

GFK-1868
New In Stock!
~~GE Fanuc Manuals~~

<http://www.pdfsupply.com/automation/ge-fanuc-manuals/programming-software/GFK-1868>

programming-software
1-919-535-3180

Proficy CIMPLICITY Machine Edition Getting Started

www.pdfsupply.com

Email: sales@pdfsupply.com

Getting Started

CIMPLICITY® Machine Edition

Version 3.00
August 2002
GFK-1868D



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 Fanuc Automation North America, Inc.

Disclaimer of Warranties and Liability

The information contained in this manual is believed to be accurate and reliable. However, GE Fanuc Automation North America, Inc. assumes no responsibilities for any errors, omissions or inaccuracies whatsoever. Without limiting the forgoing, GE Fanuc Automation North America, 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 software described herein. The entire risk as to the quality and performance of such information and software is upon the buyer or user. GE Fanuc Automation North America, Inc. shall not be liable for any damages, including special or consequential damages, arising out of the use of such information or software, even if GE Fanuc Automation North America, Inc. has been advised in advance of the possibility of such damages. The use of the information contained in the manual and software described herein is subject to GE Fanuc Automation North America, Inc. standard license agreement, which must be executed by the buyer or user before the use of such information or software.

Notice

GE Fanuc Automation North America, Inc. reserves the right to make improvements to the product described in this publication at any time and without notice.

© 1999, 2000, 2001, 2002 GE Fanuc Automation North America, Inc. All rights reserved. CIMPPLICITY is a registered trademark of GE Fanuc Automation. Any other trademarks referenced herein are used solely for purposes of identifying compatibility with the products of GE Fanuc Automation North America, Inc.

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

doc@gefanuc.com

Contents

1	Welcome	1
	System Requirements	.2
	Installation	.3
	Product Authorization	.4
	Technical Support	.6
	North America	.6
	South America	.6
	Europe	.6
2	Machine Edition	7
	Quick Start	.8
	The Machine Edition Environment	.10
	Getting to Know Machine Edition	.11
	Right-click, right-click, right-click	.11
	Getting Help	.11
	Accessing the Right Tool	.13
	Projects and the Navigator	.14
	Properties and the Inspector	.16
	Data Watch Lists	.17
	Smart Lists	.18
	The Toolchest	.19
	The Feedback Zone	.20
	Managing Variables	.21
	Machine Edition Projects	.23
	Running a Sample Project (View/Logic Developer - PC)	.23
	Developing a Machine Edition Project	.25
	Validating and Downloading a Project	.28
	Testing a View Project	.31
3	Logic Developer - PC	33
	SFC Editor	.35

Sequential Function Chart	35
Working with the SFC Editor - Offline	36
Working with the SFC Editor - Online	38
Ladder Editor	39
Ladder Program	39
Working with the Ladder Editor - Offline	40
Working with the Ladder Editor - Online	41
Instruction List Editor	43
Instruction List	43
Working with the IL Editor - Offline	44
Working with the IL Editor - Online	45
Structured Text Editor	46
Structured Text	46
Working with the ST Editor - Offline	47
Working with the ST Editor - Online	48
Control I/O Drivers	49
I/O Drivers	49
Control I/O Tool	50
Working with Control I/O	51
Controller	52
Working with the Controller	53
Warm Standby	55
Working with Warm Standby	55
4 View	57
Panel Editor	58
Panels	58
Working with the Panel Editor	59
Script Editor	60
Scripts	60
Scripting Languages	61
Working with the Script Editor	63
Logging	65

Alarms	.66
Alarm Objects	.66
Alarm Groups	.66
PLC Access I/O	.67
Drivers	.67
OPC	.68
OPC Client	.68
Machine Edition OPC Servers	.69
Web Access	.70
View Runtime	.72
Networking	.72
Languages folder	.73
Working with the Languages editor	.74
QuickPanel Applications	.76
QuickPanel Panel Editor	.77
Panels on QuickPanel targets	.77
Grid Editors	.78
Alarms	.78
Message Displays	.79
External Keypad Assignment	.80
Scripts on QuickPanel Targets	.81
QPScript Language	.82

5 Motion Developer 83

About Motion Developer	.84
Supported motion control devices	.84
Motion Applications	.85
Motion targets	.85
Motion configuration	.86
Motion programs	.86
Motion blocks	.87
Motion Scripts	.88
Motion wizards	.89
Special Wizards	.91

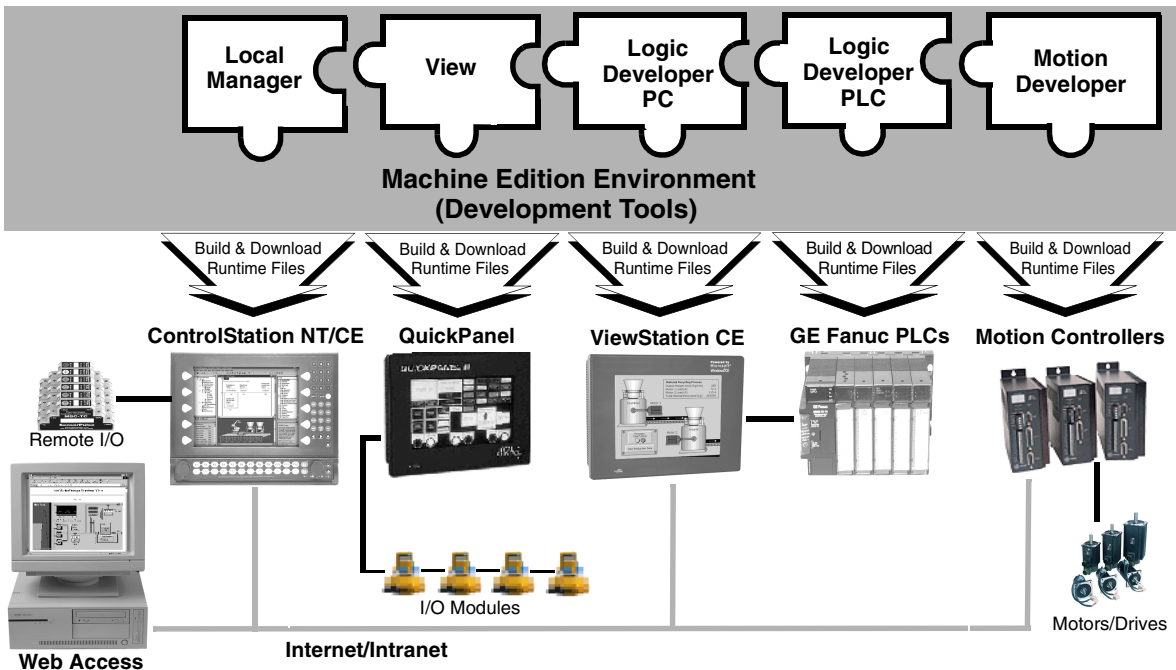
Application Builder flowchart editor	91
Motion Calculator	93
CAM Profile Editor	95
6 Local Manager	97
Local Manager	98
Version Control	99
Creating and Adding Projects to Local Manager	100
Access Control	104
The Access Control Database	104
Access Control of Projects	105
Audit Trails	107
The Audit Trail Database	107
Audit Trail Reports	108
Index	111

1

Welcome

Congratulations on your purchase of a CIMPPLICITY® Machine Edition product. This package provides all the tools necessary to create powerful control and HMI applications for a variety of run-time targets; a truly scalable solution. Machine Edition provides the following features:

- A fully-integrated environment. Every tool and editor works with the others.
- Logic Developer - PC, a scalable, computer based control software that includes a complete set of IEC standard editors (available with Control Station products).
- Logic Developer - PLC, a software tool for programming the full line of GE Fanuc PLCs (See GFK-1918 Getting Started Logic Developer - PLC).
- View, a complete system for creating HMIs (human machine interfaces) for Windows NT, Windows CE, and QuickPanel units.
- Motion Developer, for developing Whedco motion control applications.
- Web Access features that access real-time HMI data any time, from anywhere.
- Local Manager, a version control, security and audit trail system for your automation and control development process.



SYSTEM REQUIREMENTS

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

Development Environment

- Windows® NT version 4.0 with service pack 6 or later OR
Windows 2000 Professional OR
Windows XP OR
Windows 98 SE.
- Internet Explorer™ version 5.5 Service Pack 1 or later.
- 200 MHz Pentium-based workstation (300 MHz on Windows 2000 or XP).
- 128 MB RAM minimum.
- 110-310 MB hard disk space, depending on the selected products.
- 200 MB hard disk space for sample projects (optional).
- Additional hard disk space for projects and temporary files.

Windows® NT Runtime

- Windows® NT version 4.0 with service pack 4 or later OR
Windows 2000 Professional OR
Windows XP.
- 200 MHz Pentium-based workstation (300 MHz on Windows 2000 or XP).
- 64 MB RAM.
- 200 MB free hard disk space.

HRT Runtime

- Windows® NT version 4.0 with service pack 4 or later.
- VentureCom RTX version 4.3 or 5.0.
- 200 MHz Pentium-based workstation .
- 64 MB RAM .
- 200 MB free hard disk space.

Windows® CE Runtime

- For HMI and Logic components: GE Fanuc ControlStation CE, ControlStation CE II, or ControlStation CE IIx.
- For HMI only: GE Fanuc ViewStation CE, ViewStation CE II, or ViewStation CE IIx.

INSTALLATION

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

- When installing Machine Edition, select **IPI for Machine Edition** on the initial Launcher screen.
- 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, then click the What's New link under Get Started on the left hand side.

To install Machine Edition from a CD

1. **Insert the CIMPLICITY Machine Edition CD into your CD-ROM drive.**
Windows will automatically start the setup program. If the setup program does not automatically start, run *Setup.exe* in the root directory of the CD.
2. **Click Install to start the install process.**
3. **Follow the instructions as they appear on the screen.**

PRODUCT AUTHORIZATION

Before you can start developing projects in Machine Edition, you must authorize the software with a program called Product Authorization. If you don't authorize the software, you will only be able to use it for a four-day trial period. This procedure will take only a few moments and will allow you to take advantage of any product support for which you qualify. You will need to contact us by telephone, fax, or e-mail as part of the authorization process.

To authorize a copy of Machine Edition

1. **Have your serial number(s) ready. The serial numbers can be found on the License Key sheet that came with your product.**
2. **Run the Product Authorization program from the Start menu/Programs/CIMPLICITY Machine Edition/Product Authorization.**
The Product Authorization dialog box appears.
3. **Click Software, and then click Add.**
4. **Select the medium with which you are authorizing: Internet, Phone/Fax/E-mail or Floppy Disk Transfer. Click Next.**

If you choose the Internet option, proceed to step 5.

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

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

5. **Fill in the fields in the dialog box. Fields that are 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.** From 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 once 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. For example, if you bought ControlStation NT, you will need to authorize both Runtime *and* Development.

To move the authorization to another computer

You can run the software only on the computer that the Product Authorization was run on. If you want to develop your projects on a different computer, you will need to complete the following steps to move the authorization from one computer to another.

1. **Install Machine Edition on the computer that the authorization will be moved to. Run the Product Authorization program from the Start menu/Programs/CIMPLICITY Machine Edition/Product Authorization.**
The Product Authorization dialog box appears.
2. **Click Move and then click OK.**
There is a Target Site Code on the top right hand side of the screen. Write down this site code carefully. It must be accurate for the move to work. You will need the Target Site Code when you move the authorized software from the source computer.
3. **Click Authorize by disk.**
At this point, you need to go to the source computer that has the authorized software, and move the authorization to a disk.
4. **From the source computer, run the Product Authorization program.**
5. **Click Move, and then click OK. Enter the Target Site Code that you wrote down from Step 3 and click Next. Verify that the site code is correct and click OK.**
6. **Insert a blank formatted floppy disk into the floppy drive and click Next. The authorization code will be moved to the disk and a dialog box should appear telling you it was successful. Click OK.**
7. **Go back to the computer to which you are moving the authorization and insert the floppy disk. (The screen that is asking for an authorization disk should be displayed.) Click Next.**
8. **Click Finish. A screen should appear telling you the move was successful. Click OK.**
The authorization has now been moved to the new computer.

TECHNICAL SUPPORT

Support is available to registered users at no charge for 90 days after purchase. A *Support and Free Enhancements (SaFE)* subscription can be purchased from your local GE Fanuc distributor if extended support is required.

If problems arise that can't be solved using the information in your product manual, online Help system, or the GE Fanuc Technical Advisor knowledge base, contact us by telephone, fax, or mail.

When contacting us, call from a telephone near your computer and have your CIMPLICITY Machine Edition software running. Have the following information handy to help us assist you as quickly as possible:

- The serial number from your installation CD case, and the Product name and version number from the Help>About dialog box.
- The brand and model of any hardware in your system.
- Operating system and version number.
- The steps you performed prior to the problem occurring.

North America

Support Hotline: 1-800-GEFANUC (1-800-433-2682) **Fax:** (780) 420-2049
Internet: <http://www.gefanuc.com> **Email:** support@gefanuc.com
Comments about our manuals and help: doc@gefanuc.com

Mailing Address: GE Fanuc
2700 Oxford Tower, 10235 - 101 St.
Edmonton, AB, Canada, T5J 3G1

South America

Telephone: +58 (261) 760 2862 **Fax:** +58 (261) 765 0909
Internet: <http://www.gefanuc.com> (visit our Portuguese web site at www.gefanuc.com.br)
E-Mail: luis.wilhelm@gefanuc.com

Mailing Address: GE Fanuc Automation Latin America
Calle 120 con Av. 17, Los Haticos -GE Turbimeca
Maracaibo, Venezuela

Europe

Internet: For up-to-date contact information, visit www.gefanuc-europe.com and click on "Offices and Services".
For up-to-date technical information visit www.gefanuc.com/support.
E-mail: plc_europe@gefanuceur.ge.com

2


Machine Edition

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'll 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 have the choice of what type of machine your projects are downloaded to and 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've finished, you'll have a solid foundation for building your own automation projects.

- For more help getting started, see the **Tutorials** in the online help (look in the  InfoView tab of the Navigator).

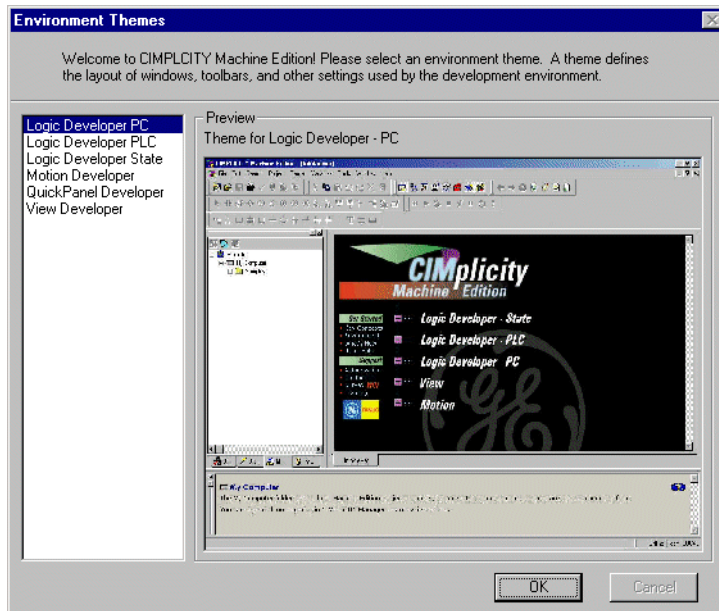
QUICK START

Machine Edition makes it easy to get started developing a project.

To start Machine Edition

1. Click  Start, point to Programs, then CIMPLICITY Machine Edition and then click CIMPLICITY Machine Edition. (See also page 23.)

After Machine Edition initializes, the Environment Themes dialog box appears.

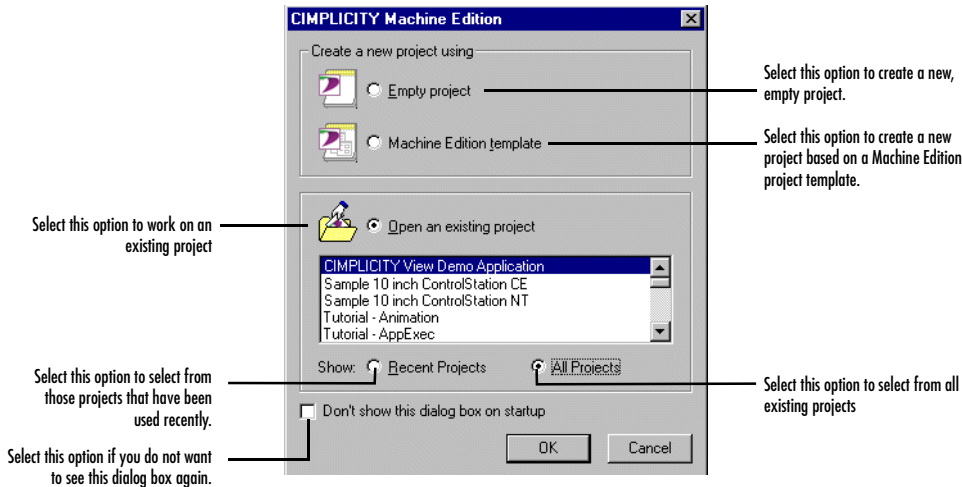


Note: The Environment Themes dialog box appears only the first time you start Machine Edition.

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 will match the preview in the Environment Themes dialog box. As you work you will probably modify the environment, open and close windows, change the selected toolbars and so on. Any changes you make will be retained in your default environment setup.

The CIMPLICITY 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 the Empty project option or the Machine Edition template option, the New Project dialog box appears to let you create a new project.
 - If you select the Open an existing project option, you can also choose from Recent Projects or All Projects. Recent Projects is selected by default.
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 open and use to familiarize yourself with Machine Edition.
 6. If desired, select the **Don't 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. See also, Machine Edition Projects page 23.

THE MACHINE EDITION ENVIRONMENT

All Machine Edition tools and editors appear in the Machine Edition window. The Run-time programs are separate Windows applications, so that 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'll 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'll look at some of the tools in the following section; for more details on the others, see the online Help.

The screenshot displays the SIMPLICITY Machine Edition interface. The main window shows a ladder logic program with rungs and components like Hopper1.Photoeye, Hopper1.Active, Hopper1.FRMTimer, and Hopper1.Valve. Surrounding the main window are several tool windows: a Navigator on the left showing a tree structure of project files; a Guided Tour on the right listing objects like Box, Conveyor, Hopper, and Robot; a Ladder Program window with a 'How do I...' and 'Related Topics' section; a Variable Watch window at the bottom showing a table of variables; and a Feedback Zone at the bottom right displaying validation messages.

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're doing.

Editor window: one of many windows where you create and edit your application.

Editor window tabs: lets you switch between open editors with the click of a mouse.

Common Machine Edition Tools

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

Toolchest: contains preconfigured objects that can be dragged 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.

Variable Name	Address	Value
Robot1	Done	Done
Robot1	Active	On
Robot1	Begin	G2
Robot1	Photoeye	I5
Robot1	Height	A15

GETTING TO KNOW MACHINE EDITION



The following are some key features of the Machine Edition environment. Learning about 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'll get things done. Machine Edition tailors the list 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.
- Search for keywords using the index: on the Help menu, select 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 GE Fanuc on the Web, and then choose:

- Technical Advisor.
- GE Fanuc Home Page.

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

The screenshot shows the Machine Edition software interface. The Help menu is open, showing options like 'Finding Information' and 'Index...'. The 'Runtime Tools' window is visible, listing various tools like 'Variable Inspector', 'Script Debugger', 'Driver Communication', 'Panel Selector', 'Logon', 'Edit Users', and 'About Box'. The 'Buttons' window is also open, displaying help text for the 'Button' tool, including a list of steps for changing the size of the button object.

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

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

Use the InfoViewer toolbar to navigate the HTML help.

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

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


Click this button to open the InfoViewer with more detailed information on the topic in the Companion.

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

Getting Help

Companion

The Companion window displays brief information about whatever item you happen to be 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's been turned off, click  on the Tools toolbar to open it again.

InfoViewer

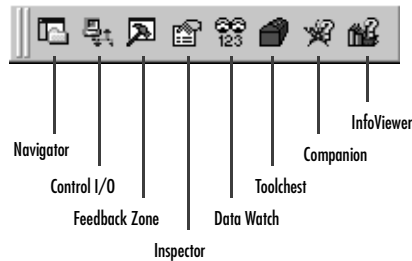
The InfoViewer window is an embedded HTML browser that provides more detailed and procedural help. It is launched whenever you access help topics from the InfoView Tab, the help index, or by pressing F1 on a selected item.

? 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's name.
- On the Tools toolbar, click the tool's icon. To display the Tools toolbar, on the **Tools** menu, choose **Toolbars** and select the Tools check box.



Tools Toolbar

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".


Projects and the Navigator

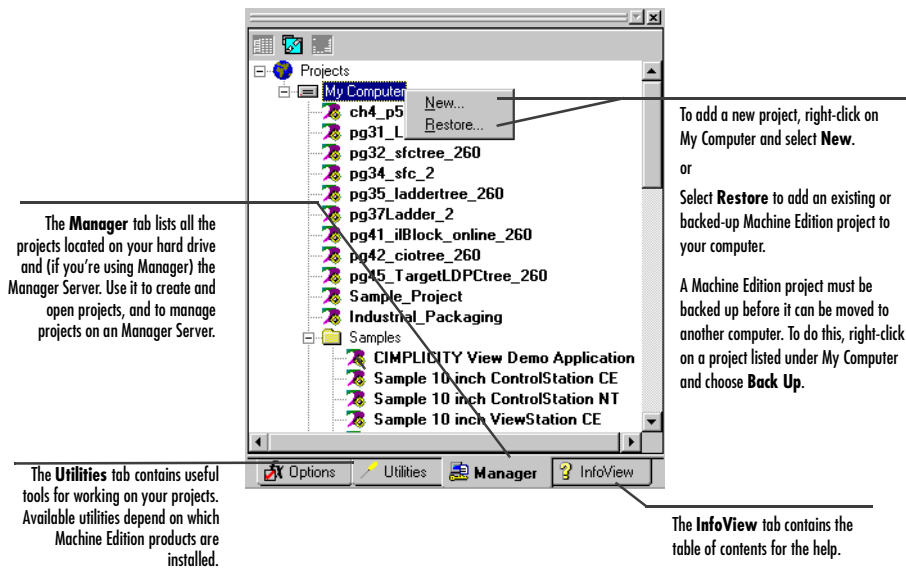
Navigator

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 only appear if 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 have access to 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”.

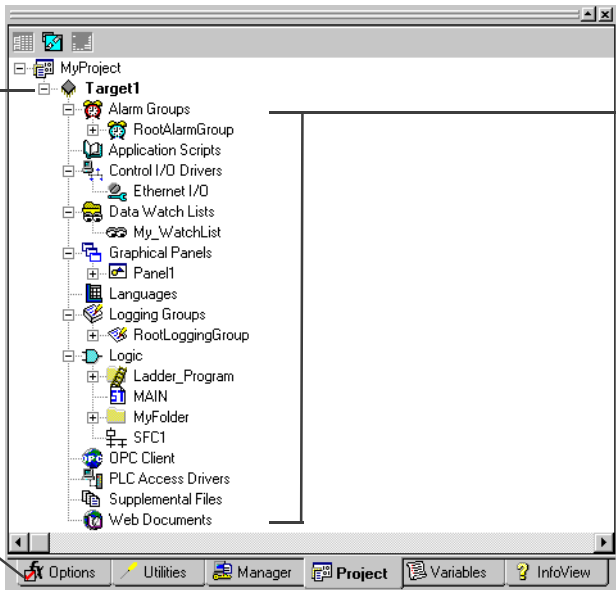


Navigator Window with Local Project List

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.



Targets are the 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 appeared when a Logic component was added to Target1.

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 simply 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 will show all those properties associated with the selected object.
- When a property changes an object's appearance, you'll 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.

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

Maximum length is [255]. All characters are valid

To edit this property, select the new value from the list box.

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

Properties are organized much like Windows folders.

Click + to expand the list of subproperties.

Click - to collapse the list of subproperties.

Select a tab for more properties.

If you're 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.

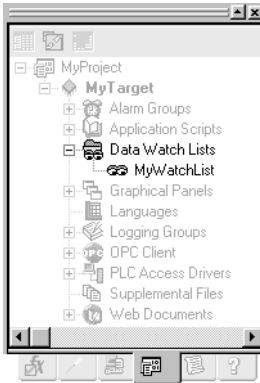
Variable [Target1]	
Name	myDINT
Description	Filler status
Publish	True
Mark As Used	False
Array Dimension 1	0
Data Source	PLC Access
Device	<Unassigned>
I/O Address	A13
Data Type	DINT
Current Value	0
Initial Value	0
Default Display Format	Decimal
Retentive	False
Engineering Units	
Range Limits	

Inspector Window

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


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

Data Watch Lists



Navigator: Project tab

Data Watch nodes

The  Data Watch (only available while editing a project) is a debugging tool that lets you monitor and edit the real-time values of variables defined in your project. This is useful while working online with a runtime target. With the Data Watch tool you can monitor individual variables or user-defined 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 IEC or Reference address of variables mapped to I/O terminals or PLC memory.

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

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.

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

Double-click a value to change it.

	Address	Value
Robot1		Done: On, Active: Off, Begin: Off, photoeye: Off
Done		On
Active	T1	Off
Begin	Q2	Off
photoeye	I5	Off
Height	AI5	77

Data Watch Window

To define a Watch List:

1. Right-click the Data Watch Lists folder and select **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. In many cases, when text entry is required, a Smart List will appear to speed your progress.

Enter the name of a variable or expression.

fill

BoilTemp
BottleCount
Cap_Status
Fill_Level
Fluid_Level
Heat_Timer
Integer1
meter
Org_1

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.

Smart List Display

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 called "fill", simply press ENTER without using the Down arrow.

The Toolchest



Toolchest

The Toolchest (only available 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—and since 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 and choose “Toolchest”.

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.

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

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

The Feedback Zone



Feedback Zone

The Feedback Zone (only available while editing a project) is an interactive window that displays output information generated by Machine Edition-enabled components. Using 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 on 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 jumps you to the location of the selected item in the project.

```

Error 2861: 'Active' is invalid [Rectangle, Rectangle257: Fill Color Expr]
Error 2861: 'bexpressent' is invalid [Rectangle, Rectangle259: 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 (eg., errors/warnings will be listed when you import variables from a PLC into a Machine Edition project).

The **Messages** tab tracks and displays operations that have been completed within Machine Edition (eg., a message is added everytime you open a project).


The **References** tab displays a list of all the places a variable is used in a project. To do so, 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

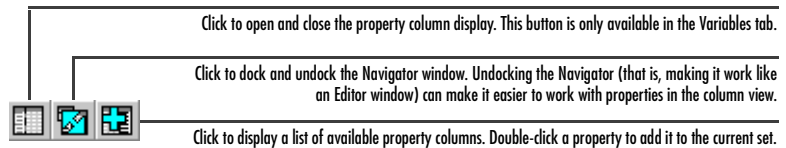
Variables are named storage spaces for data values defined in Machine Edition projects. A variable could store the current velocity of a motion controller motor, the height of a robotic arm, or any other value that an application needs to keep track of. Most variables in a project can be shared among 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 called the *Variable List*. Like most items in Machine Edition, you configure variables by editing their properties in the Inspector.

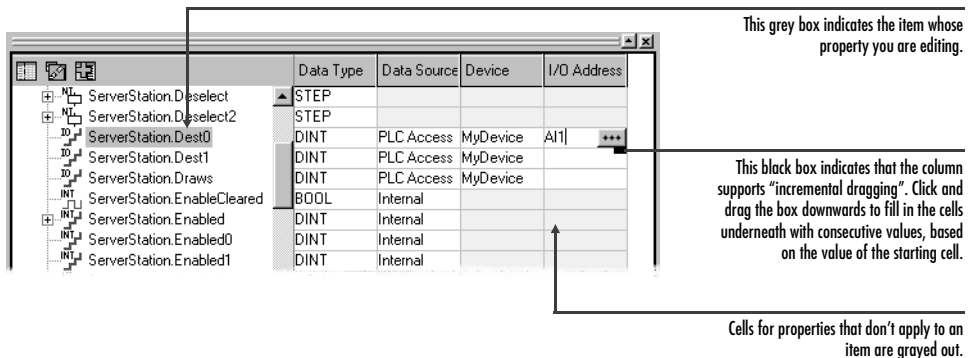
The values a variable can store depends on its *data type*. For example, a DINT data type indicates that the variable can store “Double Integers”, 32-bit values. The location where a variable’s value is stored is indicated by its *data source*. Typically, a variable’s value is either stored internally in the target’s memory or is retrieved (and sent) to external PLC 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.

Property Columns

In addition to the Inspector, the three buttons at the top of the Navigator let you 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 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 “Variables in View”.
 - For information on variables in QuickPanel projects, choose “Variable support in QuickPanel applications”.
 - For information on variables in Logic Developer - PC, choose “Variables in Logic Developer - PC”.
 - For information on variables in Logic Developer - PLC, choose “Variable support for GE Fanuc PLC targets”.
 - For information on variables in Motion Developer, choose “Variables 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 called a *project*. Each Machine Edition project is made up of *targets* and (sometimes) *components*. A target is the hardware platform where the finished project runs, such as a Windows NT computer, a ControlStation/ViewStation CE box, a motion controller, or a QuickPanel unit. Different Machine Edition products support different target types. Targets are often further subdivided into models, such as the ViewStation and ControlStation versions of CE targets. 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, View lets you add an HMI component to NT and CE targets, used for creating human-machine interfaces. Logic Developer - PC lets you add a Logic component to NT and CE targets. This lets you create ladder logic to make a PC act as a Controller. ViewStation CEs do not support logic; therefore you cannot download a project that contains logic to a ViewStation CE.

Motion Developer lets you program motion controllers from your PC. Each motion device programmed by Motion Developer is represented by a separate Motion target. Since motion targets have no additional capabilities, adding additional components is not necessary.



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

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

If you’re 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 CIMPLICITY Machine Edition from the Start menu/Programs/CIMPLICITY 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

- **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 NT target.
- **Car Wash:** an HMI and logic application that runs on a Windows NT target. This application is built in an SFC document.
- **ControlStation HMI Features:** an HMI and logic application that runs on a Windows NT target. This application also shows some web documents.
- **FC2000 Brewery:** an HMI and logic brewery application that runs on a ControlStation/ViewStation CE. (Projects containing Logic Developer - PC cannot be downloaded to a ViewStation, as ViewStation CEs do not support logic.)
- **FC2000 Zone Control:** an HMI and logic application that runs on a ControlStation/ViewStation CE. This application runs a four-zone ventilation logic system. It is to be used with the ControlStation/ViewStation CE Tutorial. (Projects containing Logic Developer - PC cannot be downloaded to a ViewStation, as ViewStation CEs do not support logic.)
- **ST - Lunar Lander:** an HMI and logic application that runs on a Windows NT target. This project demonstrates the Structured Text (ST) language and its interaction with UDFBs (User Defined Function Blocks).

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 Developer'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 select 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. For information about going online to the Controller, see "To go online to the Controller" on page 25.

To go online to the Controller

If you have chosen a project that contains ladder 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 don't 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 ladder 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. You can start and stop Runtime and the Controller by right-clicking the  target, pointing to Online Commands, and selecting Start/Stop Runtimes from the submenu. To go offline from the Controller, right-click the target and select Go Offline.

Developing a Machine Edition Project

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


For View and Logic Developer - PC, your targets can be a Windows NT computer (either the one you're developing the project on, or a remote one that you connect to through a network), or a Windows CE device (a ControlStation/ViewStation CE, CE II, or CE IIx).

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

For Logic Developer - PLC and Logic Developer - State, your target is a GE Fanuc PLC. You can also use remote I/O targets that represent a variety of remote I/O adapters.

If you're using Motion Developer, the choice is easy—simply determine whether your motion hardware is a motion controller/drive or a drive-only device (see page 85).

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 ViewStation does not support Logic components. A project can have multiple targets of different types with various components running on each 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 CIMPLICITY Machine Edition from the Start menu/Programs/CIMPLICITY Machine Edition.**

2. **Create a project using a template.**


In the  Manager tab of the Navigator, right-click  My Computer (if using Manager—right-click Machine Edition) and select New. The New Project dialog box appears.

3. **In the New Project dialog box, type a name for the new project, select a template, and 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 what type of target your project will download to as this will determine which template you choose.

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 Project with CIMPLICITY Manager".

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 NT,


CE, QuickPanel or PLC 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 and choose “Variables: an Overview”.



5. Create your application.

With Logic Developer - PC, add logic (Ladder, IL, ST, SFC) to your project and configure a control I/O driver to model your equipment. If the template you chose did not include a Logic component, add one now—right-click the target, select Add Component, and choose Logic. (ViewStation doesn’t 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 Ladder 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 (IL, ST) and organize your logic with SFC.



 **Want to know more?** In the Help Index, look up logic and choose “Ladder Logic Instructions: an Overview”.


With View, 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, select Add Component, and choose HMI. (If a target has both an HMI and a Logic component, you’ll 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 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 target and selecting **Main Wizard**.
- In the wizard page, click Motion Expert and follow the instructions on screen. For more information on using wizards, see page 89.

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 select 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 and choose “Control I/O Tool”.

 - For View, set up PLC Access I/O in the  PLC Access Drivers folder. To add a driver, right-click the PLC Access Drivers folder and select New Driver. Configure the drivers in the Inspector window. On NT targets, you might 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 Runtime 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 runtime files necessary for a target to perform its role in a completed project. The second 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 Configuration and select Communication Parameters. Click Finish when you're done.
- To download a View or Logic Developer - PC project to the Windows NT computer on which you are developing (a *local* target), ensure the target property Computer Address is set to “.” (without the quotation marks). Otherwise—for *remote* targets—type 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 the target property Computer Download Port is set to the applicable COM port.

Additional basic information on setting up Windows NT and CE targets follows.

Preparing a Remote NT target for Downloading

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

1. **If you are downloading to another NT computer (other than the ControlStation or CE box), ensure that the runtime files are installed on that device.**

ViewStation and ControlStation targets are shipped with the runtime files preinstalled. If you're using your own Windows NT computer, you'll have to install the runtime files yourself.

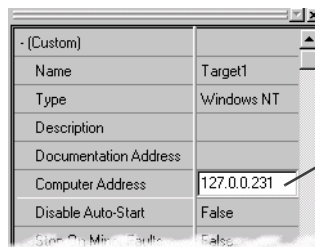
2. **Share the Machine Edition installation directory with the development computer that contains the project files.**

In the Windows NT Explorer, navigate to the Machine Edition folder, right-click on it and select Sharing. Click Shared As and in the Share Name field, type Machine Edition.

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

Start the Windows NT User Manager (from the Taskbar, click Start, point to Programs, Administrative Tools, and select User Manager). Double-click the Guest user in the top half of the display. Clear the Account Disabled check box.

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




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

? Want to know more? In the Help Index, look up downloading and choose "Preparing Windows NT Targets for Downloading".


Preparing a Windows CE Target for downloading

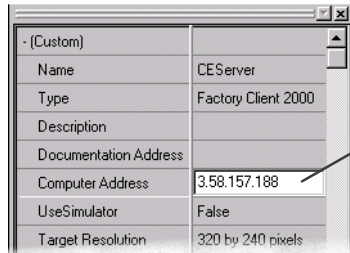
The following steps need to be completed on the ControlStation/ViewStation CE II, IIx before you can download a project to it. If you are not physically connected to a ControlStation/ViewStation CE but want to test your project, you can download the project to your local computer by setting the Use Simulator target property to True. Note that you cannot download a project that contains a logic component to a ViewStation CE, since ViewStation CEs do not support logic.

1. Physically connect the ControlStation/ViewStation CE to the Ethernet network. This step may require the help of a network administrator.
2. Within your Machine Edition project, specify the IP address of the ControlStation/ViewStation CE in the Computer Address target property.

To find the IP address of a ControlStation/ViewStation CE, double-tap the  System Information icon located on the desktop of the CE box.

By default, the ControlStation/ViewStation CE uses a DHCP (Dynamic Host Configuration Protocol) IP address. This means 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 ControlStation/ViewStation CE.

 **Want to know more?** In the Help Index, look up downloading and choose “Preparing Windows CE Targets for Downloading”.



Type the IP address of the ControlStation/ViewStation CE 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 in that it allows you to 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 ControlStation/ViewStation CE project, you need to set the target's **Use Simulator** property to True and perform at least one download after changing the setting.

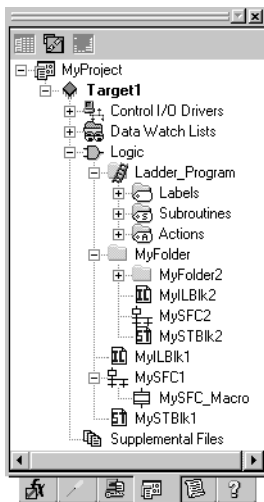
To Quick Test changes made to a graphical panel

1. Ensure that you've downloaded the HMI to the target computer at least once.
2. If you haven't already done so, open the graphical panel you want to test.
3. Right-click in the panel and select Quick Test.

View Runtime appears with a preview of the graphical panel.

3

Logic Developer - PC








Navigator: Project tab

Logic component nodes



Logic Developer - PC is the PC control section of Machine Edition providing the logic component of a project. 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 the PC Logic Controller and executed. 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 NT and CE targets.

All user-defined logic programs are contained in the  Logic folder, including the following:



-  Sequential Function Chart (SFC) Editor (see page 35)
-  Ladder Program (see page 39)
-  Instruction List Blocks (see page 43)
-  Structured Text Blocks (see page 46)

The following runtime elements are included with Logic Developer - PC:




-  Control I/O Drivers (see page 49)
-  PC Logic Controller (on target computer) (see page 52)

Note:  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 except ladder, of which there is only one, contained by the logic folder.




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 is opened in the Navigator.

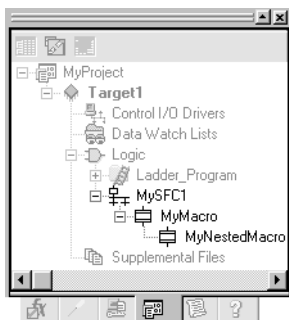
To add a logic component to an existing project

- In the  Project tab of the Navigator, right-click a  target, point to Add Component and choose 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 then choose 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



Navigator: Project tab
SFC nodes

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. 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, or ST 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 governs when and how the action will be 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 see the **SFC Tutorial** in the online help (look up SFC in the Help Index). This tutorial takes you, step-by-step, through the creation of a Sequential Function Chart that controls traffic signals.

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 then choose 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.

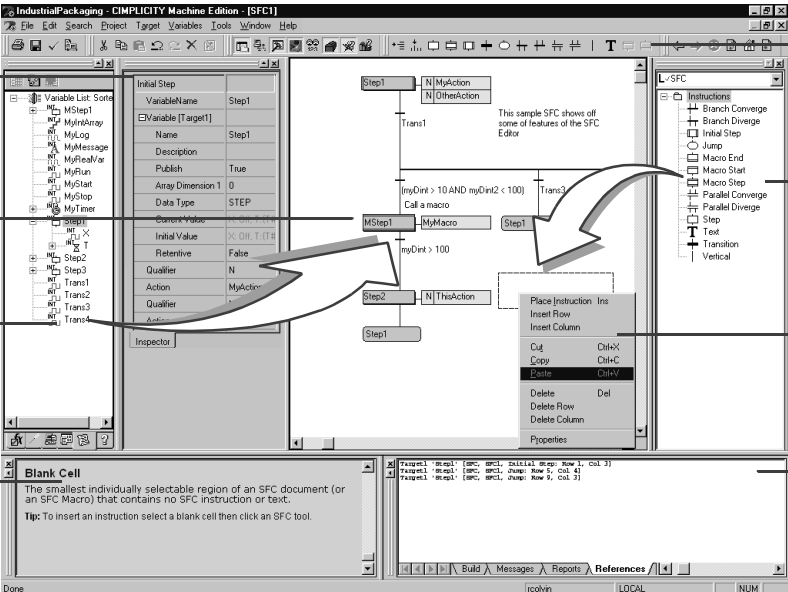
To create a new SFC macro

1. 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 choose New.
2. (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 SFC Editor interface with several callouts explaining operations:

- Configure instructions and their associated variables in the Inspector.** Points to the Inspector panel on the left, which shows the configuration for a selected instruction (Step1).
- Call a macro with a Macro Step instruction.** Points to a Macro Step instruction in the SFC diagram.
- Use the Variable List to create variables then drag them to instructions in the SFC Editor.** Points to the Variable List on the left and a variable being dragged to an instruction.
- Insert instructions, rows, and columns with the SFC toolbar.** Points to the SFC toolbar at the top of the editor.
- Drag instructions from the SFC drawer of the Toolchest to the SFC editor.** Points to the SFC drawer on the right side of the editor.
- Right-click a blank cell to start editing.** Points to a context menu that appears over a blank cell in the SFC diagram.
- Locate all instances of a variable (References) or syntax errors (Build).** Points to the Build and References buttons in the bottom toolbar.

Blank Cell
 The smallest individually selectable region of an SFC document (or an SFC Macro) that contains no SFC instruction or text.
Tip: To insert an instruction select a blank cell then click an SFC tool.

To insert an SFC instruction

1. In the SFC Editor, right-click a blank cell and choose 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 allow you to 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

The SFC editor allows you to view the execution of an SFC as it happens. The following picture illustrates this capability.

Active steps and transitions are highlighted in bold color.

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

The References tab of the Feedback Zone lists all instances of a variable. Click an entry to locate it in an editor.

Power flow in the associated ladder program (actions) is indicated in bold color while actions are executed.

Many actions can be viewed at once.

Use the Data Watch to monitor or change data values in real time.

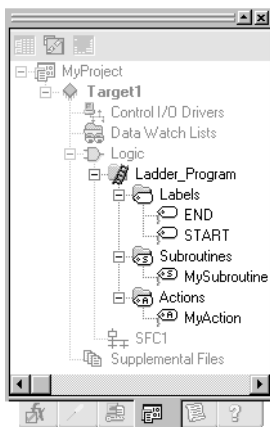
Variable	Value
Step1	X On, T: (T#0d 0h 4m 17.816s)
MyMacro	On
MyTB	T#0d 0h 4m 17.816s

To go online to the Controller's SFC

- In the Project tab of the Navigator, right-click the target and choose 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 is likely 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.





Note: You can also use User-Defined Function Blocks (UDFBs). Each UDFB is a callable subroutine that you can pass parameters to/from.

 **Want to know more?** In the Help Index, look up ladder and choose “Ladder Program: an Overview” or “UDFB”.






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.

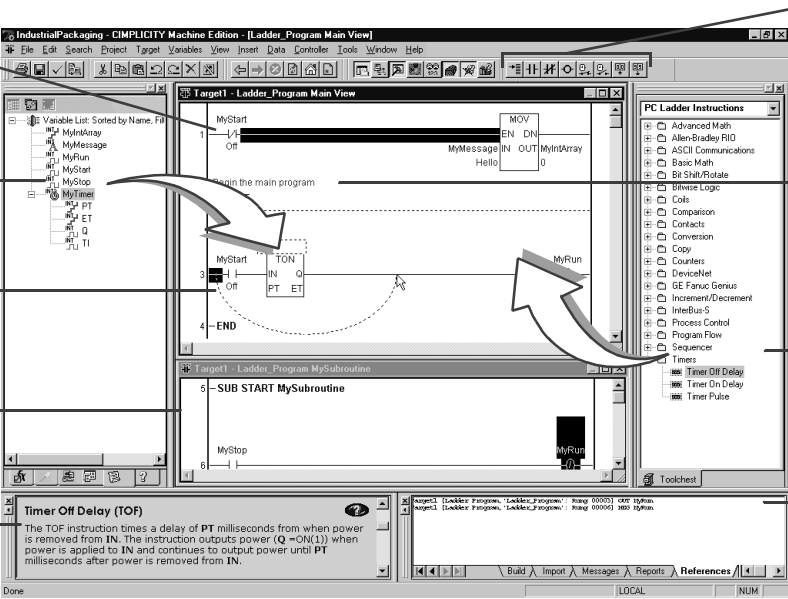
To create a new subroutine or action

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 choose 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 pointing to specific features:

- Click a shunt and type mnemonics to insert instruction (QuickEdit).** Points to a rung with a shunt and a MOV instruction.
- Drag variables to instruction parameters to configure them.** Points to a variable being dragged to an instruction parameter.
- Drag branches around instructions to create parallel execution paths.** Points to a branch being dragged around an instruction.
- Display subroutines and actions in separate editor windows.** Points to a separate window showing a subroutine.
- The Companion tracks your moves providing information on what you click.** Points to the Companion window at the bottom.
- Insert rungs and common instructions with the Ladder Instruction toolbar.** Points to the PC Ladder Instructions toolbar on the right.
- In-place edit descriptions for rungs, variables, and your ladder program.** Points to a rung description being edited.
- Drag instructions from the Ladder drawer of the Toolchest.** Points to an instruction being dragged from the toolchest.
- Locate all instances of a variable (References) or syntax errors (Build).** Points to the References and Build buttons at the bottom.

To insert a rung

- In the ladder logic, right-click a rung, shunt, label or description and choose 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 on a rung and type instruction mnemonics separated by semicolons, and then press ENTER. The instructions specified by the mnemonic string you entered will be 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 52). 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 - CIMPlicity Machine Edition - [Ladder_Program Main View]' window. The main area displays a ladder logic program with four rungs. Rung 1 is 'Initialize the message buffer' with a 'MOV' instruction. Rung 2 is 'Begin the main program' with a 'START' instruction. Rung 3 is a timer 'MyTimer' (TOF) set to 5000. Rung 4 is a 'MyTimer' (TON) set to 5000, which is energized by 'MyStart' and 'MyRun'. 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.

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

New or edited rungs are indicated with alternate color bold lines until you Write Changes.

Power flow is indicated with bold colored lines.



Lightning bolts in each corner indicate the Controller's status and change to reflect fault conditions.

Data values are displayed in real time as they change each scan by the Controller.

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

Variable	Value
MyTimer	PT: 5000, ET: 5000, Q: On, Ti: Off
MyTimer	PT: 5000, ET: 5000
MyTimer	On
MyTimer	Off
MyTimer	Off

To go online to the Controller's ladder program

- In the  Project tab of the Navigator, right-click the  target and choose Go Online. (Before you can go online to the Controller, you must first download the project.)

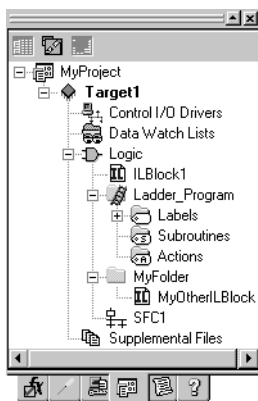
The Controller's ladder program appears in the Ladder Editor. When the Controller is running, power or state flow is graphically indicated and data values updated.

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 bBlock 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, etc.

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

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 and choose **New**.
2. (Optional) Enter a name for the new IL block then press ENTER.

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 IL Editor window titled "IndustrialPackaging - CIMPLICITY Machine Edition - [MyILBlock]". The main editor area contains assembly code with comments. Annotations point to various features:

- Drag variables to the IL Editor to complete IL statements.**: An arrow points from the "Variable List" on the left to the code.
- Multiple-line block comments to document your IL block.**: A bracket highlights a multi-line comment block at the top of the code.
- In-line comments for individual IL statements.**: An arrow points to a comment on a single line of code.
- Copy and paste an IL statement(s) from the IL editor to other sections or IL blocks.**: An arrow points to the copy icon in the toolbar.
- Simply begin typing for a smart list to appear.**: An arrow points to the start of the "LD" instruction.
- The Companion tracks your moves providing information on what you click.**: An arrow points to the "LD (operand)" instruction in the Companion window.
- Locate all instances of a variable (References) or (Build).**: An arrow points to the "Build" button in the bottom right of the Companion window.

The code in the editor is as follows:

```
(*
The following is one way to solve ((A + B) x C) + (A - (D x C) + B).
This example shows how the accumulator and the stack accumulator are used in IL logic.
*)
LD C      'Load C into accumulator.
MUL(B)   'Push C onto accumulator stack. Load B into accumulator.
ADD A    'ADD A to B; load into accumulator.
)        'Pop value of accumulator stack; multiply this value by the value in the
         'accumulator; load into accumulator.
ADD(B)   'Push accumulator value onto accumulator stack. Load A into accumulator.
SUB(B)   'Push accumulator value onto accumulator stack. Load B into accumulator.
ADD(D)   'Push accumulator value onto accumulator stack. Load D into accumulator.
MUL C    'Multiply C by D; load into accumulator.
)        'Pop value of accumulator stack; add this value to the value in the
         'accumulator; load into accumulator.
)        'Pop value of accumulator stack; subtract this value from the value in the
         'accumulator; load into accumulator.
)        'Pop value of accumulator stack; add this value to the value in the
         'accumulator; load into accumulator.
)        'Pop value of accumulator stack; add this value to the value in the
         'accumulator; load into accumulator.
)
ST mydint 'Load value of accumulator into DINT variable mydint.
```

The Companion window shows the instruction "LD (operand)" with the following details:

- accumulator ← operand
- Load a value in the accumulator.
- Operand: BOOL, DINT, LREAL variable, expression or constant.

The bottom right of the Companion window shows the "Build" button and other options: Input, Messages, Reports, References.

To insert an IL instruction

1. In the IL Editor, just begin typing. A smart list will appear 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.

To convert a name to a variable, simply right-click the name and point to **Create "name" as**. Choose a data type from the list that appears.

Working with the IL Editor - Online

The IL editor allows you to view the execution of an IL block as it runs. This is illustrated in the following diagram.

The screenshot shows the IL Editor interface with the following components and annotations:

- Project Tree (Left):** Shows a tree view with 'myProject' expanded to 'Target1', then 'Logic', 'ILBK2', and 'SFC1'. A callout points to the 'Target1' icon, stating: "Online status is indicated by the color of the target icon."
- Code Editor (Center):** Displays IL code with a grey background. A callout points to this background, stating: "The grey background indicates that the editor is online with the controller and is read-only." The code includes:


```

      Created: Monday, May 06, 2002
      -----
      LD 2      'Load the constant "2" into the accumulator
      EXPT 8    'Calculate the expression "2 to the power of 8" and load the result into
              'the accumulator.
      ST MyDINT 'Copy the value of the accumulator to the DINT variable MyDINT
      DINT MyDINT = 256

      LD #True  'Load the system variable "#True" into the accumulator
      ST MyBOOL 'Copy the value of the accumulator to the BOOL variable MyBOOL
      
```
- Block Properties (Bottom Left):** Shows 'Name: ILBK2' and 'Description'. A callout points to the 'Name' field, stating: "Variable types and their data values are displayed in real time as they change each scan by the Controller."
- Inspector (Bottom Left):** Shows a list of files being transferred. A callout points to the 'MyBOOL' entry, stating: "Right-click any BOOL variable to force it On/Off or to turn it On/Off."
- Data Watch Table (Bottom Right):** A table with columns 'Variable Name', 'Address', and 'Value'. It shows:

Variable Name	Address	Value
MyBOOL		On
MyDINT		256

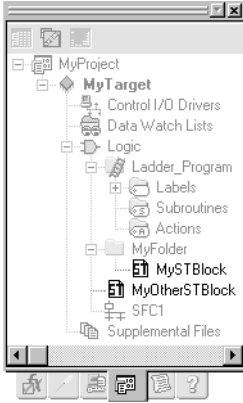
 A callout points to this table, stating: "Use the Data Watch tool to view or set data values for any variable(s) in your project."

To go online to the Controller's IL

1. In the Project tab of the Navigator, for each of the following right-click the target and choose:
2. Validate (F7).
3. Download (F8).
4. Run (F9)
5. 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 on the variable. You can also change and/or force BOOL variable states.

STRUCTURED TEXT EDITOR



Navigator: Project tab
ST block nodes

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 bBlock running in the Controller (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 an ST block is named 'MAIN', after downloading, the execution of every scan will begin the ladder program, then the SFC program(s), then the named 'MAIN' ST block.

Some of the features in ST are: timers, arrays, bitwise access and User Defined Function Blocks (UDFBs). The Return keyword causes an early exit from a subroutine.

Note: Each UDFB is a callable subroutine to and from which you can pass parameters.

Want to know more? In the Help Index, look up "ST Editor" or "UDFB".

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 then choose ST Block.
2. (Optional) Enter a name for the new ST block.

Your new ST block appears under the Logic folder or under the 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 window titled "Sample ST - Lunar Lander - CIMPLICITY Machine Edition - mySTBK". The main editor area contains the following code:

```

(*
This example shows some common ST instructions, such as a complex IF statement,
an absolute value instruction, a FOR loop, a call to a UDFB and a DINT-to-LREAL
conversion instruction.
*)
if Fuel < 0.0 then 'ZF statement
  Status := 5; 'Assign the DINT value 5 to the DINT variable "Status"
  if abs(GForce) > GForce_Die THEN 'ELSEIF statement
    Status := 4; LREAL GForce_Die
  else
    Status := 2;
end if;
for MyDint := 1 to 10 do 'FOR loop iteration statement
  PID::PID( IN PID.Clip_Min := 0.0 , PID.Clip_Max := Thrust_Max , PID.PV := Vel_Current,
  PID.Reset := #ALM_OFF, PID.SP := Vel_Command, _OUT Thrust := PID.CV);
  (* The above statement is a call to the UDFB instruction in the Toolchest named PID.
  The input parameters are Clip_Min, Clip_Max, PV, Reset and SP.
  The output parameter is CV. *)
  Thrust_Max := int_to_real(MyDint);
  'The above statement converts the value in the DINT variable MyDint to
  'an LREAL value and assigns the result to the LREAL variable Thrust_Max.
end for;

```

Annotations and their corresponding actions:

- Drag variables to the ST Editor to complete ST statements.**: Points to the Variable List on the left, which includes variables like Accel_Current, Accel_Initial, Alt_Current, Alt_Initial, Alt_Previous, Alt_Slower, AutoPilot_Enabled, Fuel, GForce, GForce_Die, and GForce_Max.
- Multiple-line block comments to document your ST block.**: Points to the multi-line comment at the top of the code block.
- Hover over a variable with the mouse cursor to see its data type.**: Points to the variable `LREAL GForce_Die` in the code.
- Copy and paste ST statements from the ST editor to other sections or ST blocks.**: Points to the copy and paste icons in the toolbar.
- In-line comments for individual ST statements.**: Points to the comment `'The above statement converts the value in the DINT variable MyDint to 'an LREAL value and assigns the result to the LREAL variable Thrust_Max.`
- The Companion tracks your moves providing information on what you click.**: Points to the "FOR" instruction in the Companion window, which shows details for the `FOR ... DO` construct.
- Locate all instances of a variable (References) or (Build).**: Points to the "References" window, which lists all instances of variables like `Target1 - Logic`, `Target1 - I/O`, `Target1 - Structured Text`, and `Target1 - SFC`.

To insert an ST instruction

1. In the ST Editor, just begin typing. To obtain a list of valid variables, right-click anywhere and choose **Insert Variable**. Enter or choose a variable from the list that appears.
2. To obtain a list of valid mnemonics, right-click anywhere and choose **Insert Keyword**. Type or choose a keyword from the list that appears, and then press ENTER.

Working with the ST Editor - Online

The ST editor allows you to view the execution of an ST block as it runs. This is illustrated in the following diagram.

The screenshot shows the SIMPLISITY Machine Edition interface. The Project Navigator on the left shows a tree structure with 'MySTBlock' selected. The main editor displays a ladder logic program with a 'WHILE' loop and a 'DO' block. A context menu is open over the 'WHILE' block, showing options like 'Watch', 'Turn On', and 'Turn Off'. The Data Watch tool at the bottom right shows the current values of variables: DintArray (1, 2, 3, 4, 5, 6, 7, 8, 9, 10), Index (10), Sum (65), and MyBOOL (On).

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

The grey background indicates that the editor is online with the controller and is read-only.

Variable types and their data values are displayed in real time as they change each scan by the Controller.

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

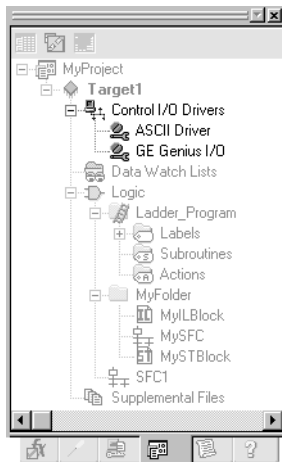
Use the Data Watch tool to view or set data values for any variable(s) in your project.

To go online to the Controller's ST

1. In the Project tab of the Navigator, for each of the following right-click the target and choose:
2. Validate (F7).
3. Download (F8).
4. Run (F9)
5. 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 on the variable. You can also change and/or force BOOL variable states.

CONTROL I/O DRIVERS



Navigator: Project tab
Control I/O nodes

A control program (SFC, ladder, IL or ST) 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 and choose "Control I/O Drivers".

I/O Drivers

Logic Developer - PC comes complete with a constantly growing library of drivers. Some of these drivers require no additional equipment to be installed on the target computer including the following:

- **ASCII:** Allows 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:** Allows 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 to be installed in the target computer. Logic Developer - PC supports most of today's most popular I/O networks including the following:

- | | |
|--|--|
| ▪ Allen-Bradley RIO | ▪ Interbus-S I/O |
| ▪ DeviceNet I/O¹ | ▪ Legacy DeviceNet I/O |
| ▪ Ethernet I/O¹ | ▪ Legacy PROFIBUS I/O¹ |
| ▪ GE Fanuc CNC I/O | ▪ Modbus Slave |
| ▪ GE Genius I/O | ▪ Modicon MB + Distributed I/O |
| ▪ GE Fanuc Series 90-30 I/O^{1,2} | ▪ Modicon Quantum-800 Series I/O |
| ▪ Honeywell SDS | ▪ PROFIBUS I/O^{1, 2} |

1. Also supported for Windows CE targets

2. Also supported for the Hard Real-Time Controller






To add a Control I/O driver to a target


- Right-click the  Control I/O Drivers node, point to New Driver then 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 portrayed as 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 and choose “Control I/O Tool”.

Working with Control I/O

The Control I/O tool interacts with other Machine Edition tools to speed the creation of your I/O system. The following picture illustrates some of the operations you can perform.

Drag variables from the ladder editor to Control I/O terminals.

Use the Variable List to create variables then drag them to I/O terminals.

Double-click a node to configure a driver, card, rack or slot.

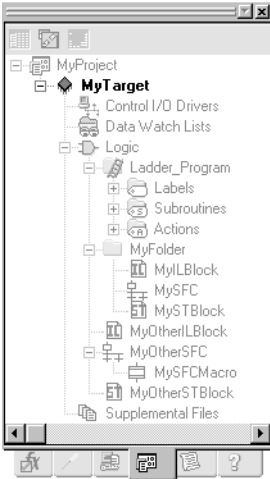
View and edit data values for variables mapped to I/O terminals.

Variable	Value
MyRun	0
MyStart	0
MyTimer	0
MyTimer [1]	0
MyTimer [2]	0
MyTimer [3]	0
MyTimer [4]	0


To configure an I/O system with the Control I/O tool

1. Right-click a driver, card, rack, or slot and choose Add xxx to build an I/O tree.
2. Double-click a driver, card, rack, or slot node to configure it.
3. Map variables to I/O terminals.

CONTROLLER

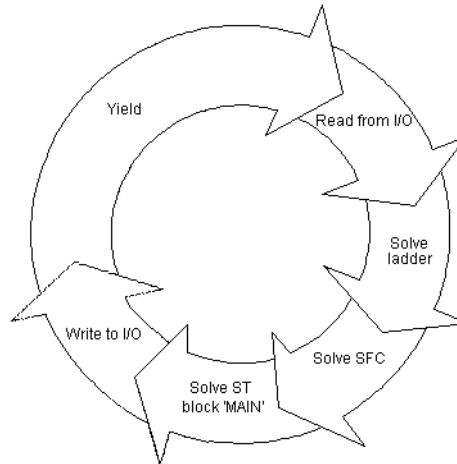


Navigator: Project tab
Target node

The Controller is a Runtime 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.
- Interfaces with Control I/O drivers.
- Updates the target's runtime database, which is shared with the View Runtime (see page 72).


The following diagram illustrates the Controller's scan cycle.



The Controller can be hosted on a:

- *Windows NT target.* The Controller appears as a Logic Developer - PC system service.
- *Windows NT target with VenturCom RTX.* The Hard Real-time Controller appears as a Logic Developer - PC system service.
- *Windows CE target.* The Windows CE Controller appears as a Logic Developer - PC program running under Windows CE.



You specify a target when creating a project. The Hard Real-time Controller is a special add-on configured only on Windows NT targets. In all cases, the Controller automatically starts when the target operating system is started.

 **Want to know more?** In the Help Index, look up Controller and choose “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 choose 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 opens. 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 choose Download.

All components on the selected target are validated then downloaded to the specified computer, if no errors exist. 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

Run Mode Store (RMS) functionality lets you 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, a Run Mode Store dialog box appears with the following options.

- *Attempt a Run Mode Store:* Machine Edition will attempt a Run Mode Store operation. That is, it will try to download your logic to the Controller while the Controller is running.
- *Stop the controller, then download:* Machine Edition will not attempt to use the Run Mode Store feature. Instead, it stops the Controller before initiating the download operation.


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, choose "Run Mode Store (RMS)".

To run the Controller


- In the  Project tab of the Navigator, right-click a  target, point to **Online or Offline Commands**, and choose **Start Runtime**.

The ladder program, SFC program(s) and the ST block named 'MAIN' begin executing on the target computer or unit. If the target also has an HMI component (see page 72), the View Runtime is started at the same time as the Controller.

To validate, download and run a project, right-click a  target and choose **Download and Start**.


To configure an HRT Controller

Follow these steps if you've purchased the optional VenturCom RTX (HRT Controller).

1. Right-click a  target and choose **Properties**.
2. In the **Inspector**, set the **Hard Real Time** property to **True**.

Any Controller operations carried out will now attempt to use an HRT Controller on the target computer. In order to use the HRT Controller, the VenturCom RTX software must also be installed on that target.

To allow an HRT Controller to run following a Windows NT failure

1. Right-click a  target and choose Properties.
2. In the Inspector, set the Delay Controller Shutdown property to True.

Any control programs (ladder, IL, ST or SFC) that are running in the specified HRT Controller will continue to run in the event of a Windows NT or hard drive failure. If the Delay Controller Shutdown property is set to False, the Controller will execute a graceful shutdown if Windows NT fails.


Warm Standby

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

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

Working with Warm Standby

Note: You must set up the warm standby hardware before you can configure the warm standby program.



 **Want to know more?** In the Help Index, choose "Warm Standby".

OPC Servers and Warm Standby

Warm Standby is fully supported by the Logic Developer - PC OPC server for SIMPLICITY Machine Edition. When a Warm Standby configuration file is detected, the OPC server:

- Generates internal objects so that it will recognize all Warm Standby configurations as logical controllers to OPC clients.
- 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 Warm Standby

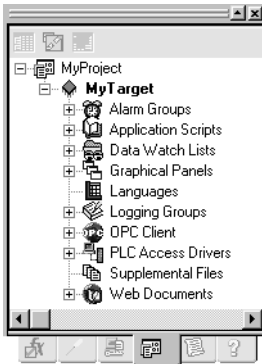
1. In the  Project tab of the Navigator, right-click a  Windows NT or Windows CE target and choose Properties.
2. In the Inspector, scroll to the **+Warm Standby** property and double-click it to expand the group. The **Controller Mode** property appears under the **-Warm Standby** group.
3. Click the Controller Mode property and choose a value from the list.

To configure an OPC server for Warm Standby

The configuration is specified in a user-defined XML file with the name WarmStandby.xml that must reside in the same folder as the LogicPC_OPC.exe file (that is, on the computer hosting the OPC server). In a default install of Machine Edition, this folder is C:\Program Files\Cimplicity Machine Edition\OPC.

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 eventually launch the project into View Runtime. For Windows NT/CE targets, Web Documents adds the option of global accessibility by publishing data from Machine Edition projects to the Internet for remote viewing.

View supports HMI creation for Windows NT, Windows CE, and QuickPanel targets. QuickPanel targets share most of the same tools and editors with NT and CE targets; for more information, see “QuickPanel Applications” on page 76.

The following are included with View:

-  Panel Editor
-  Script Editor
-  Alarms
-  PLC Access Drivers
-  Language translation support
-  Logging (NT/CE targets only)
-  OPC Client & Servers (NT/CE targets only)
-  Web Documents (NT/CE targets only)
-  View Runtime (NT/CE targets only)
-  Message groups (QuickPanel 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 choose 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 a template that includes an HMI 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 is opened in the Navigator.

PANEL EDITOR

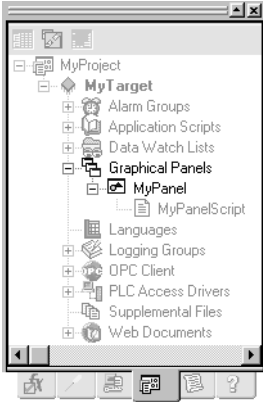
The Panel Editor is used to develop the layout of each panel in your application. With the Panel Editor, you create graphical objects and animate them with real-time data to represent actual processes.

Panels

Panels are the screens or windows for your HMI component. Each panel can be designed to represent a different view of your plant or process.

- **Graphical Objects:** Bitmaps, geometric shapes, industrial illustrations, and real-time charts are among the objects that can be added to a graphical panel.
- **Animation:** Graphical Objects can be configured with up to nine different types of animation to move, change color, display data, trigger scripts, and receive input during Runtime.

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



Navigator: Project tab
Panel Nodes

To create a new panel

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

A new panel is added beneath the Graphical Panels folder in the Navigator. If you've created a project using a template, the Graphical Panels folder most 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 will open in the Panel Editor.

To configure a panel

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

Working with the Panel Editor

The Panel Editor interacts with all of the Machine Edition tools to aid 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”.

The screenshot shows the SIMPLICITY Machine Edition interface. The main workspace displays a 'Main Tank' panel with a graph showing data over time and three pumps (Pump 1, Pump 2, Pump 3) connected to a 'Pump Flow Meter'. The interface includes a top toolbar, a left sidebar with a Project Navigator, a bottom-left Inspector, and a bottom-right Feedback Zone. A 'View Expert Objects' drawer is open on the right, showing various graphical objects like Animated Items, Bar Graphs, Buttons, IO Button, Momentary Switch, Mushroom Button, Selection Switch, Start Stop Button, Two Button, Meters, Misc, Panels, and Selectors. A white arrow points from the View Expert Objects drawer to the Panel Editor workspace.

View toolbars provide drawing and object manipulation tools.

Panels are listed in the Project tab of the Navigator.

Configure graphical objects in the Inspector.

Chart and Alarm Objects update with real time data.

Drag items from the View Expert Objects drawer of the Toolchest to the Panel Editor.

The Companion displays information for the currently selected object, tool, or property.

The Feedback Zone updates with information when you validate, download, or run your project.

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 will display instructions on how to use the selected tool.

2. Follow the instructions in the Companion to draw your object.

Once your object is drawn, set its properties in the Inspector.

To animate an object

- In the Panel Editor, double-click a graphical object.

The Animation Properties dialog box opens. The settings of each animation type is grouped on its own tab. Animations become active when the project is launched with View Runtime.

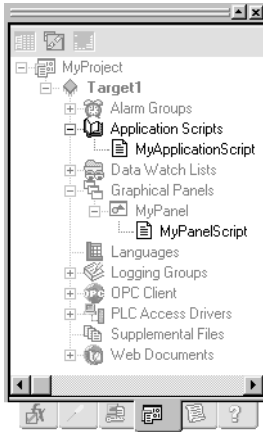
SCRIPT EDITOR

The Script Editor is a text editor that supports a various scripting languages, functions, keywords, and math operators. On Windows NT targets, you can also use Structured Query Language (SQL) in a script to access a database for which an Open Database Connectivity (ODBC) driver exists.

Scripts

Scripts are short executable programs comprised 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 the application is started (in View Run-time), when the application is shut down, or based on a frequency or condition while the application is running.
- **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 NT/CE targets only.) Touch animation scripts are activated during Run-time by keystrokes and on-screen buttons. They are written directly in configuration dialog boxes of graphical objects.
- **Global Function Library Scripts:** (Windows NT targets only.) Global Function Library scripts are associated with an entire target. By default, they are executed before any other scripts. However, they *can* be configured to run periodically or conditionally, based on the state of the application. Global Function scripts only work with scripts in Active Scripting languages, like VBScript.



Navigator: Project tab
Script nodes

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 choose either VBScript or ViewScript. (For more information on scripting, see page 61.)
2. Type a name for the new application script and press ENTER.
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, in other words, the activation type.

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 61.)
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, in other words, the activation type.

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 the  Global Functions node, point to New Library, and choose 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 NT targets, Machine Edition supports two types of scripting languages: ViewScript and VBScript Active Scripting languages (using Microsoft's Active Scripting engine). On Windows CE targets, Machine Edition supports only proprietary ViewScript language.

On QuickPanel targets, Machine Edition uses the proprietary QPScript language (see page 82).

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.

View

Script Editor

- *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 NT targets. Currently, only the VBScript Active Scripting language is supported.

By default, new projects will use VBScript. This default may be overridden by setting a user-preference.

Note: A target can contain both scripts written in ViewScript and scripts written in VBScript. However, troubleshooting such a project can be quite difficult, so 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 detailed 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.

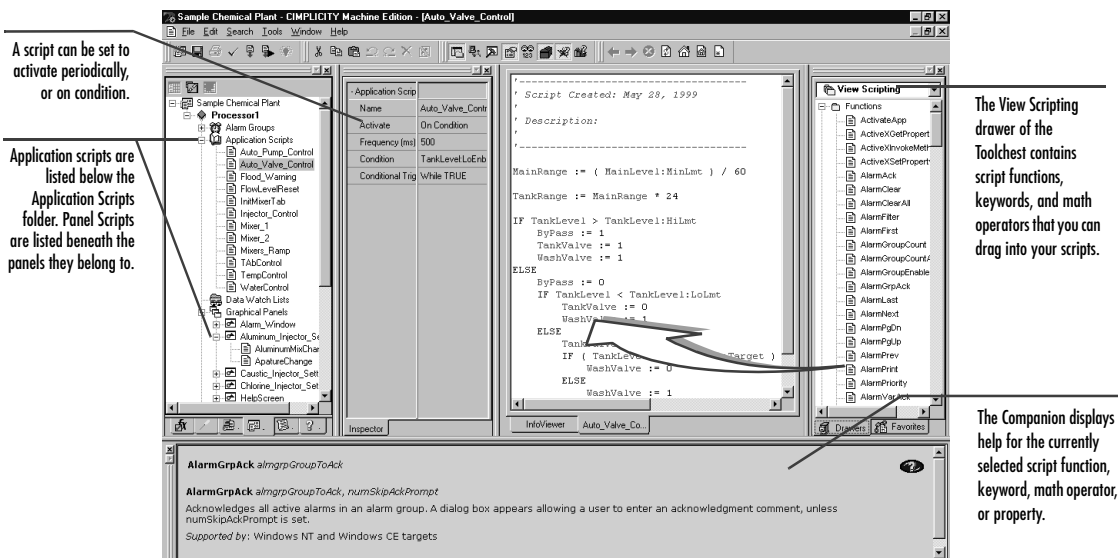
- Array objects.
- Structure objects.
- Application objects.

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 in which you can insert script words when writing scripts.

- If you are very familiar with the operators, functions, and keywords used in the Script Editor, simply start typing your script directly in the Script Editor.
- If you are not familiar with the scripting language of the Script Editor, you can easily use the right-click menu to access this information. Just right-click where you want to insert a function, operator, or keyword and select Insert. Then point to whatever it is you want to include in your script and make a selection from the list that appears. This also makes available names of variables, panels,

scripts, alarm groups and logging groups within the project that you can use in your script. For a description of a script function, keyword, or math operator, while in the Script Editor, place your cursor on the word for which you need help 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 one.

LOGGING

(Windows NT/CE targets only)

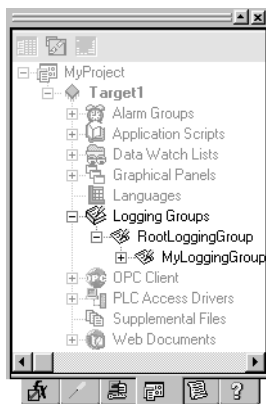
Logging keeps a history of data changes in your project so the data can later be analyzed. Logging groups and strategies are used to sort and manage logged data.

? **Want to know more?** In the Help Index, look up logging and choose “Data Logging”.

Logging Groups

Logging groups help to sort and organize your project data. Data can be collected in a default logging group or, if you want some variables to have different logging strategies, you will need to create a hierarchy of logging groups. This is recommended anyway as it helps to organize your logged data.

Logging Strategies define when data gets logged for a group of variables. A group can be configured to log data periodically, on data changes, or on a condition.



Navigator: Project tab

Logging Group nodes

To enable logging of a variable

1. In the Variables tab of the Navigator, right-click a variable and choose Properties. The Inspector will open if it isn't already.
2. Set the Enable Logging property to True.

Logging is enabled for the variable, with RootLoggingGroup as the default logging group.

To create a new logging group

1. In the Project tab of the Navigator, right-click the logging group node you want the new group to exist beneath hierarchically, and choose New.
2. Type a name for the new logging group and press ENTER.

Your new logging group appears beneath the logging group node.

To edit logging group properties

1. In the Project tab of the Navigator, right-click a logging group node and choose Properties.
2. Edit the properties in the Inspector.

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 isn't already open.

ALARMS

This section describes how alarms work for Windows NT/CE targets. For information on alarms in QuickPanel targets, see page 78.

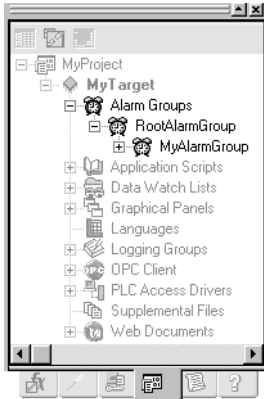
Alarms let users know when conditions in a monitored process have gone outside the acceptable range. You configure alarms through properties of a variable. When a variable satisfies one of its alarm conditions, it goes into an alarm state.

Alarm Objects

Active alarms can be displayed in runtime with *Alarm Objects*. Alarm Objects dynamically update with information about alarm states. To set this up, all variables with alarming conditions need to be assigned to an *alarm group*.

Alarm Groups

Alarm groups help you organize related alarm variables. Having more than one alarm group makes it easier to manipulate and view alarm data at runtime. A history of alarm group data can be logged daily for future analysis. You can also have an e-mail sent if any of an alarm group's alarms go into an alarm state.



To set an alarm condition

1. In the Variables tab of the Navigator, right-click the variable for which you want to set an alarm condition and choose Properties. The Inspector will open if it isn't already.
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, etc.) that you want to configure to set its alarm parameters.

These parameters establish the ranges that place the variable in an alarm state.

To create an alarm group

- In the Project tab of the Navigator, right-click the alarm group under which you want the new group to exist, and choose New. Type a name for the new alarm group and press ENTER.

Your new alarm group appears beneath the alarm groups node.

To create an alarm object

1. In the Panel Editor, right-click and choose Alarm. The Alarm tool will be selected.
2. Draw an Alarm Object, using the information in the Companion as a guide.
3. Right-click the Alarm Object and choose Properties. The Inspector will open if it wasn't already.
4. Set the Group property to the alarm group you want the Alarm Object to display information about in Runtime. During Runtime, the Alarm Object will update with the alarm status of variables belonging to the associated alarm group and its subgroups.

PLC ACCESS I/O

PLC Access I/O is used to let your Machine Edition project communicate with I/O devices controlled by something outside the Machine Edition environment.

Typically, that ‘something’ is a PLC: a special hardware device that controls input and output for other, simpler devices. For information on I/O communications with I/O devices, see “Structured Text Editor” on page 46.




Drivers

PLC Access I/O is organized into Drivers and Devices:



- **Drivers** are located in a target's PLC Access Drivers folder and represent the drivers the target uses to communicate with devices. Windows NT and CE targets can have many drivers; QuickPanel targets can have only one.
- **Devices** represent the individual devices that a driver communicates with. All devices that a driver communicates with are located in that driver folder. A driver can have several devices. This allows you to talk to multiple PLCs, or (on Windows NT and CE targets) to get pieces of data from the same PLC at different rates.

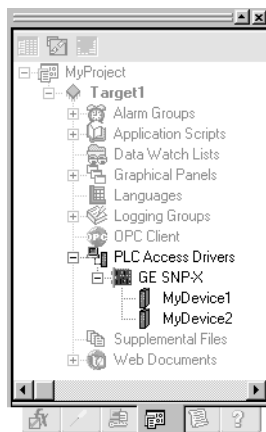
To access data values from devices of a PLC Access driver, you must also create variables with a PLC Access data source. The configuration of these variables is unique for each kind of PLC Access driver; see the online help for more details.

To add a PLC Access driver to a target

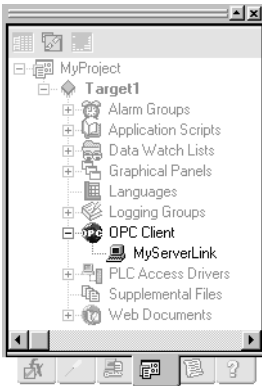
1. In the  Project tab of the Navigator, right-click the  PLC Access Drivers node and choose New Driver.
2. Select a driver from the list provided and click OK.
The driver (and a single device) are added to your project.
3. In the Inspector, configure the driver. If applicable, click the  button in the driver's Configuration property and enter values in the dialog box.

To add a device to a PLC Access driver

1. In the  Project tab of the Navigator, under  PLC Access Drivers, right-click the driver to which you want to add a new device and choose New Device.
2. Type a name for the Device and click OK.
The device is added to your project. You can now edit the device's properties in the Inspector.



Navigator: Project tab
PLC Access I/O Nodes



Navigator: Project tab
OPC Client Nodes

OPC

(Windows NT/CE targets only)

OPC (OLE for Process Control) is an industry-standard protocol that lets software applications 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 other OPC Clients.
- The OPC Client connects to an OPC Server to gain access to the server's data.

Since 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".

Since an OPC Server and 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'd set one of them up 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 an OPC Client is running on a different computer than the Server, you will have to edit the DCOM Security settings on one or both computers. For details, see the online help.

? **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 lets your Machine Edition application communicate with OPC Servers. This lets Machine Edition act as a client in the OPC Client/Server relationship.

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 choose New Server Link.
2. Type a name for the OPC Server Link and click OK.

The OPC Server link is added to your project. You can now edit its properties in the Inspector.

Machine Edition OPC Servers

Machine Edition OPC Servers let you 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.

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 the configuration of an OPC Client depends on the manufacturer, typically each piece of data retrieved from a server is called 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 can be found in the table below.

Location Syntax

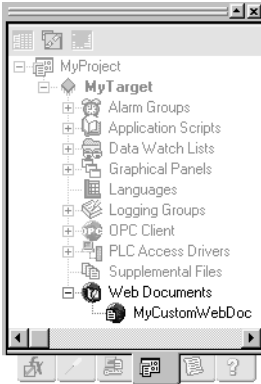
local	<variable name>
remote	<remote machine name>\ <variable name> or <remote machine IP address>\ <variable name>

- *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.

WEB ACCESS

(Windows NT/CE 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 runtime, it automatically launches an integrated web server. Users who connect to this web site are presented with the Site Index, which contains different ways to view process information.



Navigator: Project tab

Web Document nodes

- **Remote Views:** Dynamic graphical recreations of actual panels from the Runtime display.
- **Custom Web Documents:** Web documents that have been generated using templates and special HTML tags that are replaced with current project information from the project each time an instance is generated. You can view the documents online or print them for distribution.
- **Variable Monitor Browser:** An application that allows you to drag and drop variables from Web Documents servers onto four unique viewing tools for dynamic graphical representations of process data.
- **Remote Variable Inspector:** Lets you select a variable from a list of project variables in a project and immediately receive the current value.
- **Web Control:** Lets you 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 allow remote access to your project

1. For each target that you want to allow access to, set the target's **Enable Web Server** property to True.
2. For each item in those targets 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 the Web Documents Site Index

1. Start Internet Explorer 4 or later. If you don't have Internet Explorer 4 or later, you can install it from the Machine Edition installation CD.
2. In the Address box in Internet Explorer, type the computer name or IP address of the Runtime computer.
3. If you're 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.

4. If you're 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 Runtime 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 at once.

To view animated panels remotely

You can select to view graphical panels 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 Runtime 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.

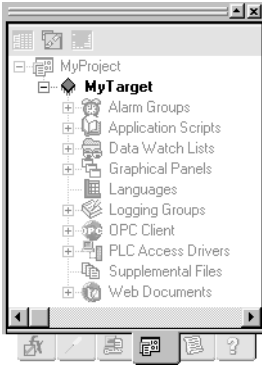
Since the remote viewer uses a different graphics engine to draw and update a graphical panel, there may be a few differences between the remote view of a panel and its actual appearance on an HMI. For example, chart objects on panels are not supported in remote views.

To create a new Custom Web Document

1. In the  Project tab of the Navigator, right-click the  Web Documents node and choose New Web Document.
2. Type a name for your new document and press ENTER. 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 of 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 choose Web Document Files. The Web Document Files folder opens.
3. Place your HTML document and any logos or pictures your HTML document uses in the Web Document Files folder. 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

Target node

VIEW RUNTIME

(Windows NT/CE 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 Runtime 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 choose **Download and Start**. All components on the selected target are validated, downloaded to the specified computer, and the Runtime 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 a project outside of Machine Edition

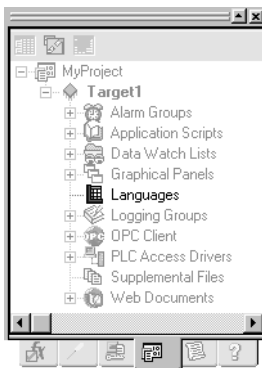
- Click the Start menu, then go to **Programs/CIMPLICITY Machine Edition**. Click **View Runtime**. Runtime starts with the last project that was downloaded to that computer. View Runtime opens with all graphical panels configured to appear at startup (that is, all graphical panels whose Visible at Startup property is True).

To interact with View Runtime

Graphical objects can be configured to allow you to 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
Languages node


LANGUAGES FOLDER

The  Languages item opens a grid that lets you configure multi-language support for an HMI. You set up simple translations between an original piece of text (called the “Source”) and one or more languages. If the target’s Language Translation property to Enabled, all text on the HMI’s graphical panels is then 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’ve made changes. Any 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.

Windows NT/CE targets: Machine Edition includes the entire language table when downloading the project to the target. Translation occurs dynamically at runtime. 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. You cannot dynamically change the current language at runtime.

 **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. When Language Translation is enabled, this is the first language used as the "Current Language".

Source	Italian	French	Japanese
Tank	Serbatoio	Réservoir	タンク
### # C			### # C
0 FT	0 piede	0 pied	0 フット
24 FT	24 piedi	24 pieds	24 フット
A			A
AL			AL
ALUMINIUM	ALLUMINIO	ALUMINIUM	ALUMINIUM
Alarm	Segnale d'allarme	Alarme	アラーム
Aluminum	Alluminio	Aluminium	アルミニウム
Auto-	Auto-	Auto-	オート
Automatic	Automatico	Automatique	オートマチック

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

If both the development and runtime computers support Unicode, you can use non-English character sets in the grid (such as Japanese, pictured). For example, In Windows 2000 you can use the IME, configured in the Regional Settings setup in the Control Panel.

- Add Row Ctrl+Alt+A
- Delete Row Ctrl+R
- Add 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

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 select **Add 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 select **Set Column Name** (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

- In the Languages Grid click and hold the header of the column you want to move. 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 select **Update Source Column**.
View searches every graphical panel in the HMI (regardless of whether or not the panel changed since the last validation) and makes sure 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 select **Set Column Font**.
A Font dialog box appears.
2. In the Font dialog box, 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're done.
The selected font set is used to display the translated text both in the Languages grid and (in Windows NT/CE targets) the downloaded HMI. Other font attributes—such as font sizes and styles—only affect how the text appears in the grid at development time.

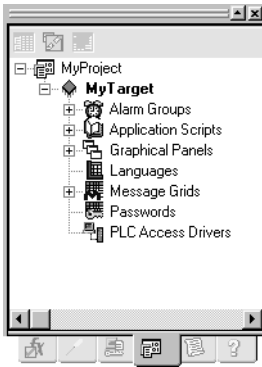
To set the default language column

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

QUICKPANEL APPLICATIONS



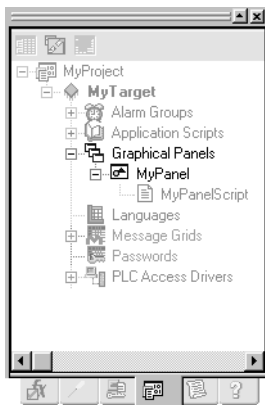
Navigator: Project tab

QuickPanel nodes

QuickPanel applications are HMI (human-machine interface) applications for QuickPanel units. CIMPLICITY View supports development of HMI applications using standard View and Machine Edition tools. QuickPanel application development includes the following items and features:

- **Panel Editor:** Design each graphical panel for your QuickPanel display, using a wide selection of graphical objects including a variety of user-configurable buttons.
- **Grid Editors:** Several items in QuickPanel targets use a grid editor. This editor looks and works much like a spreadsheet application. Items that use grid editors include Word Alarm groups, Bit Alarm groups, LMD (Local Message Display) groups, TMD (Triggered Message Display) groups, Selector Switches, Trend displays, Bar Graphs, the Passwords list, the keypad assignment feature, and LID (Local Image Display) groups.
- **Integration with Logic Developer - PLC:** If you also have Logic Developer - PLC installed, you can use variables on a GE PLC target directly in your QuickPanel application. To use this feature, both the GE Fanuc PLC 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 PLC target. (For more information about using GE PLC variables, see *GFK-1918: Getting Started Logic Developer - PLC*.)
- **Integration with the Machine Edition environment:** QuickPanel development uses all the standard Machine Edition tools and utilities, including the Inspector, the Feedback Zone, the Toolchest, and a project's Variable List. For more information, see Machine Edition Environment page 10.

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



Navigator: Project tab
Panel Nodes

QUICKPANEL PANEL EDITOR


The QuickPanel Panel Editor is used to create and edit panels for your QuickPanel HMI (Human Machine Interface). Each panel is a single screen or window of your HMI. You draw and configure graphical objects on the panel to create each screen. Navigation between panels is done through the Goto button graphical object.

Graphical objects are the various graphical elements of the HMI screen, such as rectangles, polygons, buttons, and alarm displays. All objects are available from the QuickPanel tools toolbar. Many items (such as buttons) perform actions when you tap them at runtime.

? Want to know more? In the Help Index, look up QuickPanel, and choose “QuickPanel Panel Editor”.


The Panel Editor displays the current panel as it will appear on the QuickPanel target unit when you download the project. In addition, the Panel Editor adds a tiling grid and a drawing grid. The tiling grid lets you align graphical objects with the touch-sensitive cells on the surface of a QuickPanel unit’s display.

Panels on QuickPanel targets

Panels are the screens or windows of your QuickPanel HMI (Human Machine Interface). You edit panels in the Panel Editor, where you can draw graphical objects, insert bitmaps, and develop the layout of each panel in the project. In the Navigator, panels are organized under the  Graphical Panels folder.

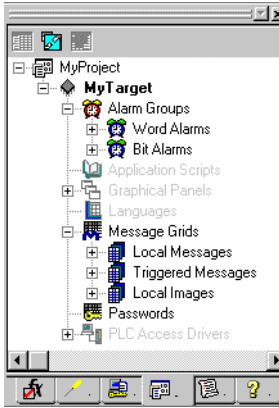
You work with Panels on a QuickPanel target in much the same way as NT and CE targets. Some of the differences are:

- QuickPanel targets support a different set of graphical objects. Many are similar, but some are unique to QuickPanel targets.
- QuickPanel targets do not support Animation in the same way as NT and CE targets. Instead, all object configuration takes place in the Inspector, or an appropriate grid editor (see page 78).

For a description of a panel’s properties, select a property and look in the Companion window. Click  in the Tools toolbar to open the Companion if necessary.

? Want to know more? In the Help Index, look up QuickPanel, and choose “QuickPanel Panel Editor”, “QuickPanel graphical panels”, and “QuickPanel graphical objects”.

GRID EDITORS



Navigator: Project tab
Grid Editor Nodes

The Grid Editor lets you create and edit several items in QuickPanel targets. This editor looks and works much like a spreadsheet application.

Items that use grid editors include:

- Word Alarm groups
- Bit Alarm groups
- LMD (Local Message Display) groups
- TMD (Triggered Message Display) groups
- Selector Switches
- Trend displays
- Bar Graphs
- Passwords list
- External Keypad assignment configuration.

Want to know more? In the Help Index, look up QuickPanel, and choose “Working with QuickPanel grid editors”.

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.

In QuickPanel target, there are two areas in which you configure alarms:



- Create and edit your desired Word Alarm Groups and Bit Alarm Groups, located under the target’s Alarm Groups folder. Each alarm group consists of a table of alarm entries. Each entry in a group specifies a variable, a value of that variable that will trigger an alarm condition, and the corresponding message to display.
- Add an Alarm Window graphical object to one of the QuickPanel application’s panels. This graphical object displays alarm state information at run-time. 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 alarms and choose “QuickPanel Alarms: an Overview”.








To create an alarm group


- 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 choose **New**. Type a name for the new alarm group and press ENTER. Names must be unique within a target.

To create an Alarm graphical object



1. In the Panel Editor, right-click and choose Alarm. The  Alarm tool will be selected.
2. Draw an Alarm Object, using the information in the  Companion as a guide.
3. Right-click the Alarm Object and choose **Properties**. The Inspector will open if it wasn't already.
4. In the Inspector, edit the alarm object's properties as desired.

Message Displays


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  LMD Group,  TMD Group, or  LID Group.


Each graphical object can be associated with only a single message group, but that message group may be associated with many graphical objects. 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 "LMD/TMD/LID Group" property.

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


- A  **Local 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  *LMD Group*.



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

- A  **Local 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  *LID Group*.


An image in an LID 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 LID and choose "LID (Local Image Display) group" and "LID (Local Image Display) grid".

External Keypad Assignment

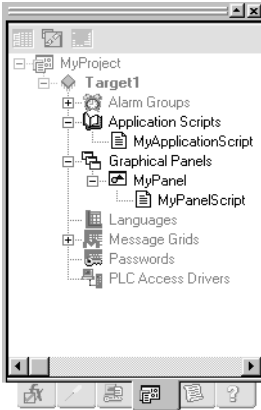
Some QuickPanel units support an external keypad. You enable the keypad by setting the target's External Keypad property to True; this enables keypad configuration elsewhere in the application. There are two ways to assign an action to a keystroke:

- In the graphical panel, graphical objects that can have an assigned keystroke will 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" and look up QuickPanel and choose "Working with QuickPanel grid editors".


SCRIPTS ON QUICKPANEL TARGETS





Navigator: Project tab
Script Nodes

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 called "QPScript". You create QuickPanel scripts using the same View Script editor as Windows NT/CE targets (see page 60).




There are two types of scripts for QuickPanel applications:

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".

To create, open and edit an application script

1. In the  Project tab of the Navigator, right-click the  Application Scripts node, and choose **New Script**.
2. Type a name for the new panel script and press ENTER.
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, and choose **New Script**.
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.

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:

- *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.

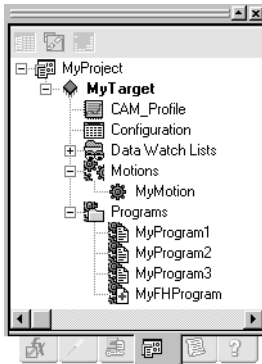
Expressions: There are two general kinds of expressions in QPScript:

- *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

Motion Developer lets you easily program Whedco's IMC, IMCjr (IMJ), and SMJ series or GE Fanuc's S2K 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 choose **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. 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 is opened in the Navigator.

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 89) can make this easy.
4. Validate and download the application to the motion controller.

The Motion Expert wizard will take you through these items step by step.

Supported motion control devices

Motion Developer supports the following lines of motion control devices.

Whedco™ motion control devices	IMC motion controllers IMCjr (IMJ) motion controllers SMJ drives
GE Fanuc™ motion control devices	S2K controller/drive devices S2K drive-only devices


- For the most up-to-date hardware documentation on Whedco motion devices, see the Whedco web site at www.whedco.com.
- Many hardware manuals and specifications for GE Fanuc motion devices can be found on the GE Fanuc web site at www.gefanuc.com.

Communications 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. Your project can have as many targets as you want, however.





Motion targets

A  *Motion target* represents a runtime destination motion device that you program and configure from Machine Edition. You'll need a separate target for each physical motion device you'll be configuring, whether it's 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 **Whedco SMJ** devices and **GE Fanuc S2K drive-only** devices. These targets only have a *Configuration script* (page 86).
- **Controller** targets represent controller/drive motion devices. This includes Whedco **IMC** and **IMCjr (IMJ)** devices, as well as **GE Fanuc S2K controller/drive** devices. These targets have a *Configuration script* (page 86), *Motion Programs* (page 86), and can have several *Motion Blocks* (page 87). In addition, they can make use of a *CAM Profile*. (see page 95).

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 select **Open**.
2. Create or edit the program in the appropriate editor.
3. After editing an item, make sure you 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 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.


Motion configuration



The motion controller  configuration node represents a script used to configure the motion controller when the application is downloaded. This script runs every time the application is downloaded to the motion controller. 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 select **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'll need to edit the script manually. You can do so in the motion script editor (right-click  Configuration and select **Open**).

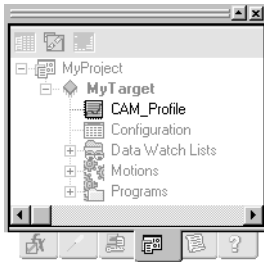
 **Want to know more?** In the Help Index, look up "Motion Controller Configuration".

Motion programs

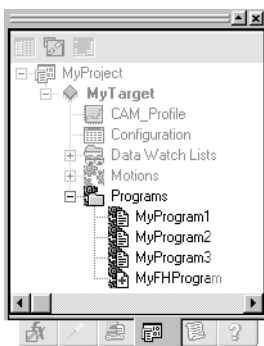
A  program represents one of the motion controller's programs. Programs are scripts that control the operation of a motion controller. With programs, you can create complex behaviors for the controller.

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".



Navigator: Project tab
Configuration node

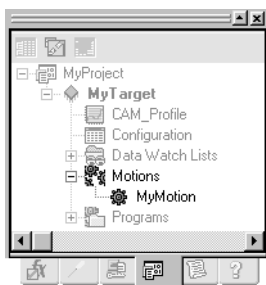


Navigator: Project tab
Motion Program nodes

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 91.
- The Motion Script Editor lets you edit the script for a motion program manually. For details, see page 88.

? **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 90).

You can create a motion block with the New Motion Block wizard (right-click the Motions folder and select 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 selecting **Add**.

Once a motion block is created, you can edit it by right-clicking the Motion and selecting **Open**. You edit motion blocks and motion programs either with the Motion Calculator (see page 93) or in the Motion Developer Script Editor (see page 88).


Motion blocks have the following characteristics and restrictions.

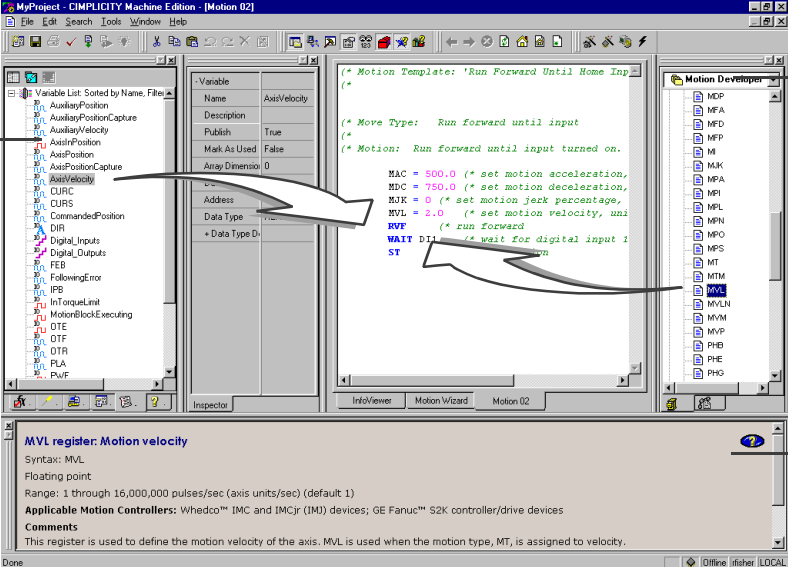
- Motion blocks complete executing one line of code before proceeding to the next line of code.
- Only one motion block per axis can run at the same time as the executing program or programs.
- When a motion block is executed, it overrides the currently executing motion block.
- You cannot use labels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




The screenshot shows the Motion Developer software interface. The main window is titled "MyProject - CIMPLICITY Machine Edition - (Motion 02)". It features a menu bar (File, Edit, Search, Tools, Window, Help), a toolbar, and several panes:

- Variable List:** A tree view on the left showing various variables like AxisPosition, AxisVelocity, etc.
- Inspector:** A central pane showing details for the selected variable, "AxisVelocity".
- Script Editor:** A large text area on the right containing motion script code. A callout points to the code: `MAC = 500.0 (* set motion acceleration, MDC = 750.0 (* set motion deceleration, MJK = 0 (* set motion jerk percentage, MVL = 2.0 (* set motion velocity, uni RUF (* run forward WAIT DT1 (* wait for digital input 1 ST`
- Motion Developer:** A toolchest on the right containing a list of registers, commands, and operators. A callout points to the list: "The Motion Developer drawer in the Toolchest contains registers, commands, and operators you can drag to a motion script."
- Companion:** A window at the bottom showing details for the selected register, "MVL register: Motion velocity". A callout points to it: "The Companion displays help for the currently selected register, command, or operator."

Additional callouts include:

- "You can drag variables from the Variables tab directly into a motion script." (pointing to the Variable List)
- "You can also add items into your script by right-clicking in the editor window." (pointing to the Script Editor)

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

- If you are very familiar with the registers, commands, and operators used in Motion scripts, simply 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 select **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 can be dragged 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

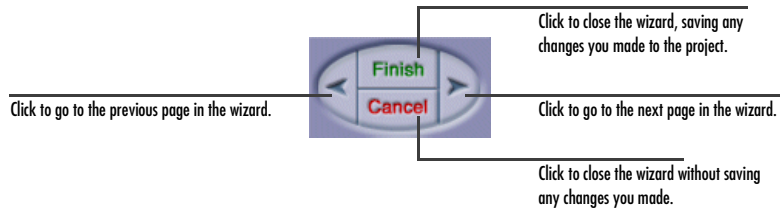
Wizards are a series of screens that take you, step by step, through the creation and configuration of your motion controller application. In Motion Developer, wizards appear in separate windows, so you can switch back and forth between them.

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	Lets you create motion programs as a visual flowchart. See page 91.
Motion Calculator	Lets you define simple, incremental motions for motion blocks. See page 93.
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 need to make selections or enter parameters before continuing, the “next” button will be 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, 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 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**.
- The *Controller Functions* and *Drive Functions* wizards let you 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”.
-

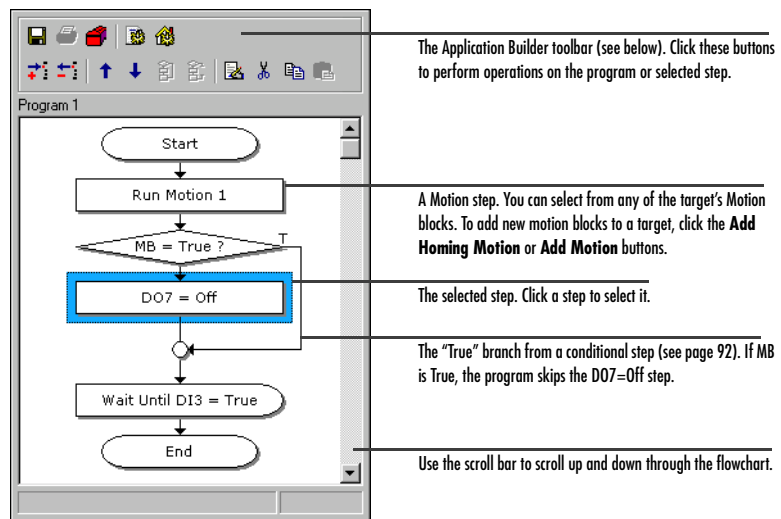
SPECIAL WIZARDS

Two of the wizards in Motion Developer—specifically, the Application Builder and the Motion Calculator—let you edit parts of your motion application in graphical editors.

Application Builder flowchart editor

The Application Builder is a wizard that lets you 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:



In the page itself, two buttons to the right of the flowchart open wizards that let you 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 93).

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.

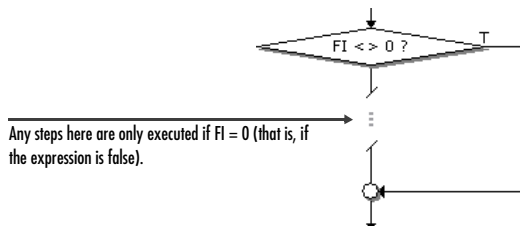
- **Start and End steps:** Flowchart execution begins at the Start step. When execution reaches the End step, the program stops.
- **Logic steps:** Performs a logic-oriented action, such as setting or resetting a particular register.
- **Motion Program steps:** Performs a motion program action, such 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 92 (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:** Lets you insert a comment into the flowchart. Remarks are included in the script downloaded to the motion controller.
- **Comment steps:** Lets you 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-
 - 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 selecting **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 select **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.

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

Typically, you'd 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:

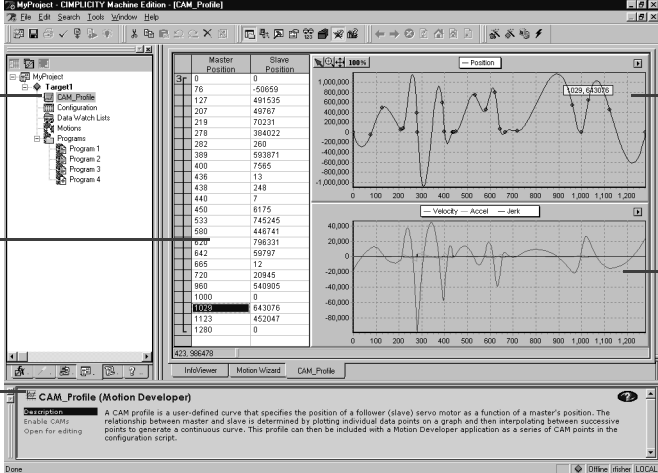
- You can fill in two (and only two) of Velocity, Total Time, and Distance. That is, one of these parameters must be left at 0 and the other two filled in with non-zero values. The Motion Calculator automatically calculates the missing parameter in the Calc column.
- If you leave Acceleration time (Acc. 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.

CAM PROFILE EDITOR

The Motion Developer CAM editor lets you create and edit a single electronic CAM profile for a motion controller target. The 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 edit a target's CAM Profile by double-clicking on its  CAM_Profile node. The CAM editor display looks something like this:



Double-click the **CAM_Profile node in the Navigator to open the CAM editor.**

The CAM profile table allows numeric editing and curve order grouping

Master Position	Slave Position	Curve Order
0	0	3
76	-50659	1
127	491535	1
207	457057	1
219	70231	1
278	384022	1
283	280	1
399	933971	1
400	7565	1
436	13	1
438	248	1
440	7	1
450	6175	1
523	745245	1
580	446741	1
627	796231	1
641	59797	1
665	12	1
720	20945	1
860	540995	1
1000	0	1
1039	643076	1
1123	452047	1
1280	0	1

Curves of the profile can be adjusted and edited graphically

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

The Companion tracks your moves and provides information on what you click.

The CAM_Profile (Motion Developer)
 A CAM profile is a user-defined curve that specifies the position of a follower (slave) servo motor as a function of a master's position. The relationship between master and slave is determined by plotting individual data points on a graph and then interpolating between successive points to generate a continuous curve. This profile can then be included with a Motion Developer application as a series of CAM points in the configuration script.


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 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 configuration script.


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 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 edit a motion controller target's CAM profile

- In the Project tab of the Navigator, under the motion controller target whose CAM you want to enter, right-click  CAM_Profile and choose **Open 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.

6

Local Manager



Navigator: Manager tab

Project nodes

The **Local Manager** is Machine Edition's local systems and project manager. It is the version control, security, and audit trail system for your automation and control development process.

You do not need to use Local Manager to work with Machine Edition products. However, if you have security concerns, or if you want to track your application development process, it can be very useful.

LOCAL MANAGER

The Local Manager refers to Machine Edition's project and systems management features. Local Manager features let you:

- Keep track of changes made to projects and settings, and keep a history of previous versions of projects (see **Version Control**, page 99).
- Control access to projects and settings under CIMPLICITY Manager (see **Access Control**, page 104).
- Monitor and record actions performed by users (see **Audit Trails**, page 107).

The Local Manager must be enabled before you can use its features. While enabled, users of Machine Edition on the local computer will have to log on to the Manager system. As the name implies, the Local Manager only works with projects stored on the local computer.

Want more? Additional system management features—such as scheduled events and a centralized Server for project and database storage—are available with the CIMPLICITY Manager product.

To enable Local Manager





1. On the File menu, point to CIMPLICITY Manager, then select **Enable Local Manager**.

If Disable Local Manager appears instead, then the Local Manager is already enabled.



If Enable Local Manager is grayed out, then the Local Manager components weren't included when installing CIMPLICITY Machine Edition. You will have to reinstall Machine Edition from the installation CD, making sure to enable the Local Manager component.

2. Exit and restart Machine Edition.
3. Log on to the Manager system.

Since this is the very first time the Local Manager has been enabled on this computer, you will have to log in as the Administrator user. Type "Administrator" for the user name, and leave the password blank.

Several new folders are now available. These are  *Access Control*,  *Audit Trail*, and  *Local Manager* (under the  Projects folder).

4. Set up your Access Control system.

Create  Users and  Groups for people who need access to items on the computer (for details, see chapter 7). You can work with other new features (adding existing projects to Local Manager or setting up Audit Trails) later.

Want to know more? In the Help Index, look up project history and choose "Setting up a Local Manager system".

VERSION CONTROL

Version Control is probably Local Manager’s most important feature. With Version Control, all your data is stored in a central location. It also allows you to:

- ensure that only one user can make changes to an item or project at a time;
- keep track of who is currently working on an item or project;
- safely store previous versions of projects; and
- return a project to a previously-archived version.

Note: With Local Manager, only the previous ten versions are retained.

Want to know more? In the Help Index, look up version and choose “How Version Control works under Local Manager”.

All Local Manager projects and settings are stored in a separate database. When you want to work on something under a Local Manager system—whether it’s a project, Access Control settings, or something else—you go through the following steps:

1. **Check out the project or item.**

Local Manager copies the item to a separate working directory and marks the item as being “checked out”. As long as the item is checked out, no other user can make changes to that item.

2. **Make changes to the item.**

Changes are made in the working directory only. At this point, the version stored by the Local Manager system is unchanged.

3. **When you are finished, check in the item.**


Local Manager copies the updated version from the working directory and removes the “checked out” flag. Other users can now check the item out to make more changes later.

Security for Version Control

Security in Local Manager is called *Access Control* and is explained in detail on page 104. Briefly, each user that is created in Local Manager is a member of one or more *groups* that have security permissions assigned to them. These permissions let users of that group perform certain actions. In addition, each project has one or more groups assigned to it. Users can only work with a project if they are a member of a group assigned to that project.

Want to know more? In the Help Index, look up “Access Control Permissions”.

Project history

When you check in a project to the system, Local Manager automatically archives the old version. (Old versions of other items, such as Access Control user/group lists, are *not* archived.) You can get a report of all archived versions of a project by right-clicking it and selecting **History Report**. History reports appear in the Reports tab of the  Feedback Zone.



In the Feedback Zone, right-clicking a previous version of a project gives you the following commands:


- **Get:** Get a copy of the selected version onto the local computer.
- **Label:** Assign a label to the selected version.
- **Make Latest Version:** Make the selected version the current version.

Want to know more? In the Help Index, look up project history and choose “Project History Reports”.

Checking out other items


Projects aren’t the only things you can check out. Local Manager also places the following under Version Control:



- The  Access Control database (see page 104)
-  Audit Trail settings (see page 107)

For example, to edit a user’s security permissions, the Administrator would check out the  Access Control database, make the desired changes, then check it back in.

Creating and Adding Projects to Local Manager

There are three ways to get a project into the Local Manager system:



- You can create a new project on the Server. For more information, see “To create a new project” on page 101.
- If there are any existing projects on your system not under Local Manager, you can move them there. In the Navigator, these projects appear under the  My Computer folder. To add them to Local Manager, right-click the project and select **Add to Server**.
- If you have a project exported or backed up, you can restore it directly into Local Manager. For more information, see “To add a previously-created project to Local Manager” on page 101.

Under Local Manager, projects are stored under the  Local Manager folder, itself located under the main  Projects folder. Any new projects you create are located in that folder.

To open a project when using Local Manager

1. In the Manager tab of the Navigator, under the  Projects folder, right-click the project you want to open and select **Open**.

The Projects folder is divided into two folders.

-  My Computer contains Machine Edition projects that have not been added to the Local Manager system.
- The  Local Manager folder contains all projects managed by the Local Manager system. Under this folder, you must have a project checked out to be able to open it.

To create a new project

You must have the proper Access Control permissions to perform this task.

1. In the Manager tab of the Navigator, under the  Projects folder, right-click the  Local Manager folder and select **New**. The New Project dialog box appears.

You cannot create a new Machine Edition project if a project is already open.

2. Type a unique project name (maximum 32 characters).
3. Select a project template.

A sample of the project appears in the Project Template dialog box.

4. Click **OK**.


The project is created on the Server. If you're creating a new Machine Edition project, it is automatically checked out and opened.

Once you create a new project on the Server, it is recommended that you also add groups to the project, to allow users access to it. For more information, see "Access Control" on page 104.

To add a previously-created project to Local Manager

You must have the proper Access Control permissions to perform this task.

- In the Manager tab of the Navigator, under the  My Computer folder, right-click the project you want to add and select **Add**.

The project moves from the My Computer folder to the  Local Manager folder. It is also automatically checked out by you.

Once you create a new project on the Server, it is recommended that you also add groups to the project, to allow users access to it. For more information, see "Access Control" on page 104.

To check out a project or item from the Server

- In the **Manager** tab of the Navigator, right-click the project you want to check out, and select **Check Out**.
To check out a project, the project must not be already checked out by anyone. You must also be logged onto the Local Manager system and have the CIMPPLICITY Manager CheckInOut permission.
Local Manager copies the item to the user's local computer (the "client" computer), and marks the item as being "checked out". As long as the item is checked out, no other user can make changes to that item.

To check in a project or item to the Server

1. In the **Manager** tab of the Navigator, right-click the project you want to check in, and select **Check In**.
The project must be closed and checked out by the current user on the current client computer. You must also be logged on to the Local Manager and have the CheckInOut permission.
2. Select from the available options.
 - **Remove local copy:** Removes the copy of the project from the working directory after the check in is complete.
 - **Keep local copy:** Keeps a copy of the project in the working directory after the check in is complete. If the project is not a Machine Edition project, it is marked as read-only.
 - **Keep checked out:** Updates the version in Local Manager with your changes, but keeps the project checked out.
3. (Optional) Include a comment (maximum 256 characters) of the changes made while the project was checked out.
4. Click **OK**.
Local Manager copies the updated version from the working directory to the Local Manager system.

To view a project history report

- In the **Manager** tab of the Navigator, under the **Local Manager** folder, right-click the project for which you want a history report and select **View History Report**.
A history report appears on the indicated project. You can right-click versions in the project history report for an additional menu of commands.

Want to know more? In the Help Index, look up project history and choose "Project History Reports".

To undo a check out of a project or item

1. In the Manager tab of the Navigator, right-click the project you checked out, and select **Undo Check Out**.




Normally, to undo a check out, the item must be checked out by you on the current computer. You must also be logged on to Local Manager, and have the CheckInOut permission.

If the item is checked out by someone else (or by you on a different computer), you can still undo the check out if you have the OverrideCheckOut permission.

2. If you're undoing a check out of a project, in the **Undo Check Out** dialog box, select one of the following option(s), and then click **OK**.
 - **Remove local copy:** Deletes the working copy of the project.
 - **Replace local copy:** Overwrites the working copy with the version in Local Manager.
 - **Keep checked out:** Keeps the project checked out so you can continue working on it, but cancels the changes you made since you last checked it out. This option is only available with the Replace local copy option.
 - **Leave local copy as is:** Keeps a read-only copy of the project with the changes you made while it was checked out.


ACCESS CONTROL


The  Access Control folder in the Navigator's Manager tab contains these items:

-  **Users** represent individual users. When you log on to a Manager system, you must identify which User you are.
-  **Groups** are groups of Users. Each group is assigned a set of permissions. Users in a group inherit its permissions.
-  **Permissions** are security settings assigned to groups. In general, a permission allows or disallows a particular action or set of actions.

Casually, we say that a user “has” a permission if they’re a member of a group for which that permission is set to True. Most permissions are True/False settings; exceptions are documented in the online help.


Administrators do not assign permissions to individual users. Instead, they assign permissions to groups, then add users to those groups. This lets an administrator assign permissions to a set of users with similar roles and responsibilities.

Under a group node, permissions are organized based on the feature or application they’re associated with. You configure the actual permissions in the Inspector. For example, selecting a group’s  CIMPLICITY Manager permissions node makes permissions that control access to Manager features appear in the Inspector. By setting the appropriate permission in the Inspector, you allow or disallow access to specific Manager features for that group.

- For more details on specific permissions, select its  permissions node in the Navigator and press F1. Or, for a short description, ensure the Companion window is open and select the permission in the Inspector.

Users can be members of more than one group. If this is the case, they get the most advantageous set of permissions for all groups they are a member of.

The Access Control Database

A list of users and groups are stored in the Access Control database. To make any changes to users, groups, or permission settings (including adding new users or groups), you must first check out the  Access Control database. The changes you make don’t take effect until the Access Control database is checked back in to the Local Manager system.

Each computer running Local Manager maintains its own Access Control database. If you need to work on projects from more than one computer, you must be added as a user to all of them.

Access Control of Projects




Access Control for projects depend on which groups are assigned to the project. The users or members of a group that are assigned to a project can check the project in and out, as long as they also have the CIMPLICITY Manager CheckInOut permission (see “To assign a group to a project or layout” on page 106). Since users can only edit a project when it’s checked out of the Server, this lets the administrator control who can make changes to a given project.

System Groups and Users




The following groups and users cannot be deleted:

- **Administrators group:** This group has every available permission and is automatically assigned to every project managed by CIMPLICITY Manager.
- **Everyone group:** All users on a Server are members of the Everyone group, and have minimal permissions. By assigning a permission to the Everyone group, you can assign that permission to every user in the Access Control database.
- **Administrator user:** The Administrator user is a member of the Administrators group. When you start Local Manager for the very first time, you must log in as the Administrator.
- **Guest user:** The Guest user can be assigned to any group and inherits that group’s permissions.




To create a new user

1. In the Manager tab of the Navigator, right-click the  Access Control folder, and select Check Out Access Control.
2. Under the  Access Control folder, right-click the  Users folder and select New User.
3. Type a name for the user.
User names can only use the characters 0-9, a-z, A-Z, and _. Spaces are not allowed in the user name, and the maximum number of characters is 20.
4. Configure the user’s properties in the Inspector.

To create a new group


1. In the Manager tab of the Navigator, right-click the  Access Control folder, and select Check Out Access Control.
2. Under the  Access Control folder, right-click the  Groups folder and select New Group.
3. Type a name for the group.
Group names can only include the characters 0-9, a-z, A-Z, and _. Spaces are not allowed in a group name, and the maximum number of characters is 20.
4. Type a description for the group in its Description property in the Inspector.

To assign permissions to a group

1. In the Manager tab of the Navigator, right-click the  Access Control folder, and select Check Out Access Control.
2. Under the Access Control folder, in the  Permissions section of the  Group folder, right-click the type of permissions you want to set, and select Properties.

Except for custom project permissions, you cannot edit permissions for the Administrator group.

3. Configure the properties in the Inspector window.



For more details on specific permissions, select its  permissions node in the Navigator and press F1. For a short description, ensure the Companion window is open and select the permission in the Inspector.

Want to know more? In the Help Index, look up permissions and choose the appropriate permissions topic.

To add a user to a group

1. In the Manager tab of the Navigator, right-click the  Access Control folder, and select Check Out Access Control.
2. Under the  Groups folder, right-click the  Users folder of an individual  Group and select Add User. Select a user from the list that appears.

- or -

In the  Users folder (directly under Access Control), right-click the  User you want to assign to a group and select Add to Group. Select a group from the list that appears.


To assign a group to a project or layout

To add a group to a project, you must be a member of a group that has the EditProjectList permission. To add a group to a layout, you must have the EditLayouts permission.

1. Check out the project you want to add a group to.
2. In the Manager tab of the Navigator, right-click the project you checked out and select Add Group.
3. Choose one of the groups in the list.
The group is added to the project along with its permissions and users.
4. Check in the project.

AUDIT TRAILS

CIMPLICITY Manager can track and record actions performed by its users. If a user performs an action and the administrator has flagged that type of action to be audited, a record is logged in the audit trail database.

You specify which actions to track by configuring the Server's *audit trail settings*. Located under the  Audit Trail folder in the Navigator, audit trail settings are grouped by feature or product type.

The settings are configured in the Inspector. If an audit trail setting is set to True, Local Manager tracks and records all actions of that type. For example, if the CIMPLICITY Manager audit trail setting PasswordChange is set to True, Local Manager logs a record every time a user's password is changed.

The Audit Trail Database


Audit trail records are added to Local Manager's audit trail database, which is stored in Microsoft Access™ format. You can use Microsoft Access (or another application capable of reading Microsoft Access databases) to view, sort, and generate custom reports on audit trail records. Though not as simple as the Audit Trail Report Manager, this gives you the most flexibility in searching the audit trail database.

The audit trail database file is located on the CIMPLICITY Manager Server computer under "...\\CIMPLICITY Machine Edition\\fxServer\\Audit Database\\Audit.mdb", where "...\\CIMPLICITY Machine Edition" is the Machine Edition installation directory.

Since audit trail records are constantly being logged to the Server, the audit trail database file can get very large. It's a good idea to occasionally archive the database file to another computer to free disk space on the computer.

To edit Audit Trail settings

1. In the Manager tab of the Navigator, right-click the  Audit Trail folder, and select Check Out Audit Trail.
2. Under the  Settings folder, right-click on CIMPLICITY Manager, Scheduler or the project type whose audit trail settings you want to edit, and select Properties.
3. Edit the settings for your selection in the Inspector window.

For a description of a particular audit trail setting, ensure the  Companion window is open, and click the audit trail setting in the Inspector.

To archive the Audit Trail database

1. In Windows Explorer, move the Audit Trail database file (*.mdb) from the Server computer to the computer you want to archive the database to.



The audit trail database file is located on the CIMPLICITY Manager Server computer under "... \CIMPLICITY Machine Edition \fxServer \Audit Database \Audit.mdb", where "... \CIMPLICITY Machine Edition" is the Machine Edition installation directory.

2. Rename the moved file so that the next time you archive the database, you won't overwrite the previously archived file.

Include the current date in the new file name to make it easier to find in the future. Local Manager will create a new audit trail database file the next time an audit trail action is recorded. The file will have the same name, and will be created in the same directory you moved the database from.

You can view records in an archived database by opening the archived file in any database or report application capable of reading Microsoft Access™ databases.

Audit Trail Reports

The easiest way to view records in the audit trail database is through the Audit Trail Report Manager, located under the  Reports folder in the Audit Trail folder. To open the Report Manager, right-click the  Audit Trail Report Manager node and choose **Open**.

Generated Audit Trail Reports can be displayed on screen and, if desired, sent to a printer.

- If you have an application that can read Microsoft Access databases, you can use it to view and sort audit trail records.

To generate an Audit Trail report

1. In the  Reports folder, right-click  Audit Trail Report Manager and select **Open**. The Audit Trail Report Manager dialog box appears.

The very first time you open the Audit Trail Report Manager you will be asked to locate the audit trail database file (*.mdb) on the Server computer.

The audit trail database file is located on the CIMPLICITY Manager Server computer under "... \CIMPLICITY Machine Edition \fxServer \Audit Database \Audit.mdb", where "... \CIMPLICITY Machine Edition" is the Machine Edition installation directory.

2. Select the Report Template you want to create.
3. Configure the following settings for the report.

- Select a **Date** range.
 - Set the **Filters** you want to use.
 - Select how you want the report **sorted** (by date, user, computer name, or project name).
 - Select the **page size**.
4. **(Optional) If you want to print the report, select a printer.**
Select **Default** to print the report on the printer configured as the Windows default.
 5. **Click Print Preview.**
The Report Viewer dialog box displays the report you requested. You can then print or save the report to a file.

A

- Access Control 104
 - for projects 104
 - for Version Control 101
 - how it works 104
- Access Control database 104
- actions 39, 40
- Active scripting 62
- adding
 - Control I/O drivers 50
 - language columns 74
 - logic component 34
 - Machine Edition projects 14
 - OPC server links 69
 - PLC Access drivers 67
 - users to groups 106
- Administrator user 105
- Administrators group 105
- alarm groups 66, 78
- alarm objects 59, 66
- alarms 66, 78
 - QuickPanel 78
- Allen-Bradley RIO 49
- animation 58
 - objects 59
 - touch animation scripts 60
- Application Builder
 - about 91
 - conditional steps 92
 - inserting steps 93
 - See *alsteps*, flowchart
- application scripts 60, 63, 81
- applications, motion 85
- ASCII drivers 49
- assigning groups to projects and layouts 106
- assigning permissions 106
- Audit Trail database 107
- Audit Trail Reports 108
- Audit Trail settings 107
- Audit Trailing 107
 - database. See Audit Trail database
 - how it works 107

- reports. See Audit Trail Reports
- settings. See Audit Trail settings

Audit Trails. See Audit Trailing

authorization. See product authorization

B

- backing up projects 14
- blocks, motion See *so* motion blocks
- branches 40
- Build tab 20

C

- Calculator, Motion See Motion
- Calculator
 - CAM profiles
 - about 95
 - editing 96
 - editor 95
 - CE box. See CE targets
 - CE targets 23, 25
 - downloading 29
 - chart objects 59
 - checking in items 99, 102
 - checking out items 99, 100, 102
 - undoing check outs 103
 - client
 - networking 72
 - OPC 68
 - Companion 12, 40, 44, 47
 - component
 - logic 34
 - conditional steps 92
 - configuration
 - CAM commands 95
 - motion script 86
 - configuring
 - alarm conditions 66
 - Control I/O 51
 - Controller 53
 - hard real-time Controller 54
 - panels 58
 - PLC Access I/O 67

- properties 16
- user preferences 15

Control I/O 50

- tool 50
- working with 51

Controller 52

- configuring 53, 54
- going online 25
- hard real-time 54
- running 54
- scan cycle 52

Controller Functions wizard 90

controller/drive targets, motion 85

correcting, errors 28

creating

- actions 40
- alarm groups 66
- alarm groups (QuickPanel) 79
- alarm objects 66
- alarm objects (QuickPanel) 79
- application scripts 60, 81
- custom web documents 71
- groups 105
- logging groups 65
- macros 36
- panel scripts 60, 81
- panels 58
- projects 14, 26, 57, 101
- SFCs 35
- subroutines 40
- touch animation scripts 61
- users 105
- variables 26

custom web documents 71

customer support 6

D

- Data Watch 41
- debugging scripts 73
- developing. See creating
- DeviceNet 49
- devices
 - motion control 84

devices, adding PLC 67
 downloading 25, 28, 53
 CE targets 29
 motion device targets 28
 NT targets 29
 drawing graphical objects 59
 drawing tools 59
 drive targets, motion 85
 drivers

 adding
 Control I/O 50
 PLC Access I/O 67
 ASCII 49
 Control I/O 49, 50
 Control Peer 49
 OPC Client 68
 PLC Access I/O 67

E

editing
 application scripts 60, 81
 CAM profiles 96
 custom web documents 71
 ladder programs
 offline 40
 online 41
 logging groups 65
 motion scripts 88
 panel scripts 60, 81
 panels 58
 quick editing
 ladder program 41
 SFCs 37
 quick test 31
 SFCs
 offline 36, 44, 45, 48
 online 38
 steps, flowchart 93
 editors
 grid (QuickPanel) 76, 78
 ladder 39
 panel 58
 QuickPanel panel 77
 script 60, 81
 errors and warnings
 correcting 28
 Feedback Zone 20
 Everyone group 105
 external keypad 80

F

Factory Client 2000. *See* CE targets
 Factory Layouts
 assigning groups to 106
 Feedback Zone 20
 fixing, project errors 28
 flowchart editor *See* Application
 Builder
 flowchart steps *See* steps, flowchart

G

GE Fanuc Genius 49
 getting help 11
 getting projects 102
 global function library scripts 60
 graphical objects 58, 59
 graphical panels. *See* Panels
 grid editor
 alarms 78
 keypad assignment 80
 messages 79
 groups 104
 adding users to 106
 assigning to projects and
 layouts 106
 creating 105
 LID (Local Image Display) 76
 system 105
 Guest user 105

H

hard real-time
 Controller 54
 VenturCom RTX 54
 hardware
 motion devices 84
 help 11
 history report 100
 Honeywell SAS 49

I

I/O
 configuring
 Control 51
 PLC Access 67
 Control 50
 OPC 68
 PLC Access 67

IEC addresses 51
 Import tab 20
 index, search 11
 InfoView tab 14
 InfoViewer 12
 inserting
 ladder instructions 40
 motion script items 88
 rungs 40
 script functions 63
 SFC instructions 37
 Inspector 16
 installing Machine Edition 3
 Internationalization. *See* Languages
 folder

K

keypad assignment 80
 keyword search 11

L

labels 39
 Ladder Editor 39
 offline 40
 online 41
 ladder instructions 40
 ladder programs
 defined 39
 editing
 offline 40
 online 41
 opening 39
 quick editing 41
 Language translation 73
 Languages 73
 Languages folder 73
 adding new languages 74
 default language 75
 grid 74
 renaming languages 74
 Languages grid 74
 display font 75
 updating Source language 75
 LID groups 76
 Local Image Display (LID) 80
 Local Image Display groups 76
 Local Manager 97
 Local Manager system
 add projects to 101

- check in to 99
- check out from 99
- Local Message Display (LMD) 79
- local projects
 - adding to Local Manager 101
- locating
 - labels 39, 43, 44, 45, 46, 47, 48
 - variables 36, 40, 44, 47
- logging data 65
- logging groups 65
- logging strategies 65
- Logic Developer
 - PC 33 to ??
- Logic Developer - PC 33 to 56
- logic, adding 34

M

- Machine Edition
 - downloading 28
 - environment 10
 - getting help 11
 - key features 11 to 21
 - Logic Developer - PC ?? to 55
 - OPC servers 69
 - projects 14, 23
 - properties 16
 - toolbars 13
 - validating 28
 - View 57 to 82
- macros 35, 36
- Main Motion Wizard 90
- Manager tab 14
- Manager, Local 97
- message grids 79
- messages 79
 - Local Image Display (LID) 80
 - Local Message Display (LMD) 79
 - Triggered Message Display (TMD) 79
- Messages tab 20
- Modicon Quantum 800 49
- Motion applications
 - configuration script 86
 - programs 86
- motion applications 85
- motion blocks
 - about 87
 - editing with calculator 93
 - opening 85
 - switching editors 86
- Motion Calculator
 - about 93
 - using 94
- motion configuration 86
 - opening 85
- Motion Developer
 - applications 85
 - supported hardware 84
- Motion programs
 - fault-handing program 86
 - opening 85
 - switching editors 86
- motion programs 86
- motion scripts 88
- motion targets 85
- Motion Wizards
 - about 89
 - Application Builder 91
 - Controller Functions 90
 - Main Wizard 90
 - navigating through 89
 - See also Motion Calculator
 - special wizards 91
- moving product authorization 5
- multi-language support. See Languages folder

N

- Navigator 14
 - property column view. See Property column view
- networking 72
- NT computers. See NT targets
- NT targets 23, 25
 - downloading 29

O

- objects. See graphical objects
- offline
 - Ladder Editor 40
 - SFC Editor 36, 44, 45, 48
- online
 - Ladder Editor 41
 - SFC Editor 38
 - to the Controller 25
- online help 11
- OPC 68

- client 68
- server links 68, 69
- servers 69
- opening
 - application scripts 60, 81
 - ladder programs 39
 - macros 35
 - motion blocks, motion programs, and motion configuration 85
 - panel scripts 60, 81
 - panels 58
 - projects 15, 23, 26, 34, 57, 83, 101
 - SFCs 35
- Options tab 15
- options, user preferences 15

P

- Panel Editor 58, 77
 - animation 58, 59
 - graphical objects 58
 - working with 59
- panel scripts 60, 63, 81
- panels 58, 77
 - quick testing 31
- peer driver, control 49
- permissions 104
 - assigning 106
- PLC Access I/O 67
- preview panels. See quick test
- product authorization
 - authorizing Machine Edition 4
 - moving to another PC 5
- profiles, CAM See steps, flowchart
- programs, motion See motion programs
- Project History 100
- Project tab 15
- Projects 99
 - Access Control 104
 - adding to Local Manager 101
 - assigning groups to 106
 - checking in 102
 - checking out 102
 - creating 101
 - getting copies of 102
 - history report 100
 - opening 101

- projects 14
 - backing up 14
 - creating 26
 - downloading 28
 - Motion Developer 83
 - opening 15, 23, 26, 34, 57, 83
 - running 23, 72
 - validating 28
 - with HMI component 57
 - with logic component 34
- properties 16
 - configuring 21
- Property column view 21
- Property Inspector. *See* Inspector

Q

- QPScript language 82
- quick editing
 - ladder program 41
 - SFCs 37
- quick start 8
- quick test, panels 31
- QuickPanel alarm groups 78
- QuickPanel alarm window 78
- QuickPanel Applications 76
- QuickPanel Grid Editor 76, 78
- QuickPanel Panel Editor 76

R

- References tab 20
- remote views 71
- Reports
 - variable 22
- Reports tab 20
- requirements, hardware and software 2
- rungs 39
 - inserting 40
 - write changes 42
- running
 - Controller 54
 - projects 23, 72
 - sample applications 23
 - View Runtime 72

S

- sample applications 23
- scan cycle, Controller 52
- script commands. *See* script functions

- Script Editor 60, 81
 - inserting script functions 63
 - scripts 60, 81
 - working with 63
- script functions 63
- scripting languages
 - Active scripting 62
 - QPScript 82
 - VBScript 61
 - ViewScript 61
- scripts 60, 81
 - application scripts 60, 81
 - debugging 73
 - global function library scripts 60
 - inserting script functions 63
 - motion 88
 - panel scripts 60, 81
 - touch animation scripts 60
- scripts, motion
 - inserting items 88
- search, index 11
- security. *See* Access Control
- Sequential Function Charts. *See* SFCs
- servers
 - networking 72
 - OPC 69
- setting. *See* configuring
- settings
 - audit trail. *See* Audit Trail settings
- SFC Editor
 - offline 36, 44, 45, 48
 - online 38
- SFC instructions, inserting 37
- SFCs
 - editing 36, 44, 45, 48
 - opening 35
 - quick editing 37
- shortcut keys, View Runtime 72
- shunts 40
- Smart Lists 18
- steps, flowchart
 - conditional 92
 - editing 93
 - inserting 93
 - step types 92
- subroutines 40
- support 6
- switching editors, motion blocks and motion programs 86
- System groups and users 105

- system requirements 2

T

- tags. *See* variables
- targets 23, 25
 - motion 85
- technical support 6
- testing
 - panels 31
 - scripts 73
- toolbars
 - drawing tools 59
 - ladder instruction tools 40
 - Machine Edition tools 13
- Toolchest 19, 40
- tools
 - Companion 12
 - Control I/O 50
 - Data Watch 41
 - Feedback Zone 20
 - InfoViewer 12
 - Inspector 16
 - Navigator 14
 - overview 10
 - toolbars 13
 - Toolchest 19, 40
- touch animation scripts 60
- tracking
 - see* Audit Trailing
- translations. *See* Languages folder
- Triggered Message Display (TMD) 79

U

- undoing check outs 103
- Unused Variable report 22
- user preferences 15
- users 104
 - adding to groups 106
 - creating 105
 - system 105

V

- validating 20, 28, 53
- Variable
 - property column view 21
- Variables 21
 - managing 21
 - reports 22
- variables

- creating 26
- locating 36, 40, 44, 47

Variables tab 15, 21

VBScript language 62

VenturCom RTX 54

Version Control 99

- security 101

View 57 to 82

- testing the HMI 31

View Runtime 72

- running 72
- shortcut keys 72

ViewScript language 61

W

Warm Standby 55

- and OPC servers 55
- configuring 56
- configuring OPC server for 56

Web Access 70

Site Index 70

Windows CE targets. See CE targets

Windows NT targets. See NT targets

windows. See Panels

wizard, Controller Functions 90

Wizard, Main 90

wizards, motion See Motion Wizards

writing motion scripts 88

writing scripts 63