Display) grid", and "Trigger Expression dialog box (TMD grids)".

- An **Indexed Image Display** shows different images depending on the value or state of a single associated variable. The values and related images are set by the display's associated *IID Group*. An image in an IID Group is

typically a bitmap, imported from a BMP file. Images can also be sets of simple graphical objects (Arcs, Bitmaps, Circles, Lines, Pie Wedges, Polygons, Polylines, Rectangles, Round Rectangles, and Text).

? **Want to know more?** In the Help Index, look up IID and choose "IID (Local Image Display) group" and "IID (Local Image Display) grid".



Navigator: Project tab
Script nodes

SCRIPT EDITOR

The Script Editor is a text editor for the various HMI scripting languages, such as ViewScript, VBScript, and QPScript. On Windows PC targets, you can also use Structured Query Language (SQL) commands to access a database for which an Open Database Connectivity (ODBC) driver is installed.

Scripts

Scripts are short executable programs composed of a sequence of instructions that tell a project, panel, or touch-animated object how to react to events during run-time. There are four types of scripts:

- **Application Scripts:** An application script is associated with an entire target and can be configured to execute when you start or quit the application (in View Run-time) or based on a frequency or condition while the application runs.
- **Panel Scripts:** A panel script is associated with a specific panel in a target and can be configured to execute when the panel opens, when the panel closes, or based on a frequency or condition while the panel is open.
- **Touch Animation Scripts:** (Windows PC and QuickPanel View/Control targets only.) At run time, Touch Animation scripts are activated with keystrokes and on-screen buttons. They are written directly in configuration dialog boxes of graphical objects.
- **Global Function Library Scripts:** (Windows PC targets only.) Global Function scripts are associated with an entire target. By default, they are executed before any other scripts, but they can be configured to run periodically or on a condition. They must be written in an Active Scripting language, like VBScript.

To create, open, and edit an application script

1. In the Project tab of the Navigator, right-click the Application Scripts node, point to **New Script**, and click either **VBScript** or **ViewScript**. (For more information on scripting, see page 76.)
2. Enter a name for the new application script.

A new script is added to the Application Scripts node.

3. Double-click the  script to open it in the Script Editor and begin editing the script.
4. In the Inspector, edit the application script properties.

To create, open, and edit a panel script

1. In the  Project tab of the Navigator, right-click a  panel node, point to New Script, and choose either VBScript or ViewScript. (For more information on scripting, see page 76.)
2. Type a name for the new panel script and press ENTER.
A new script is added to the panel node.
3. Double-click the  script to open it in the Script Editor and begin editing the script.
4. In the Inspector, edit the panel script properties.

To create a touch animation script

1. In the Panel Editor, double-click the object you want to animate. The Configure Animation dialog box appears.
2. Click the Touch tab.
3. Select the Enable Touch Action Animation check box and choose a touch action command that will execute a script.
4. In the large edit box, type the script that will run when the object is touched.

To create a global function library script

1. In the  Project tab of the Navigator, right-click  Global Functions, point to **New Library**, and click **VBScript**.
A new script is added to the Global Functions node.
2. Double-click the script to open it in the Script Editor and begin editing the script.
3. In the Inspector, edit the Global Function Library script properties, in other words, the name.

Scripting Languages

On Windows PC and some QuickPanel View/Control targets, Machine Edition supports two types of scripting languages: the proprietary ViewScript language and VBScript Active Scripting languages (using Microsoft's Active Scripting engine). On other QuickPanel View/Control targets, Machine Edition supports only proprietary ViewScript language.

On QuickPanel targets, Machine Edition uses the proprietary QPScript language (see page 96).

ViewScript scripting language

The ViewScript scripting language consists of the following:

- *Keywords:* View script keywords are a basic set of commands that are used to control the flow of a script and to create local variables.
- *Operators:* View operators are used to assign variable values and to perform mathematical operations
- *Script Functions:* View script functions are commands that can be written into the body of your script to monitor and react to changes in variable values. Functions are grouped in the following categories: ActiveX, Alarm Management, Animation, Application Access, CSV, File Management, Initialization File Management, List and Combo Box, Logging, Miscellaneous, Network, Panel Management, Screen Navigation, Security, Serial Communication, Statistical Process Control, SQL, String, System, Trend Management, Video/Sound, and Web.
- *Comments:* Placing comments in, or between, the lines of your script is useful for debugging your script and for future reference.

? **Want to know more?** In the Help Index, look up "Viewscript" and choose "ViewScript HMI scripting language".

Active Scripting

Active Scripting lets Machine Edition use external scripting languages through a COM interface. Active Scripting languages are supported only on Windows PC and some QuickPanel View/Control targets. Currently, only the VBScript Active Scripting language is supported.

By default, new projects use VBScript. You can override this default by setting a user preference.

A target can contain scripts written in either ViewScript or VBScript. Because troubleshooting such a project can be quite difficult, this practice is not recommended.

? **Want to know more?** In the Help Index, look up "scripts" and choose "View Scripts: an Overview", or look up "scripting" and choose "Active Scripting in View: an Overview".

Validating a script written in an Active Scripting language

Active scripting languages are validated at run-time. You can also validate a project at any time during development to check:

- Syntax for View function calls.
- Syntax for variable, array, and structure element references.

- Limited checks for VBScript syntax.

VBScript Syntax

A detailed explanation of VBScript syntax is beyond the scope of this manual. For more information on the VBScript language usage, syntax and examples, look up VBScript at Microsoft's website (www.microsoft.com).

? **Want to know more?** In the Help Index, look up "VBScript" and choose "VBScript language: an Overview".

Active Scripting Objects

Active Scripting provides a mechanism to extend a scripting language through custom COM objects. Variable syntax is, therefore, limited to using properties and methods on a COM object.

There are four types of objects in VBScript:

- Variable objects.
- Array objects.
- Structure objects.
- Application objects.

? **Want to know more?** In the Help Index, look up "VBScript" and choose "VBScript language: an Overview".

Working with the Script Editor

The Script Editor interacts with other Machine Edition tools to provide maximum flexibility when editing a script. The following picture illustrates some of the operations you can perform.

? **Want to know more?** In the Help Index, look up "editor" and choose "Working in the View Script Editor".

There are many ways to insert script words when writing scripts.

- If you are very familiar with the operators, functions, and keywords used in the Script Editor, start typing your script directly in the Script Editor.
- Otherwise, right-click where you want to insert a function, operator, or keyword and click **Insert**. Then point to the appropriate submenu for the item you want to insert and select it from the list that appears. Submenus are also available for names of the project's variables, panels, scripts, alarm groups and logging groups. For a description of a script function, keyword, or math

A script can be set to activate periodically or on a condition.

Application scripts are displayed under the Application Scripts folder. Panel Scripts are displayed under the panels to which they belong.

The View Scripting drawer of the Toolchest contains script functions, keywords, and math operators that you can drag into your scripts.

The Companion displays help for the currently selected script function, keyword, math operator, or property.

Application Script Properties:

Name	Auto_Valve_Control
Activate	On Condition
Frequency (ms)	500
Condition	TankLevel.LoErb
Conditional Trip	While TRUE

```

Script Created: May 28, 1999
Description:
-----
MainRange := ( MainLevel:MinLmt ) / 60
TankRange := MainRange * 24
IF TankLevel > TankLevel:HiLmt
  ByPass := 1
  TankValve := 1
  WashValve := 1
ELSE
  ByPass := 0
  IF TankLevel < TankLevel:LoLmt
    TankValve := 0
    WashValve := 1
  ELSE
    TankValve := 1
    IF ( TankLevel > TankLevel:HiLmt )
      WashValve := 0
    ELSE
      WashValve := 1
  
```

Companion Window Content:

AlarmGrpAck *almgrpGroupToAck*

AlarmGrpAck *almgrpGroupToAck, numSkipAckPrompt*

Acknowledges all active alarms in an alarm group. A dialog box appears allowing a user to enter an acknowledgment comment, unless numSkipAckPrompt is set.

Supported by: Windows NT and Windows CE targets

operator, while in the Script Editor, move the cursor on the word and press F1.

- All the math operators, script functions, and script keywords are also listed in the Toolchest and can be dragged into your script. The Companion provides a description of each item.

GRID EDITORS

With various grid editors, you can create and edit several items in your HMI targets within a spreadsheet-like table. Among the features that use grid editors are:

- Word Alarm groups
- Bit Alarm groups
- IMD (Local Message Display) groups
- TMD (Triggered Message Display) groups
- Selector Switches
- Bar Graphs
- Language translations

-  Trend displays
-  Passwords list (*QuickPanel targets only*)
- External Keypad assignment configuration

The exact functionality depends on the specific editor, but in general you can:

- Double-click a cell to edit its contents.
- Right-click a cell to perform various operations on the cell.
- Select an entire row (by clicking the grey button row headers on the left), then right-clicking the selection to perform operations on the row

? **Want to know more?** In the Help Index, look up "Working with QuickPanel grid editors". You can also press F1 while working in the grid for a description of that editor.

ALARMS

Alarms are messages that appear on a display in response to specified conditions in the system. These messages typically inform the operator of a situation that requires immediate attention. When a variable satisfies one of its alarm conditions, it goes into an alarm state.

The alarm system for a HMI application consists of several different elements.

Alarm Groups

There are three general types of alarms, organized into three subfolders under the  Alarm Groups folder:

-  **Word Alarm Groups** and  **Bit Alarm Groups** are tables of variables, variable values, and associated alarm messages. When the value of a variable matches an entry in one of these tables, it goes into an alarm state and the appropriate message is added to any Alarm Display objects on the target.
-  **Variable Alarms** (*Windows PC and QuickPanel View/Control targets only*) are sets of variables for which alarm limits have been configured. Unlike Bit or Word Alarms, you assign a variable to an alarm group in properties of the variable itself. Variable alarm groups help you organize related alarmed variables. Also, using more than one variable alarm group makes it easier to manipulate and view alarm data at run time. A history of alarm group data can be logged daily for future analysis. You can also configure e-mail notification if any of a variable alarm group's variables go into an alarm state.



Navigator: Project tab
Alarm nodes

Alarm Display Objects

Active alarms can be displayed at run time with *Alarm Display* objects. At run time, Alarm Displays dynamically update with information about alarm states.

- If you use variable alarm groups (Windows PC and QuickPanel View/Control targets only), you can associate an Alarm Display with a specific variable alarm group. Only variable alarms for alarms for variables in that group and its subgroups will appear on that Alarm Display.
- If you use Bit or Word alarms, alarm states and messages are based on specific values. When a variable and its value match an entry in one of the target's alarm groups, the corresponding message appears in the Alarm Window.

 **Want to know more?** In the Help Index, look up either "Alarms (Windows PC and QuickPanel View/Control): an Overview" or "Alarms (QuickPanel): an Overview".

To set an alarm condition for Variable Alarms (*Windows PC and QuickPanel View/Control* targets only)

1. In the  Variables tab of the Navigator, right-click the variable for which you want to set an alarm condition and click **Properties**.
The Inspector displays the variable's properties.
2. In the Inspector, double-click the Alarming property to set the general alarm parameters (the priority of the alarm, its alarm group, and how long to keep a history of alarm events).
3. Double-click the specific type of alarm (discrete alarms, limit alarms, deviation alarms, and so on) that you want to configure to set its alarm parameters.
These parameters establish the ranges that place the variable in an alarm state.

To add a Variable Alarm Group

- In the  Project tab of the Navigator, under the target's  Alarm Groups folder, right-click the  alarm group to which you want to add the new group and click **New**. Enter a name for the new alarm group.
Your new alarm group appears beneath the alarm groups node. You can now select this new alarm group when configuring properties of the target's variables.

To add a Bit or Word Alarm group

1. In the  Project tab of the Navigator, under the target's  Alarm Groups folder, right-click the  Bit Alarms or  Word Alarms folder (as appropriate) and click **New**.
2. Enter a name for the new alarm group.

Names must be unique within a target. You can double-click the Alarm Group to edit it.

To create an Alarm Display graphical object

1. In the Panel Editor, right-click and click **Alarm**.
The  Alarm tool is selected.
2. Draw an Alarm Object, using the information in the  Companion as a guide.
3. Right-click the Alarm Object and click **Properties**.
The Inspector displays the object's properties.
4. In the Inspector, edit the alarm object's properties as desired.
The Alarm Display objects supported by Windows PC and QuickPanel View/Control are not the same as those supported by QuickPanel targets. The objects, though similar, have distinct functionalities. For help on a specific property, click the property when the Companion window is open.



Navigator: Project tab
Logging Group nodes

LOGGING DATA

(Windows PC and QuickPanel View/Control targets only.)

Logging keeps a history of data changes in your project for later analysis.

Logging groups help sort and organize your project data. By default, all data is logged using the same logging strategy. If you want variables to have different logging strategies, you can create a hierarchy of logging groups.

Logging Strategies define when data is logged for a group of variables. You can configure a group to log data periodically, on data changes, or on a condition. For example, one logging group might log data every minute, whereas another might log data every five minutes.

If you are using a Proficy Historian 2.0 (or later) system, you can specify that logging of data from a View HMI be sent to a Historian Collector. A Collector sends data to a central Historian Server for storage and analysis. With a Collector, logging groups have no effect.

 **Want to know more?** In the Help Index, look up "Logging: an Overview", "Logging Groups: an Overview", "View Historian Collector", and "Data Logging Windows".

To enable logging of a variable

1. In the Variables tab of the Navigator, right-click a variable and click **Properties**.

The Inspector displays the variable's properties.

2. Set the Enable Logging property to **True**.
Logging is enabled for the variable, with RootLoggingGroup as the default logging group.

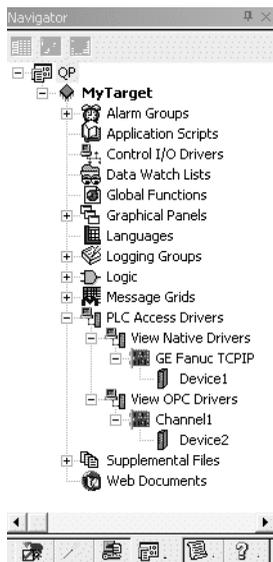
To add a new logging group

1. In the  Project tab of the Navigator, right-click the  logging group node where you want to add the new group, and click **New**.
2. Enter a name for the new logging group.

Your new logging group appears beneath the logging group node.

To view or edit the logging group properties, right-click it and click **Properties**.

During Runtime, the group will log variable data according to its properties. View information about selected properties in the Companion. Click  on the Tools toolbar to open the Companion if it is not already open.



Navigator: Project tab
PLC Access I/O Nodes

PLC ACCESS I/O

PLC Access I/O is used to enable your Machine Edition HMI unit to communicate with I/O devices on the run-time network. Typically, an I/O device is a Controller: a special hardware device that controls input and output for other, simpler devices. Different brands and types of Controllers communicate using different protocols, requiring different drivers.

Drivers

PLC Access I/O is organized into View Native Drivers and View OPC Drivers. The latter are supported for Windows PC and QuickPanel View/Control targets.

The View Native Drivers folder can contain the following:

-  **Drivers** represent the drivers the target uses to communicate with devices. Windows PC and QuickPanel View/Control targets can have many drivers; QuickPanel targets can have only one.
-  **Devices** represent individual devices that a driver communicates with. All devices that a driver communicates with are located in that driver folder. Most drivers can have several devices. This enables you to talk to multiple Controllers, or (on Windows PC and QuickPanel View/Control targets) to get pieces of data from the same Controller at different rates.

The View OPC Drivers folder can contain multiple channels and devices:

-  **Channels** represent drivers that support serial communications between a target QuickPanel View HMI and a PLC device by means of an OPC server. Each channel can communicate with multiple devices.
-  **Devices** represent actual hardware that the HMI communicates with by means of an OPC server. A device can be a Controller, I/O module, PC Control, or serial device.

 **Want to know more?** The IPI lists all of the supported PLC Access drivers. See “View, Supported H/W”. For more information, in the Help Index, look up “PLC Access Driver I/O”, “View Native Drivers”, and “View OPC Drivers”.

To access data values from devices of a PLC Access driver, you must also create variables with a PLC Access data source. The configuration of these variables is unique for each kind of PLC Access driver; for more details, see the online help.

If you want to connect to a GE IP Controller (whose application is created with Logic Developer - PLC), you can use *proxy variables* to simplify configuration of I/O variables in the HMI target. Proxy variables are located in an HMI target and have HMI-only properties, but represent another variable on a GE IP Controller target. A proxy variable’s I/O-related properties (such as its I/O address) are automatically configured based on its source variable in the Controller target.

 **Want to know more?** In the Help Index, look up “Proxy Variables: an Overview”.

To add a View native driver to a target

1. In the  Project tab of the Navigator, expand the  PLC Access Drivers node.
2. Right-click the  View Native Drivers node, point to **New Driver**, and select a driver.

The driver and a single device are added to your project and the Inspector displays the driver’s properties.

3. In the Inspector, configure the driver. If applicable, click the  button in the driver’s Configuration property and enter values in the dialog box.

To add a device to a View Native driver

1. In the  Project tab of the Navigator, under  PLC Access Drivers and under  View Native Drivers, right-click the driver to which you want to add a new device and click **New Device**.

The device, with a unique default name, is added to your project.

2. (Optional.) Enter a different name for the device.

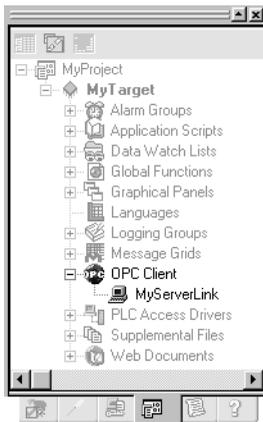
The Inspector displays the device's properties. You can now edit the device's properties in the Inspector.

To add a View OPC driver to a target

1. In the  Project tab of the Navigator, expand the  PLC Access Drivers node.
2. Right-click the  View OPC Drivers node, point to **New Channel**, and select a driver.
The New Channel wizard appears.
3. Follow the wizard to the end, in order to configure the new channel (new driver).
After you exit the wizard, the configured driver is added to your project.

To add a device to a View OPC driver

1. In the  Project tab of the Navigator, under  PLC Access Drivers and under  View OPC Drivers, right-click the driver to which you want to add a new device and click **New Device**.
The New Device wizard appears.
2. Follow the wizard to the end, in order to configure the new device.
After you exit the wizard, the configured device is added to your project.



Navigator: Project tab
OPC Client Nodes

OPC

(Windows PC and QuickPanel View/Control targets only.)

OPC (OLE for Process Control) is an industry-standard protocol that enables software applications to transmit I/O data to each other. The two applications have a client and server relationship. You can think of OPC Servers as being producers of information and OPC Clients as being consumers:

- The OPC Server has data that it makes available to various OPC Clients.
- The OPC Client connects to an OPC Server to gain access to the server's data.

Because OPC is an industry standard, any OPC Client can access data from any OPC Server, regardless of the manufacturer. Machine Edition implements both the client and the server side of OPC communications.

? **Want to know more?** In the Help Index, look up "OPC: an Overview".

Because an OPC Server and OPC Client are simply programs or applications, they can run on the same computer. For example, in your factory you could have

another application that supports an OPC Client and a Machine Edition application running on the same computer. To allow them to share data, you would set up one of them as an OPC Server and the other as an OPC Client, then have the client application establish a connection to the server.

OPC communications rely heavily on Microsoft's OLE and DCOM technology. If the OPC Server and an OPC Client are running on different computers, you must edit the DCOM Security settings on one or both computers.

 **Want to know more?** In the Help Index, look up "DCOM" and choose "Configuring DCOM Security for OPC Communications".

The OPC specifications are set and maintained by the OPC Foundation, a non-profit organization. You can find detailed technical information about OPC on the OPC Foundation web site at www.opcfoundation.org.

OPC Client

The OPC Client driver enables your Machine Edition application to communicate with OPC Servers. This lets Machine Edition act as a client in the OPC Client/Server relationship. The View OPC Client driver supports all interfaces required under the OPC 1.0a, 2.0, and 3.0 specifications.

The OPC Client folder is added to a target when you add an HMI component. Each OPC Server that the application communicates with is located under the OPC Client folder, much like devices of a PLC Access driver.

To access data values from an OPC Server, you must also create and configure variables with an OPC data source.

To add an OPC Server link to the OPC Client folder

1. In the  Project tab of the Navigator, right-click the  OPC Client and click **New Server Link**.
2. Enter a name for the OPC Server Link and click **OK**.
The OPC Server link is added to your project. Edit its properties in the Inspector.

Machine Edition OPC Servers

With Machine Edition OPC Servers, you can share your application's data with external OPC Client applications. This lets your Machine Edition application act as a server in an OPC Client/Server relationship. The Machine Edition OPC Servers support all required OPC interfaces in versions 1.0a and 2.0 of the OPC specifications.

Machine Edition OPC Servers are automatically installed as part of View Runtime and the Controller (sometimes referred to as the Logic Developer - PC Controller). Downloading and running a project to a target automatically enables the appropriate Machine Edition OPC Servers, based on the target's components. For example, if your project has a Logic component, Machine Edition automatically enables the Logic Developer - PC OPC Server.

While configuring an OPC Client depends on the OPC Server, typically each piece of data retrieved from a server is named an "OPC Item". All OPC Clients need to know how to address the data used by the server. The syntax for retrieving OPC items from a Machine Edition OPC Server is as follows:.

Location Syntax

local	<variable name>
remote	<remote machine name>\ <variable name> or <remote machine IP address>\ <variable name>

- *Local* indicates that the appropriate Runtime is located on the same computer as the Machine Edition OPC Server. *Remote* indicates that it is located on a different computer.
- The Machine Edition OPC Servers do not use access paths. Instead, as described above, the path for remote access is included as part of each OPC Item address.



Navigator: Project tab
Web Document nodes

VIEW WEB ACCESS

(Windows PC and QuickPanel View/Control targets only.)

Web Documents provide remote access to Machine Edition project data over an Intranet or the Internet. When a Machine Edition project goes to run time, it automatically launches an integrated web server. (You can configure this web server with the ME Web Server applet in the Windows Control Panel.) Users who connect to this web site are presented with the Site Index, which contains different ways to view process information.

- **Remote Views:** Dynamic graphical recreations of actual panels from the View Runtime display.
- **Custom Web Documents:** Web pages generated using templates with special HTML tags. These tags are periodically replaced with current project data at run time. You can view web documents online or print them for distribution.

- **Variable Monitor Browser:** With this application, you can drag and drop variables from Web Documents servers onto four unique viewing tools for dynamic graphical representations of process data.
- **Remote Variable Inspector:** With this tool, you can select a variable from a list of project variables in a project and immediately receive the current value.
- **Web Control:** With Web Control, you can write changes to a target's data from Web Documents. *Before using this feature, ensure that your intranet installation is secure and that no safety concerns exist with web control of a target.*

🔍 **Want to know more?** In the Help Index, look up "web" and choose "Web Access: an Overview".

To enable remote access to your HMI project

1. For each target that you want to allow remote access to, set the  target's Publish HMI property to **True**.
2. For each panel and variable in the target that you want users to be able to access, set its Publish property to **True**.

For example, to allow users to view a graphical panel remotely, set the panel's Publish property to **True**.

To access a published HMI's Site Index

1. Start Internet Explorer 5.5 SP4 or later.
2. In the Address box in Internet Explorer, enter the computer name or IP address of the Runtime computer.

If you are not sure what the Runtime computer's computer name or IP address is, check the properties of the TCP/IP driver in the Control Panel Network setup or contact your network administrator.

If you are testing the project on the local computer, enter "**localhost**" (without the quotation marks) as the Address. This connects the browser to the local computer.

- When you connect to the run time computer, the Site Index appears with a list of links to possible ways to view your process data.
- The Site Index also includes a link to a page where you can download the Web Documents Browser. With it, you can configure graphs, tables, and ticker tapes to report on variables from several Machine Edition applications simultaneously.

To view an HMI's graphical panels remotely from a web browser

You can also view graphical panels remotely from the Web Documents Site Index.

1. Start Internet Explorer 4 or later.
2. In the Address box in Internet Explorer, enter the computer name or IP address of the run time computer.
The Site Index appears.
3. Click **Remote Views**.
4. Click the panel you want to view.

After a few moments, the panel will load and run. Note that the panel must be published for you to be able to view it remotely. To publish a panel, set its Publish property to **True**.

Because the remote viewer uses a different graphics engine to draw and update a graphical panel, there may be a few differences between the remote view of a panel and its actual appearance on an HMI. For example, historical trend objects on QuickPanel View/Control panels are not supported in remote views.

To add a new Custom Web Document

1. In the  Project tab of the Navigator, right-click the  Web Documents node and click **New Web Document**.
2. Enter a name for your new document.
Your new  Custom Web Document appears beneath the Web Documents node. You can edit properties of the Custom Web Document in the Inspector.

To edit a Custom Web Document

1. Outside Machine Edition, create an HTML, XML or text (txt.) document.
2. In the  Project tab of the Navigator, right-click the  Custom Web Document you want to edit and click **Web Document Files**.
3. In the Web Document Files folder that appears, place your HTML document and any logos or pictures your HTML document uses.
The Web Document is created from these files when you download your project. You can view Custom Web Documents from the Site Index.



Navigator: Project tab
Languages node

LANGUAGES FOLDER

The  Languages item opens a grid that enables you to configure multilanguage support for an HMI. At development time, you set up simple translations between an original piece of text (named the “Source”) and one or more languages. At run time, if the target’s Language Translation property is set to Enabled, all text on the HMI’s graphical panels is translated based on one of the language columns in the grid—the “Current Language”. That is, every occurrence in the original text is replaced with its corresponding item in the Current Language.

During a validation or download, Machine Edition also checks the Source column against all graphical panels to which you have made changes. Text entries (in text objects and button objects) that do not appear in the Source column are automatically added to the grid as rows with empty translations for all languages.

The default language column used for translation is typically indicated with yellow shading. You can also specify the Source language as the default language, in which case the shading does not appear. How language translation works depends on the target type:

Windows PC and QuickPanel View/Control targets: Machine Edition includes the entire language table when downloading the project to the target. Translation occurs dynamically at run time. The first language used for translation is the default language. Within HMI scripts, you can change the current language used for translation by calling the SetLanguage script function.

QuickPanel targets: Translation occurs only during the download operation, based on the currently-specified default language. You cannot dynamically change the current language at run time.

? **Want to know more?** In the Help Index, look up “Languages folder”.

Working with the Languages editor

When you double-click a target's  Languages node, the Languages grid appears.

The "Source" column contains the original words and phrases to be used during the translation process, as they appear in the target's graphical panels.

This column's shaded yellow background indicates that it is the default language. If no column has yellow shading, then the Source language is the default.

Source	Italian	French	Japanese
Tank	Serbatoio	Réservoir	タンク
### # C			### #
0 FT	0 piede	0 pied	ゼロ フット
24 FT	24 piedi	24 pieds	二十四 フット
A			ア
AL:			アル

Insert Row	Ctrl+F3
add Variables	Ctrl+F10
Insert Column	Ctrl+G
Delete Column	Ctrl+J
Set Default Column	Ctrl+L
Set Column Font ...	Ctrl+Alt+F
Rename Column	Ctrl+N
Find ...	Ctrl+F
Copy Cell	Ctrl+C
Paste	Ctrl+V

To enter a value into a cell, double-click it (or select the cell and start typing).

If both the development and run time computers support UniCode, you can use non-English character sets in the grid (such as kanji, pictured). For example, In Windows XP you can use the IME, configured in the Regional Settings setup in the Control Panel.

The Languages Editor

To edit or enter the translation of a particular word for a language, double-click its cell (or select the cell and either start typing or press F2). For a menu of commands, right-click anything in the Languages grid.

To add a new language column

1. Right-click in the Language Grid and click **Insert Column**.
A new, empty column is added to the grid.
2. Enter the name of the new language in the column's header text box. To cancel the addition of the column, leave the text box blank and click outside the column header (or press **ESC**).
You can now edit the translation for the new language as normal.

To rename a language column

- Right-click in the Languages Grid and click **Rename Column** (or double-click the column header).
Enter the new name for the language in the column's header text box.

To move a language column

1. In the Languages Grid click and hold the header of the column you want to move.
2. Drag the mouse to the place where you want to move the column, then release the mouse button.

To update entries in the Source column with new text entries in graphical panels

- In the Navigator, under the target whose Languages grid you want to update, right-click the Languages node and click **Update Source Column**.
View searches every graphical panel in the HMI (regardless of whether or not the panel has changed since the last validation) and ensures that all text entries can be found in the Languages grid.

To change the display font for a language

1. Right-click in the Languages grid, and click **Set Column Font**.
2. In the Font dialog box that appears, select the font you want to use to display the entries in the Languages grid, along with the font style and other attributes.
The grid's cells automatically resize to fit the height of the largest font used in the grid.
3. Click **OK** when you are done.
The selected font set is used to display the translated text both in the Languages grid and (in Windows PC and QuickPanel View/Control targets) the downloaded HMI. Other font attributes—such as font sizes and styles—affect only how text appears in the grid at development time.

To set the default language column

- In the Languages grid, right-click the column you want to set as the default language and click **Set Default Column**.
The yellow highlight moves to the selected column. The next time you download the project to the target with translations enabled, the default language will be used by default. Note that you can also set the Source column as the default.

To delete a language column from the grid:

- In the Languages grid, right-click the column you want to remove and click **Delete Column**.
The column and all its translations are deleted.
This operation cannot be undone. Before deleting a language, ensure that you no longer need information on translated words.

VIEW RUNTIME

(Windows PC and QuickPanel View/Control targets only.)

View Runtime is the program that runs your project's completed HMI (human machine interface) on a target computer. It uses the files downloaded to the target during the download process. It is a separate application from Machine Edition; while you can start View Runtime from within Machine Edition, you do not need to be running Machine Edition to use it.

? **Want to know more?** In the Help Index, look up "runtime" and choose "View Runtime: an Overview".

Networking

View Networking provides you with the flexibility to exchange data between run time stations. Based on client-server relationships, and supporting a distributed database system, View Networking allows access to any variable or alarm group running on any other target on the computer network.

? **Want to know more?** In the Help Index, look up "networking" and choose "View Networking: an Overview".

To run a project from Machine Edition

- In the  Project tab of the Navigator, right-click a  target and click **Download and Start**.
All components on the selected target are validated, downloaded to the specified computer, and View Runtime is started. Errors listed in the Build tab of the Feedback Zone must be corrected before a download can be completed. HMI components are downloaded to View Runtime while logic components are downloaded to the Controller.

To run an HMI project outside Machine Edition

- On the **Start** menu, point to **Programs, Proficy, and Proficy Machine Edition**, and click  **View Runtime**.

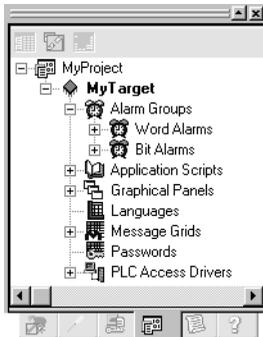
Runtime starts with the last project that was downloaded to that computer. View Runtime opens with all graphical panels configured to appear at startup (that is, all graphical panels whose Visible at Startup property is set to True).

To interact with View Runtime

You can configure graphical objects so that you can change variable values, trigger graphical animations, and execute custom scripts with a simple touch or click.

You can monitor variable values, scripts, drivers, and the integrated web server in Runtime through interfaces that can be opened using the following shortcut keys:

To ...	Press ...
log in	CTRL+L
log out	CTRL+U
close Runtime	CTRL+Z
inspect and change variable values	CTRL+I
view and debug scripts	CTRL+G
view driver diagnostic information	CTRL+D
open a panel	CTRL+O
monitor the Web Documents server	CTRL+W
display version information	CTRL+B



Navigator: Project tab
QuickPanel nodes

QUICKPANEL APPLICATIONS

QuickPanel applications are HMI (human-machine interface) applications for QuickPanel units. View supports development of HMI applications using standard View and Machine Edition tools.

Development of an HMI for QuickPanel units is nearly identical to that of Windows PC and QuickPanel View/Control units. Some of the differences are as follows:

- **Graphical Objects:** QuickPanel units support a slightly different set of graphical objects than Windows PC and QuickPanel View/Control targets. Also, QuickPanel units do not support Animation beyond that inherent in each type of graphical object.
- **QuickPanel Panel Editor:** The Panel Editor for QuickPanel applications includes a tiling grid. This lets you easily align graphical objects with the touch-sensitive cells on the QuickPanel unit's screen.
- **Alarms:** QuickPanel units support only Bit and Word Alarms.
- **Scripts:** Scripts in QuickPanel applications use the proprietary QPScript language. QuickPanel applications do not support ViewScript or VBScript. For more information, see page 96.
- **Integration with Logic Developer - PLC:** If you have Logic Developer - PLC installed, you can use variables defined on a GE IP Controller target directly in your QuickPanel application. To use this feature, both the GE IP Controller target and the QuickPanel target must be part of the same project. You must also configure an appropriate PLC Access driver and device on the QuickPanel target, setting its PLC Target property to point to the GE IP Controller target. (For more information about using GE IP Controller variables, see *GFK-1918: Getting Started Logic Developer - PLC.*)

❓ **Want to know more?** In the Help Index, look up "QuickPanel" and choose "QuickPanel Applications: an Overview", "QuickPanel Target", and "QuickPanel vs. Windows PC and QuickPanel View/Control application development".

External Keypad Assignment

Some QuickPanel units support an external keypad. You enable the keypad by setting the target's External Keypad property to **True**; this enables keypad configuration elsewhere in the application. There are two ways to assign an action to a keystroke:

- In the graphical panel, graphical objects that can have an assigned keystroke display a Key Assignment property. In this property, you can select a key that will activate that graphical object just as if an operator tapped it on the display.

- Actions not associated with a graphical object that appears in the panel are configured in the Keypad Assignment grid. Each panel has a separate Keypad Assignment grid.

To open the Keypad Assignment grid for a panel, select the panel's Keypad Assignment property and click its  button. Each row of the grid specifies the action to perform when the user presses the indicated key while that panel (and only that panel) is open.

? **Want to know more?** In the Help Index look up “keypad” and choose “Keypad Assignment Grid”. Also look up QuickPanel and choose “Working with QuickPanel grid editors”.

Scripts on QuickPanel Targets

Scripts are sets of instructions that tell a project or panel how to react to events during Runtime. Scripts on QuickPanel targets are written in a proprietary language named “QPScript”. You create QuickPanel scripts with the same View Script editor as Windows PC and QuickPanel View/Control targets (see page 75).

QuickPanel applications use two types of scripts:

Application Scripts: Application scripts are associated with a QuickPanel target. These scripts can be executed any time your project is running on the target platform. Application scripts are located under the  Application Scripts folder in the Navigator.

Panel Scripts: Panel scripts are associated with a graphical panel on a QuickPanel target. All of a panel's scripts are located under its  Panel node in the Navigator.

? **Want to know more?** In the Help Index look up “QuickPanel” and choose “QuickPanel Scripts: an Overview”.

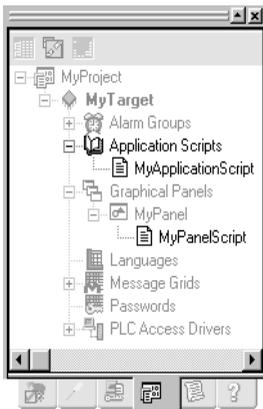
QPScript Language

The QPScript language is composed of the following elements.

Statements: There are three general kinds of statements in a QPScript script:

- *Assignment statements* set a variable's value to the result of an expression.
- *Function calls* execute one of the QPScript functions to perform a specific operation.
- *Branching statements* change the flow of script execution.

Keywords: The elements that make up these statements can be broken down into the following categories:



Navigator: Project tab
QuickPanel Script Nodes

- *Operators* are used in mathematical expressions and conditional expressions.
- *Functions* are predefined routines that perform various operations in the QuickPanel application.
- *Keywords* are other reserved words used in the QPScript language.

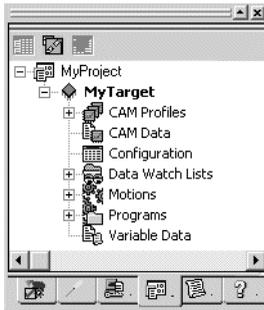
Expressions: There are two general kinds of expressions in QPScript:

- *Mathematical expressions* produce numerical results.
- *Conditional expressions* produce boolean results and are used in if-then-else clauses.

? **Want to know more?** In the Help Index, look up “QPScript” and choose “QPScript Language Reference”.

5

Motion Developer*



Navigator: Project tab
Motion device nodes

With Motion Developer, you can easily program GE's S2K or Whedco's IMC, IMCjr (IMJ), and SMJ series of motion control devices. With Motion Developer, you can:

- Run easy-to-use wizards to set up and configure a complete motion control system in a few mouse-clicks.
- Add motion Controller targets to a Machine Edition project.
- Edit motion blocks and programs for motion Controllers with the Motion Developer Script editor.
- Download programs and motion blocks to motion Controllers through a serial port or a DeviceNet network.
- Work online with a downloaded motion application, viewing and monitoring variable values as the motion Controller's programs run.

Motion Developer is fully integrated with the Machine Edition environment and Machine Edition tools.

To create a Motion Developer project

1. In the  Manager tab of the Navigator, right-click  My Computer and click **New**.
The New Project dialog box appears.
2. In the Project Name box, type a name for your project.
3. From the Project Template list, choose either the **Motion Controller** or **Motion Drive** template and click **OK**.
When you select a template, a preview of what it includes is displayed.
4. Select the Series, Model, and Motor that corresponds with the Controller or drive hardware and click **OK**.
Your new project is created and the Project tab is opened in the Navigator. If you later need to change the Series, Model, or Motor, run the *Target Configuration ▶ Controller & Motor Setup* wizard from the Main Wizard home page of the Motion Developer target. (You can get to the Main Wizard by right-clicking the target and selecting **Main Wizard**.)

ABOUT MOTION DEVELOPER

You use Motion Developer to develop *motion applications* on Windows-based computers. Motion applications are loosely defined as a series of motion device targets with programs, motion blocks, and configuration data. These applications can then be downloaded to the motion Controller hardware, either directly (through the computer's serial port) or through a DeviceNet network.

In general, you go through the following phases when developing a motion application:

1. Create a new Machine Edition project.
2. Add a Motion Controller or Motion Drive target to the project, depending on the kind of motion device.
3. On each target, create the configuration script and (if appropriate) the programs and motion blocks for the motion Controller or drive. The Motion Wizards (see page 106) can make this easy.
4. Validate and download the application to the motion Controller.

The Motion Expert wizard takes you through these items step by step.

Supported motion control devices

Motion Developer supports the following lines of motion control devices.

GE IP motion control devices	S2K Controller/drive devices S2K drive-only devices
Whedco motion control devices	IMC motion Controllers IMCjr (IMJ) motion Controllers SMJ drives

For more information, visit www.ge-ip.com.

Communication between the development computer and motion control devices is supported over the COM port and over DeviceNet networks.

MOTION APPLICATIONS

A typical Motion application consists of a single target. However, your project can have as many targets as you want.

Motion targets

A  *Motion target* represents a run-time destination motion device that you program and configure from Machine Edition. You need a separate target for each physical motion device you configure, whether it is a Controller or a drive-only device.

When adding a Motion target, you specify either a *Controller* or a *Drive* target. This determines the models of Controller you can select in its Controller Type property.

- **Drive** targets represent drive-only motion devices. This includes **GE S2K drive-only** devices and **Whedco SMJ** devices. These targets have a *Configuration script* only (page 102).
- **Controller** targets represent Controller/drive motion devices. This includes **GE S2K Controller/drive** devices and Whedco **IMC** and **IMCjr (IMJ)** devices. These targets have a *Configuration script* (page 102), *Motion Programs* (page 103), and can have several *Motion Blocks* (page 104). In addition, they can make use of a *CAM Profile* (page 112).

To edit configuration scripts, motion programs, and motion blocks

1. In the  Project tab of the Navigator, under the motion target, right-click  Configuration, or the  motion program or  motion block you want to edit and click **Open**.
2. Create or edit the program in the appropriate editor.
3. After editing an item, download the program or project to the target hardware.

To switch between a wizard editor and the motion script editor

Motion blocks and Motion programs can be edited either in a graphical wizard or (for advanced users) in the motion script editor.

- In the  Project tab of the Navigator, under the motion Controller target's  Programs folder, right-click the  motion program or  motion block whose editing method you want to change.
 - If the item is set up to be edited in a wizard and you want to edit its script directly, select **Text View**.
 - If the item is set up to be edited in scripts directly and you want to return to using the wizards, select **Flowchart View** or **Calculator View**, as appropriate.

When switching back to the graphical wizard, changes you made in the script editor will be lost. The item will return to the state it was in when last edited in the wizard.



Navigator: Project tab
Configuration node

Motion configuration

The motion Controller  configuration node represents a script used to configure the motion Controller when the application is downloaded. This script is used every time the entire application is downloaded to the motion Controller. It can be downloaded any number of times. Typically, it sets default values for registers and performs any necessary initialization commands.

- You can create and edit a configuration script with the Motion Configuration wizard (right-click  Configuration and click **Wizard**). This takes you through motion Controller configuration step by step, creating a script based on selected motor and axis parameters.
- If you need more complexity, you can edit the script manually. You can do so in the motion script editor (right-click  Configuration and click **Open**).

 **Want to know more?** In the Help Index, look up "Motion Controller Configuration".

Variable Data

The  Variable Data node contains commands that configure variable registers for the target motion Controller. These include all VB, VF, VI, and VS registers. Typically, these settings are acquired when you import a motion application from a motion Controller unit and specify that variable values are included with the import operation.

Only variables with data types and ranges specified during the import operation are included. Later, when the application is downloaded to a motion target, these

commands are executed on the connected motion Controller, duplicating the imported variables.

? **Want to know more?** In the Help Index, look up “Variable Data (Motion Controller)”.

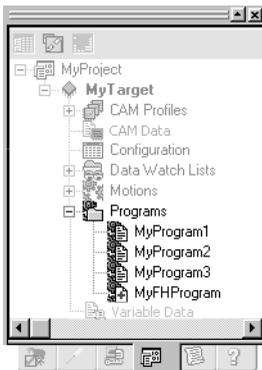
CAM Data

The  CAM Data node contains commands that configure the CAM table for the target. The CAM table defines a CAM profile, a curve that specifies the position of a follower servo motor as a function of the master’s position.

Typically, these settings are acquired by including CAM points when validating a target. You may also acquire CAM Data settings when you import a motion application from a motion device that has CAM data. Either way, Motion Developer automatically handles most CAM Data operations, so you should not have to edit CAM Data settings at all.

CAM profiles are described in greater detail starting on page 112.

? **Want to know more?** In the Help Index, look up “CAM Data (Motion Controller)”.



Navigator: Project tab
Motion Program nodes

Motion programs

A  program represents one of the motion Controller’s programs. Programs are scripts that control the operation of a motion Controller. With programs, you can create complex behaviors for the Controller.

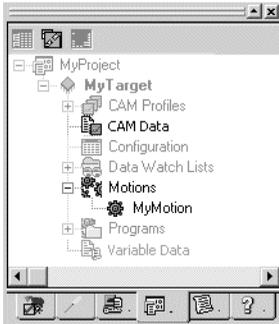
Motion Controller targets have four programs. These are added to the target when you first create it. The last program ( Program 4) is the fault-handling program, which determines the behavior of the Controller when an exception or fault occurs. All programs for a target are located in its  Programs folder.

? **Want to know more?** In the Help Index, look up “Fault-handling motion program”.

You can edit programs either in the Application Builder (by default) or the Motion Script Editor.

- The Application Builder is a wizard that lets you visually specify the program flow as a flowchart. For details, see page 108.
- With the Motion Script editor, you can edit the script for a motion program manually. For details, see page 105.

? **Want to know more?** In the Help Index, look up “Motion Programs”.



Navigator: Project tab
Motion block nodes

Motion blocks

A  Motion node represents a motion block for the motion Controller target. Motion blocks are defined motions (that is, motion scripts) that can be called by motion programs. With motion blocks, you can create complex motions—such as blended moves—without the need for conditional or WAIT statements.

- You can run motion blocks separately with the Controller Functions wizard (see page 107).

You can create a motion block with the New Motion Block wizard (right-click the Motions folder and click Wizard). This creates a complete motion block based on your motion system.

- Advanced users can create an empty motion block by right-clicking the Motions folder and clicking **Add**.

After creating a motion block, you can edit it by right-clicking the motion block and clicking **Open**. You edit motion blocks and motion programs either with the Motion Calculator (see page 110) or in the Motion Developer Script editor (see page 105).

Motion blocks have the following characteristics and restrictions:

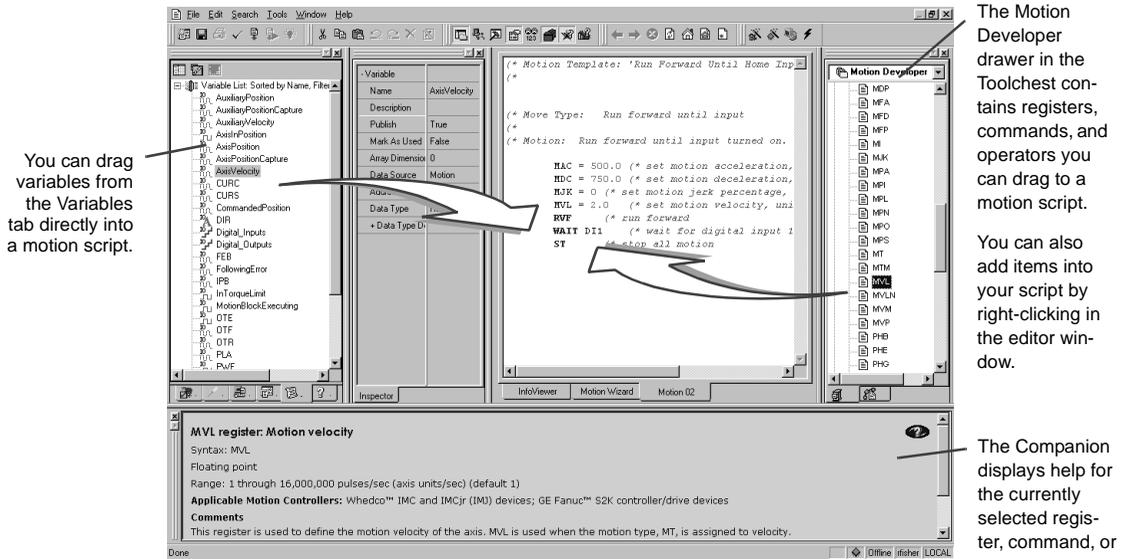
- Motion blocks complete executing one line of code before proceeding to the next line of code.
- Only one motion block per axis can run at the same time as the executing program or programs.
- When a motion block is executed, it overrides the currently executing motion block.
- You cannot use labels in a motion block.

 **Want to know more?** In the Help Index, look up "Motion blocks".

MOTION SCRIPTS

The Motion Script Editor interacts with other Machine Edition tools to provide the most flexibility when editing a script. The following diagram illustrates some of the operations you can perform.

? **Want to know more?** In the Help Index, look up “editor” and choose “Working in the Motion Script Editor”.



There are several ways to insert items when writing motion scripts.

- If you are very familiar with the registers, commands, and operators used in Motion scripts, start typing your script directly in the Script Editor.
- If you need help with script keywords, you can add items with right-click menus. Right-click where you want to insert an item and click **Insert**. Then, point to what you want to include in your script and make a selection from the list that appears. For a description of a register or command in the Script editor, move the cursor on the word for which you need help and press F1.
- The Motion drawer in the Toolchest contains items that you can drag into your script. Leave the Companion window open for a description of each item.
- You can also drag variables directly from the Variables tab of the Navigator into your script.

MOTION WIZARDS

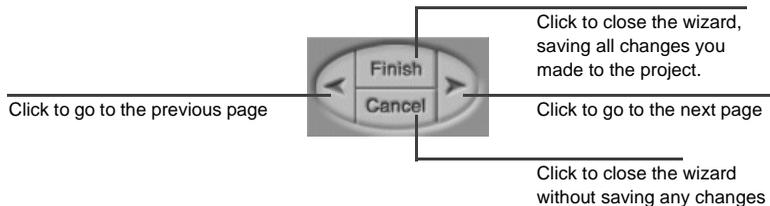
Wizards are a series of screens that take you, step by step, through the creation and configuration of your motion Controller application. In Motion Developer, wizards appear in separate windows, so you can switch back and forth between them.

The most commonly used Motion Developer wizards are as follows.

Main Wizard	Provides access to several other wizards and tools for a motion Controller target. Especially useful is the Motion Expert wizard, which takes you through the creation of a complete application step by step.
Application Builder	Enables you to create motion programs as a visual flowchart. See page 108.
Motion Calculator	Enables you to define simple, incremental motions for motion blocks. See page 110.
Controller Functions	Provides access to several tools and online operations for the motion Controller.

Navigating through Motion wizards

Pages of a Motion Developer wizard work just like HTML pages in a web browser. Click buttons and hypertext links to move from page to page or to perform the labeled task. Within a single wizard, you can move between steps by clicking buttons in the top left corner of each page:



Buttons may be disabled depending on the situation. For example, if you are using the AUTOTUNE page, the “Cancel” button is disabled.

Depending on the page, two additional buttons may be enabled in the top right corner:



Click to jump to the first page in the wizard.



Click for more specific help on the current wizard or the current step.

Many wizard pages contain tables of parameters. Edit the parameters based on your system needs, then click **Next**. When editing parameters, you can also do the following:

- Click **Use Last Saved Values** to revert back to the last values saved for this wizard page. Click **Use Default Values** to fill in parameters with default or calculated values.
- Click a parameter's  button (to the right of its entry text box) for a short description, including its minimum, maximum, and default values. Click the  button for a more detailed description of that parameter.

Wizards for different targets, motions, and programs open in separate windows. You can switch between windows by clicking the tabs at the bottom of the editor window display.

Two of the most important wizards are the Main Wizard and the Controller Functions/Drive Functions wizard:

- The *Main Wizard* page provides access to most other major Motion Developer wizards and tools. This includes the *Motion Expert* wizard, which takes you through the creation of a complete Motion Developer project step by step. To open the Main Wizard page for a target, right-click the  target and select **Main Wizard**.
- With the *Controller Functions* and *Drive Functions* wizards, you can perform diagnostic and other operations on the motion Controller. Functions on these wizards have no effect if the local computer is not connected to a motion Controller. To access these wizards, double-click the target you want to perform the functions on or, while working with a target, click the  Controller/Drive Functions button.

Want to know more?

- For a description of the wizards available from the Main Wizard page, in the Help Index, look up “Main Wizard page”.
 - For a description of features available from the Controller Functions wizard page, in the Help Index, look up “Controller Functions wizard”.
-

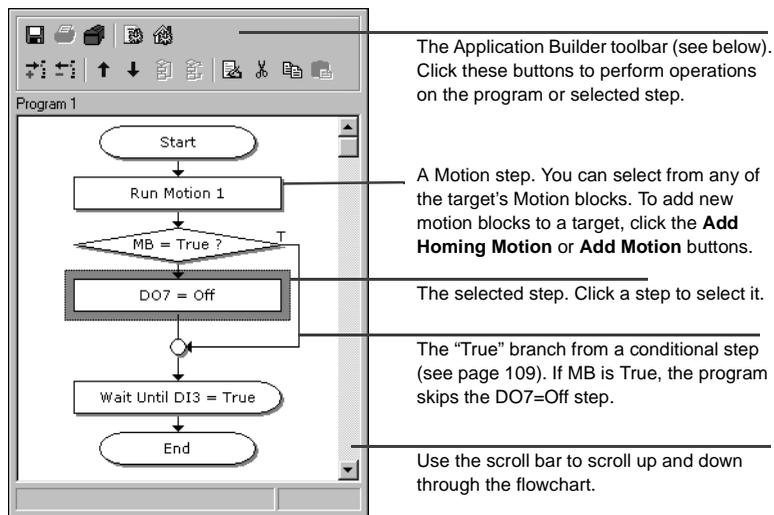
SPECIAL WIZARDS

The Application Builder and the Motion Controller wizards deserve special mention. They enable you to edit parts of your motion application in graphical editors.

Application Builder flowchart editor

With the Application Builder, you can edit your motion programs as a simple flowchart. This makes it easy to see exactly how your finished application will run.

The flowchart editor looks something like this:



The following two buttons on the toolbar open wizards that enable you to add new motion blocks to the target.

- Click **Add Homing Motion** to add one of many predefined homing motions.
- Click **Add Motion** to add a predefined non-homing motion, or to use the Motion Calculator (see page 110).

After adding the desired motion blocks, click **Finish** to return to the flowchart editor.

? Want to know more? In the Help Index, look up "Motion Application Builder".

Basic flowchart step types The flowchart is composed of the following basic step types.

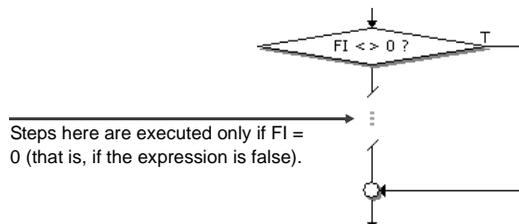
- **Start and End steps:** Flowchart execution begins at the Start step. When execution reaches the End step, the program stops.
- **Logic steps:** Performs a logic-oriented action, such as setting or resetting a particular register.
- **Motion Program steps:** Performs a motion program action, such as starting or stopping a program.
- **Motion block steps:** Runs a motion block.
- **Delay steps:** Pauses program execution.
- **Conditional and Connector steps:** Evaluates a given condition and branches execution depending on the results. For more details, see Conditional Steps on page 109 (or look up “Connecting and disconnecting conditional steps” in the online help).
- **User-defined steps:** Performs a piece of motion script entered by you.
- **Remark steps:** Enables you to insert a comment into the flowchart. Remarks are included in the script downloaded to the motion Controller.
- **Comment steps:** Enables you to insert a comment into the flowchart. Comments are *not* included in the script downloaded to the motion Controller.

Conditional steps are described in greater detail below.

? **Want to know more?** In the Help Index, look up “Application Builder step types”.

Conditional steps

Conditional steps represent a simple conditional statement. When program execution reaches a conditional step, it branches depending on a specified expression:



- If the expression is **True**, execution branches to the right (indicated by a “T”).
- If the expression in the statement is **False**, execution branches downward.

You can choose from several conditional expressions, including several that refer to fault registers. Selecting a conditional step or connector highlights the line connecting it to its partner.

To insert a new step into the Application Builder flowchart

1. In the Application Builder flowchart, select the step immediately preceding the spot where the new step will go.
To add a step to the beginning of the flowchart, select the Start step.
2. Click the  Add Command button on the Application Builder toolbar.
In the menu that appears, point to the type of command or step you want to add, then select the exact step from the submenu. Connector steps, Comments, and User-defined steps are found under the **Standard Elements** option.
Some command types may be unavailable depending on your project. For example, you cannot add a Motion step until the target has some motion blocks.
3. The new step appears after the originally selected step.
If necessary, you can move the step to a different position, or (if appropriate) edit the step's parameters.

To edit a step in the Application Builder flowchart

- In the Application Builder, do one of the following:
 - Double-click the step.
- or -
 - Right-click the step and choose Edit (if the step is editable).
- or -
 - Select the step, then click  Edit Command on the Application Builder toolbar.
- or -
 - Select the step, then press CTRL+E.

Only certain steps have parameters that can be edited. If a selected step cannot be edited, the Edit Command button is disabled.

Motion Calculator

The Motion Calculator is the default editor for motion blocks. With it, you can create simple incremental motion blocks based on mathematical calculations, viewing the results of changes as you make them.

You can open the Add Motion wizard by right-clicking the Motions folder in the Navigator and clicking **Wizard**. Unless you configure a motion block to be edited as text, the Motion Calculator also opens when you right-click a motion block and click **Open**.

The Motion Calculator looks something like this:

Click one of these to select the kind of motion you want.

Click these buttons to zoom in and out of the graph of the plotted motion.

Click the arrows to pan around the graph of the plotted motion.

Click here to print the graph of the plotted motion on the default printer.

You can enter values for two of these three parameters. (The third is calculated.)

Enter new values for Acceleration time and Deceleration Time to override the defaults.

Click here to clear all values for the current motion.

Calculated values appear in the Calc column.

Enter your own values in the Data column.

Click here to fill in the Calc column and view (or update) a velocity/time graph of the calculated motion.

Motion Calculator Interface:

Motion Type: Triangular Trapezoidal

Profile: Velocity vs. Time graph showing a trapezoidal profile. The y-axis (Velocity) ranges from 0 to 7, and the x-axis (Time) ranges from 0 to 12. The profile starts at (0,0), rises linearly to (4,6), stays constant at 6 until time 8, and then falls linearly to (12,0).

Zoom: In, Out

Panning: [Four directional arrows]

Print

Trapezoidal Data

	Data	Calc	Units
Velocity	6	6	Units/sec
Total Time	12	12	sec
Distance	0	48	Units
Acc. Time	0	4	sec
Dec. Time	0	4	sec

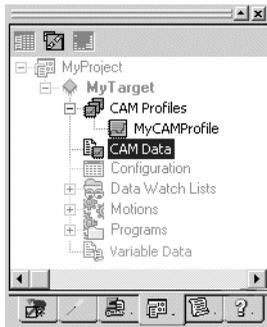
Calculate Clear

Typically, you would enter desired values in the Data column and click **Calculate**. The Motion Calculator updates the appropriate values in the Calc column and draws a velocity/time graph of the calculated motion. Units in this graph are based on those in the Units column.

When entering values into the calculator, keep in mind the following:

- You can fill in two (and only two) of Velocity, Total Time, and Distance. That is, one of these parameters must be left at 0 and the other two filled in with non-zero values. The Motion Calculator automatically calculates the missing parameter in the Calc column.
- If you leave Acceleration time (Acc. Time) or Deceleration time (Dec. Time) as 0, they are set to one-third the total time for the motion. You can override this by filling in the appropriate parameter.

When the motion is configured the way you want, click **Add Motion** (above the calculator in the wizard page) to add the motion block to your application.



Navigator: Project tab
CAM Profile nodes

CAM PROFILE EDITOR

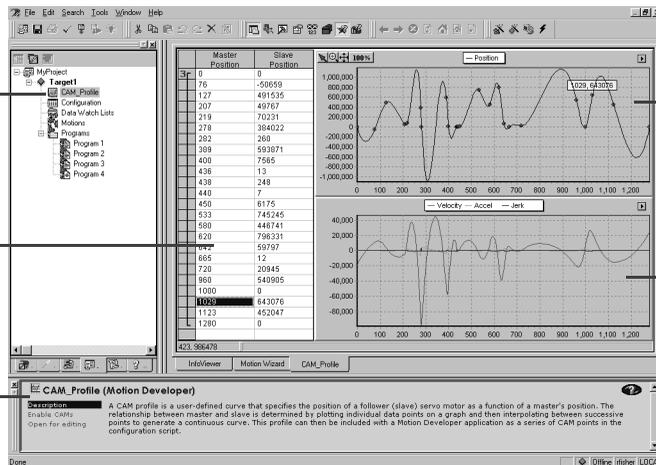
The Motion Developer CAM editor lets you create and edit electronic CAM profiles for a motion Controller target. A CAM profile is a curve that specifies the response of a slave servo to a master position index. The CAM profile is written to the target's configuration script. You can create as many profiles as you want, but only one can be downloaded to a target at a time. The profile that is downloaded is specified in the target's "CAM Profile" property.

You can edit a target's CAM Profiles by double-clicking its Profile node under the CAM Profiles folder. The CAM editor display appears as follows:

Double-click a CAM Profile node in the Navigator to open the CAM editor.

The CAM profile table supports numeric editing and curve order grouping

The Companion automatically provides information on what you click.



Curves of the profile can be adjusted and edited graphically

Secondary curves show the velocity, acceleration, and jerk of the CAM profile.

You can adjust the curves of your CAM profile to suit the specific needs of your project. With the CAM editor you create profiles by defining points on a master/slave position curve. Groups of adjoining points are allocated to sectors. Each sector is assigned a polynomial curve fit order (1,2,3) that specifies how the curve will be interpolated between points.

To include CAM Profile data when downloading your project, ensure that the target's Include CAM Points property is set to **True**. When downloading a project to a target with a defined CAM Profile, Motion Developer automatically converts the curve to a series of `CAMx=value` commands. These commands are added to the target's CAM Data node.

The number of CAMx commands added to the configuration script depends on the target's **CAM Point Resolution** property.

- If CAM point Resolution is 1 degree, 360 CAM commands are added to the script.
- If CAM point Resolution is 0.1 degrees, 3600 CAM commands are added to the script.

Downloading 3600 CAM commands can be very time-consuming. It is therefore recommended that you use a Resolution of 1 degree, at least during the initial development stages.

 **Want to know more?** In the Help Index, look up "Motion Developer CAM Editor: an Overview".

To add a new CAM Profile to a motion Controller target

1. In the  Project tab of the Navigator, right-click the  motion Controller target's  CAM Profiles folder and click **New Profile**.
A new Profile appears under the CAM Profiles folder.
2. Enter a new name for the Profile.
You can now open the profile in the CAM Editor. Note that only one CAM profile can be downloaded to a motion Controller at a time.

To edit a CAM profile on a motion Controller target

- In the  Project tab of the Navigator, under the  motion Controller target's  CAM Profiles folder, right-click the  Profile you want to edit and click **Open Profile** (or just double-click the Profile).
The CAM editor opens with the target's CAM profile.
For more details on using the profile editor, look up "Editing a Motion CAM Profile" in the online help.

To specify the CAM Profile that gets downloaded to the motion Controller target

1. In the  Project tab of the Navigator, right-click the  motion Controller target for which you want to select a CAM Profile and click **Properties**.
The  Inspector displays the CAM profile's properties.
2. If necessary, in the Inspector, set the target's Include CAM point property to **True**.
This ensures that Machine Edition includes CAM profile data the next time you download the application to the motion Controller.
3. In the Inspector, select the desired CAM Profile in the target's CAM Profile property.

This property appears only if Include CAM point is set to True. You can select from all CAM Profiles configured for the target.

Index

A

- actions 45, 46
- Active scripting 77
- adding
 - alarm groups (Bit and Word) 81
 - CAM Profiles 113
 - Control I/O drivers 63
 - custom web documents 89
 - language columns 91
 - logging groups 83
 - logic component 40
 - Machine Edition projects 20
 - OPC server links 86
 - panels 72
 - PLC Access drivers 84
- alarm groups 80
- alarms 80
 - Alarm Display objects 81
 - alarm groups 80
 - Bit Alarms 80
 - setting alarm conditions 81
 - Variable Alarms 80
 - Word Alarms 80
- Allen-Bradley RIO 62
- animation
 - graphical objects 74
 - touch animation scripts 75
- Application Builder
 - about 108
 - conditional steps 109
 - inserting steps 110
 - See *al/sosteps*, flowchart
- application scripts 75, 79, 96
- applications, motion 101
- ASCII drivers 62
- authorization. See product authorization

B

- backing up projects 20
- Bit Alarms 80
- blocks, motion. See motion blocks
- blue arrows. See docking markers
- branches 46

- Build tab 26

C

- Calculator, Motion See Motion Calculator
- CAM data 103, 112
- CAM point resolution 113
- CAM Profiles
 - specifying for download 113
- CAM profiles
 - about 112
 - adding 113
 - editing 113
 - editor 112
- client
 - networking 93
 - OPC 86
- Companion 18, 46, 50, 53, 56
- component
 - logic 40
- conditional steps 109
- configuration
 - CAM data 103
 - motion script 102
 - variable data 102
- configuring
 - alarm conditions 80
 - Control I/O 64
 - Controller 66
 - panels 73
 - PLC Access I/O 83
 - properties 22
 - user preferences 21
- Control I/O 63
 - tool 63
 - working with 64
- Controller 65
 - configuring 66
 - going online 31
 - going online (IL) 51
 - going online (ST) 54
 - running 67
 - scan cycle 65
- Controller Functions wizard 107

controller/drive targets, motion 101
correcting, errors 35
creating

- actions 46
- alarm display objects 82
- alarm groups (Variable) 81
- application scripts 75
- FBDs 55
- IL blocks 49
- macros 42
- panel scripts 76
- projects 20, 32, 71
- SFCs 42
- ST blocks 52
- subroutines 46
- touch animation scripts 76
- variables 33

custom web documents 89
adding 89
editing 89

D

Data Watch 47
debugging scripts 94
developing. *See* creating
DeviceNet 62
devices
 adding to PLC Access driver 84, 85
 motion control 100
 PLC Access 83, 84
docking markers 19
downloading 31, 35, 66

- QuickPanel View/Control 36
- motion device targets 35
- QuickPanel View/Control targets 36
- specifying CAM profile for 113
- Windows PC targets 35

drawing graphical objects 73
drawing tools 73
Drive Functions wizard 107
drive targets, motion 101
drivers

- adding
 - Control I/O 63
 - PLC Access I/O 84
- ASCII 62
- Control I/O 62, 63

Control Peer 62
OPC Client 86
PLC Access I/O 83

E

editing
 application scripts 75
 CAM profiles 113
 custom web documents 89
 FBDs

- offline 56
- online 58

- IL blocks
- offline 50, 51
- ladder programs
- offline 46
- online 47
- motion scripts 105
- panel scripts 76
- panels 73
- quick editing
- ladder program 47
- SFCs 43
- quick test 37
- SFCs
- offline 42
- online 44
- steps, flowchart 110
- structured text
- offline 53
- online 54

editors

- function block diagram (FBD) 55
- grid (QuickPanel) 79
- instruction list (IL) 49, 52
- ladder 45
- panel 72
- script 75, 96
- structured text (ST) 52

errors and warnings
 correcting 35
 Feedback Zone 26

external keypad 95

F

FBD Editor
 online 58
FBD editor

- offline 56
- FBD instructions 56
 - inserting from the FBD editor 57
 - inserting from the Toolchest 57
- FBD text boxes
 - inserting from the FBD editor 57
 - inserting from the Toolchest 56
- FBDs
 - creating 55
 - editing 56, 58
 - opening 55
- Feedback Zone 26
- fixing, project errors 35
- flowchart editor *See* Application Builder
- flowchart steps *See* steps, flowchart
- full-text search 17
- Function Block Diagram (FBD) editor 55

G

- GE Genius 62
- getting help 17
- global function library scripts 75
- graphical objects 72
 - animating 74
 - drawing 73
- graphical panels. *See* Panels
- grid editor
 - keypad assignment 95
 - messages 74

H

- hardware
 - motion devices 100
- Hardware Key Authorization 5
- help 17
- Historian Collector 82

I

- I/O
 - configuring
 - Control 64
 - PLC Access 84
 - Control 63
 - OPC 85
- IEC addresses 64
- IL blocks
 - creating 49

- editing 50, 51
- opening 49
- IL Editor
 - offline 50
- IL editor
 - offline 51
- Import tab 26
- index, search 17
- Indexed Image Display (IID) 74
- Indexed Message Display (IMD) 74
- InfoView tab 20
- InfoViewer 18
- inserting
 - FBD instructions 56
 - FBD instructions (FBD editor) 57
 - FBD instructions (Toolchest) 57
 - FBD text boxes (FBD editor) 57
 - FBD text boxes (Toolchest) 56
 - IL instructions 50
 - ladder instructions 46
 - motion script items 105
 - rungs 46
 - script functions 78
 - SFC instructions 43
 - ST instructions 53
- Inspector 22
- Inspector. *See* Inspector
- installing Machine Edition 4
- Instruction List (IL) Editor 49
- Instruction List (IL) editor 52
- Internationalization. *See* Languages folder

K

- keypad assignment 95
- keyword search 17

L

- labels 45
- Ladder Editor 45
 - offline 46
 - online 47
- ladder instructions 46
- ladder programs
 - defined 45
 - editing
 - offline 46
 - online 47
 - opening 45

- quick editing 47
- Language translation 90
- Languages folder 90
 - adding new languages 91
 - grid 91
 - moving language columns 92
 - renaming languages 92
- Languages grid 91
 - default language 92
 - delete language 93
 - display font 92
 - updating Source language 92
- locating
 - labels 45
 - variables 42, 46, 50, 53, 56
- logging data 82
 - enabling logging for a variable 82
- logging groups 82
 - adding 83
- logging strategies 82
- Logic Developer
 - PC 39 to ??
- Logic Developer - PC 39 to ??
- logic, adding 40

M

- Machine Edition
 - downloading 35
 - environment 16
 - getting help 17
 - key features 17 to 27
 - OPC servers 86
 - projects 20, 29
 - properties 22
 - toolbars 19
 - validating 35
- macros 42
- Main Motion Wizard 107
- Manager tab 20
- message grids 74
- messages 74
 - Indexed Image Display (IID) 74
 - Indexed Message Display (IMD) 74
 - Triggered Message Display (TMD) 74
- Messages tab 26
- Modicon Quantum 800 62
- Motion applications
 - configuration script 102

- programs 103
- motion applications 101
- motion blocks
 - about 104
 - editing with calculator 110
 - opening 101
 - switching editors 102
- Motion Calculator
 - about 110
 - using 111
- motion configuration 102
 - opening 101
- Motion Developer
 - applications 101
 - supported hardware 100
- Motion programs
 - fault-handing program 103
 - opening 101
 - switching editors 102
- motion programs 103
- motion scripts 105
- motion targets 101
- Motion Wizards
 - about 106
 - Application Builder 108
 - Controller Functions 107
 - Drive Functions 107
 - Main Wizard 107
 - navigating through 106
 - parameters 107
 - See *also* Motion Calculator
 - special wizards 108
- moving product authorization 8
- multi-language support. See Languages folder

N

- Navigator 20
 - property column view. See Property column view
- networking 93

O

- objects. See graphical objects
- offline
 - FBD editor 56
 - IL Editor 50
 - IL editor 51
 - Ladder Editor 46
 - SFC Editor 42

- ST editor 53
- online
 - FBD Editor 58
 - Ladder Editor 47
 - SFC Editor 44
 - ST editor 54
 - to the Controller 31
 - to the Controller (IL) 51
 - to the Controller (ST) 54
- online help 17
- OPC 85
 - client 86
 - server links 86
 - servers 86
- opening
 - application scripts 75
 - FBDs 55
 - IL blocks 49
 - ladder programs 45
 - macros 42
 - motion blocks, motion programs, and motion configuration 101
 - panel scripts 76
 - panels 72
 - projects 21, 30, 32, 40, 71, 99
 - SFCs 42
 - ST blocks 52
- Options tab 21
- options, user preferences 21

P

- Panel editor 72
 - animation 74
 - graphical objects 72
 - working with 73
- panel scripts 75, 79, 96
- panels 72
 - adding 72
 - configuring 73
 - editing 73
 - opening 72
 - quick testing 37
- parameters, Motion Wizards 107
- peer driver, Control 62
- phrase search 17
- PLC Access I/O 83
- preview panels. *See* quick test
- Product Authorization 5

- product authorization
 - authorizing Machine Edition 7
 - moving to another PC 8
- Proficy Historian 2.0 82
- profiles, CAM *See* steps, flowchart
- programs, motion *See* motion programs
- Project tab 21
- Projects
 - Shared 29
- projects 20, 29
 - adding 20
 - backing up 20
 - creating 20, 32
 - downloading 35
 - Motion Developer 99
 - opening 21, 30, 32, 40, 71, 99
 - running 30, 93
 - sharing between Machine Edition workstations 29
 - validating 35
 - with HMI component 71
 - with logic component 40
- properties 22
 - configuring 27
- Property column view 27
- proxy variables 84

Q

- QPScript language 96
- quick editing
 - ladder program 47
 - SFCs 43
- quick start 14
- quick test, panels 37
- QuickPanel
 - scripts 96
- QuickPanel Applications 95
- QuickPanel Grid editor 79
- QuickPanel View/Control
 - downloading 36
- QuickPanel View/Control targets 32
 - downloading 36

R

- References tab 26
- remote views 89
- Reports
 - variable 28
- Reports tab 26

- requirements, hardware and software 2
- resolution, CAM point 113
- rungs 45
 - inserting 46
 - write changes 48
- running
 - Controller 67
 - projects 30, 93
 - sample applications 30
 - View Runtime 93

S

- sample applications 30
- scan cycle, Controller 65
- script commands. *See* script functions
- Script editor 75, 96
 - inserting script functions 78
 - scripts 75, 96
 - working with 78
- script functions 78
- scripting languages
 - Active scripting 77
 - QPScript 96
 - VBScript 76
 - ViewScript 76
- scripts 75, 96
 - application scripts 75, 96
 - debugging 94
 - global function library scripts 75
 - inserting script functions 78
 - motion 105
 - panel scripts 75, 96
 - touch animation scripts 75
- scripts, motion
 - inserting items 105
- search, full-text 17
- search, index 17
- Sequential Function Charts. *See* SFCs
- servers
 - networking 93
 - OPC 86
- setting. *See* configuring
- SFC Editor
 - offline 42
 - online 44
- SFC instructions, inserting 43
- SFCs
 - editing 42
 - opening 42

- quick editing 43
- Shared Projects 29
- shortcut keys, View Runtime 94
- shunts 46
- Site Index 88
- Smart Lists 24
- Software Key 6
- ST blocks
 - creating 52
 - opening 52
- ST editor
 - offline 53
 - online 54
- ST instructions
 - inserting 53
- steps, flowchart
 - conditional 109
 - editing 110
 - inserting 110
 - step types 109
- structured text
 - editing 53, 54
- Structured Text (ST) editor 52
- subroutines 46
- switching editors, motion blocks and motion programs 102
- system requirements 2

T

- tags. *See* variables
- targets 29, 32
 - motion 101
- testing
 - panels 37
 - scripts 94
- toolbars
 - drawing tools 73
 - ladder instruction tools 46
 - Machine Edition tools 19
- Toolchest 25, 46, 56
- tools
 - Companion 18
 - Control I/O 63
 - Data Watch 47
 - docking markers 19
 - Feedback Zone 26
 - InfoViewer 18
 - Inspector 22
 - Navigator 20
 - overview 16

- toolbars 19
- Toolchest 25, 46, 56
- touch animation scripts 75
- translations. *See* Languages folder
- Triggered Message Display (TMD) 74

U

- Unused Variable report 28
- user preferences 21

V

- validating 26, 35, 66
- Variable
 - property column view 27
- Variable Alarms 80
- variable data 102
- Variables 27
 - managing 27
 - reports 28
- variables
 - creating 33
 - locating 42, 46, 50, 53, 56
 - proxy 84
 - watching values 58
- Variables tab 21, 27
- VBScript language 77
- View 71 to 97

- testing the HMI 37
- View Runtime 93
 - running 94
 - shortcut keys 94
- ViewScript language 76, 77

W

- Warm Standby 67
 - and OPC servers 69
 - configuring 68, 69
 - configuring OPC server for 69
- warnings 35
 - Feedback Zone 26
- watching variable values 58
- Web Access 87
 - Site Index 88
- Windows PC targets 29, 32
 - downloading 35
- windows. *See* Panels
- wizard, Controller Functions 107
- wizard, Drive Functions 107
- Wizard, Main 107
- wizards, motion *See* Motion Wizards
- Word Alarms 80
- word search 17
- writing motion scripts 105
- writing scripts 78

