CIMPLICITY 11

Networking

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

doc@ge.com

Networking

Chapter 1. CIMPLICITY Server to Viewer Announcements

About CIMPLICITY Server to Viewer Announcements

CIMPLICITY running on one computer can see which projects are running on your network.

As well as supporting Server to Viewer announcements, Servers may also be networked together to allow the display of data from many Servers on a Server.

CIMPLICITY v.6.2 and over provides straightforward methods for sending and receiving project announcements from the Server to the Viewer.

Options for connecting from Viewer to a Server are as follows.

Connect to:	Description
Project	(Recommended) The user controls the data source the Viewer will display.
Node name	Data from the project that started first on the node will display on the Viewer.
IP address	Data from the project that started first at the IP address will display on the Viewer.

Following is an overview for setting up, testing and troubleshooting a Viewer to Server connection.

Step	Description
1 *(page 9)*	Set up the Viewer to Server connection.
2 *(page 10)*	Check that point values display on the Viewer.
3 *(page 14)*	CimView screen special Viewer configuration.
4 *(page 16)*	Troubleshooting checklist.

1. Set up the Viewer to Server Connection

1. Connect the Server and Viewer to the network.
2. Ping the Server from the Viewer to obtain a basic confirmation that the Viewer can access the Server.
3. Install CIMPLICITY Server on the Server.
4. Install CIMPLICITY Viewer on the Viewer.
5. Create and start a project on the Server.
6. Enable Broadcast/Multi-cast on the Server to send project announcements over the network.
    a. Open the project that will be broadcast and/or multi-cast to the Viewers in the CIMPLICITY Workbench.
    b. Click Project>Project Properties on the Workbench menu bar. The Project Properties dialog box opens.
    c. Select the Options tab.
    d. Check one or both of the following.

A	Enable project broadcast
B	Enable project multicast

Note: Confirm that there are no other projects broadcasting on the network with the same name.

2. Check that Point Values Display on the Viewer

A *(page 10)*	Start the Server and the Viewer.
B *(page 11)*	Get a point value from the Server using a command line utility.
C *(page 13)*	Get a point value from the Server through the Point Control Panel.

1. Start the Server and the Viewer

1. (On the Server) make sure a project is running.
    a. Open the project you will test in the CIMPLICITY Workbench.

b. Click the Run button on the Workbench toolbar to start the project.

2. (On the Viewer) start the Viewer.
   a. Click Start on the Windows task bar.
   b. Select (All) Programs>Proficy HMI SCADA - CIMPLICITY version>CIMPLICITY Options .

   The CIMPLICITY Options dialog box opens.

   a. Select the Projects tab.

   The Viewer name should display in the **Computer name** field.

   a. Click Start Viewer.
   • VIEWER displays in the Running projects box.
   • The Start Viewer button is disabled.
   a. Click OK.

   The CIMPLICITY Options dialog box closes.

   a. Get a point value from the Server using a command line utility

   CIMPLICITY provides a basic command line utility that you can use to connect to the Server in a simple straight-forward manner; extraneous factors can be kept at a minimum.

   The utility is PTQ_onchange.exe.

3. (On runtime Viewers only. PTQ_onchange.exe is not installed.)
   a. Copy PTQ_onchange.exe on the Server.

   The file's location on the Server is ...\Proficy CIMPLICITY\exe\

   Where

   ...\CIMPLICITY is the CIMPLICITY root directory.

   a. Paste PTQ_onchange.exe into the following directory on the runtime Viewer.

   ...\CIMPLICITY\exe\

4. Click Start>Run on the Windows task bar>Start menu.

   A Run dialog box opens.

5. Enter Cmd in the **Open** field.

   A Command window opens.

6. Display the C:\> prompt.

7. Get a point value using the Server's IP address as follows.

A	**Contact the Server by IP address.**	
	a. Type ptq_onchange.exe b. Press Enter.	
B	**Name the qualified point that should return a value, using the IP address.**	
	a. Type `\\`<Server IP address>`\`<Point ID>.	
	Where	Is the
	<Server IP address>	IP address of the Server being contacted, which qualifies the point.
	<Point ID>	ID of the point that should return a value.
	a. Press Enter.	
	A CIMPLICITY® Login dialog box opens.	
C	Log in to the project.	
	a. Enter a valid **User ID** and **Password** for the project you are trying to access. b. Click OK to close the CIMPLICITY® Login dialog box.	
	Result: The point value displays.	
D	Press Ctrl+C on the keyboard to disassociate from Point Management.	

If there is a problem accessing the Server or displaying the point value, go to the Troubleshooting checklist *(page 16)*.

8. Type ptq_onchange.exe
9. Press Enter.
10. Type `\\`<Server IP address>`\`<Point ID>.
11. Press Enter.
12. Enter a valid **User ID** and **Password** for the project you are trying to access.
13. Click OK to close the CIMPLICITY® Login dialog box.
14. Re-use the ptq_onchange.exe utility, using the Server's node name and the same point ID.
    a. Type the following at the c:\> prompt.

`ptq_onchange.exe`.

    a. Press Enter.

Type `\\`<Server node name>`\`<Point ID>

    a. Enter your **User ID** and **Password**.

Result: The point value displays.

a. Press Ctrl+C on the keyboard to disassociate from point management.

If there is a problem accessing the Server or displaying the point value, go to the Troubleshooting checklist *(page 16)*.

15. Re-use the ptq_onchange.exe utility, using the Server's project name and the same point ID.
    a. Type the following at the c:\> prompt.

`ptq_onchange.exe`.

a. Press Enter.
b. Type `\\`<Project name>`\`<Point ID>
c. Enter your **User ID** and **Password**.

Result: The point value displays.

a. Press Ctrl+C on the keyboard to disassociate from point management.

If there is a problem accessing the Server or displaying the point value, go to the Troubleshooting checklist *(page 16)*.

a. Get a point value from the Server through the Point Control Panel

16. Click Start on the Windows task bar.

17. Select (All) Programs>Proficy HMI SCADA - CIMPLICITY version>Point Control Panel .

The Point Control Panel opens on the Viewer.

18. Click the Add Points button.

The Select a Point browser opens.

19. Check if you are able to see point values.

A	Select the project name (recommended).
	Note: You can also enter the Server's IP address or node name in the Project field. However, the available points will be in the first project that started on the Server. If more than one project is running on the Server, it may not be the project you want.
B	Click Browse.
C	Select one or more points to display.
D	Click OK.

Result: The values should display in the Point Control Panel.

If there is a problem accessing the Server or displaying the point value, go to the Troubleshooting checklist *(page 16)*.

20. Open a CimView screen on the Viewer.

    Note: Review CimView screen Viewer configuration guidelines *(page 14)* for CimView screen setup on the Viewer.

21. Check if you are able to see values for points configured by the CimView screen after one has been set up.

    Note:

    - If you can see point values in the Point Control Panel, but not on your CimView screen *(page 14)*, the problem could be with your point qualification *(page 16)*.
    - If you can see point values in the Point Control Panel and on a CimView screen, you are connected.
    - If you cannot see point values go through the troubleshooting checklist *(page 16)*.

    If you cannot see point values after troubleshooting, contact CIMPLICITY support.

3. CimView Screen Viewer Configuration Guidelines

3. CimView Screen Viewer Configuration Guidelines

Step	Description
3.1 *(page 14)*	Specify Viewer global parameters.
3.2 *(page 15)*	Set up the CimView screen source for the Viewer.
3.3 *(page 16)*	Set up points to be seen from a Viewer.

3.1 Specify Viewer Global Parameters

- Create the same global parameter values for the Server and Viewers.
- Create different global parameter values for the Server and Viewer.

- `/Keypad` option on a Viewer.

Create the same global parameter values for the Server and Viewers.

1. (On the Server) make sure that you have configured the required global parameters for the CIMPLICITY project.
2. Open the <project name>\master\ directory in Windows Explorer.
3. Copy the following files.
   - glb_parms.dat
   - glb_parms.idx
4. Paste the files in each Viewer's ...\Proficy CIMPLICITY\Data\ directory.

   Create different global parameter values for the Server and Viewer.

5. Create a new dummy project.

   Note: This project will be used only to create global parameters for the Viewer.

6. Configure global parameters with the values that will be used on one or more of the Viewers.
7. Open the <project name>\master\ directory in Windows Explorer.
8. Copy the following files.
   - glb_parms.dat
   - glb_parms.idx
9. Paste the files in the appropriate Viewer's ...\Proficy CIMPLICITY\Data\ directory.
10. Repeat 1 – 5 until you have created, copied and pasted all of the global parameters that must be different for different Viewers.

    /Keypad option on a Viewer

    If you want users to display a keypad on the Viewer the option configure the …\Proficy CIMPLICITY\Data\gefkeypad.cfg file on each Viewer.

3.2. Set up the CimView Screen Source for the Viewer

There are two options for displaying CimView screens on the Viewer.

- Each viewer can have its own copy of the screen files (.cim) in a local directory.
  - Benefit: The screens will load faster.

    - Issue: A screen designer or system administrator will have to make sure that whenever a screen is modified, it will be re-copied to every Viewer it is on.

- Screens are loaded on the Viewer from a shared directory.
    - Benefit: Screen changes are sure to display on each Viewer.
    - Issue: The speed of screen loading is dependent on the speed of network configuration and traffic.

Note:

- Screens are most commonly loaded from a shared directory.
- If you want users to display a keypad on the Viewer, configure a …`\Proficy CIMPLICITY \Data\gefkeypad.cfg` file to use the `/Keypad` option on each Viewer.

3.3. Set up Points to be Seen from a Viewer

Use the following techniques to make sure the viewer can locate the projects for each point:

- Fully qualify all points.
- Fully qualifying all points used on the screen takes some up front planning when the screens are being created. It has the advantage that you can have a single screen viewing points from multiple projects and does not require any special shortcuts to launch the screen. The fully qualified points will use the project indicated in the point ID.
- Use a `/project` command line option to open the screens from a short cut.
- The `/project` command line option will specify the project to use for any unqualified points.

4. Troubleshooting Checklist

4. Troubleshooting Checklist

Step	Description
4.1 *(page 17)*	Double-check the CIMPLICITY license key.
4.2 *(page 17)*	Computer Name/IP Address Resolution.
4.3 *(page 17)*	Multiple NIC's on the Server.

Step	Description
4.4 *(page 18)*	Connections on Port 32000.

4.1. Double-check the CIMPLICITY License Key

1. Make sure that the license hardware key is secure in the Server's and Viewer's parallel or USB port.
2. Click Start on the Windows task bar.
3. Select (All) Programs>Proficy Common>License Viewer.

   The license's Management Console window opens.
4. Make sure that the CIMPLICITY Development or Runtime Server is enabled.

   Note: The status is under Options>System Type.

4.2. Computer Name/IP Address Resolution

If you are trying to connect to the Server by node name, it is possible that your Viewer and Server may not be able to resolve each other’s node names.

(On both the Server and Viewer)

1. Click Start on the Windows task bar.
2. Select (All) Programs>Proficy HMI SCADA - CIMPLICITY version>CIMPLICITY Options .

   The CIMPLICITY Options dialog box opens.
3. Select the Hosts tab.
4. Identify the Host nodes and IP addresses.

4.3. Multiple NIC's on the Server

If you are trying to connect to the Server by IP address and the Server has multiple NIC (network) cards, make sure that you are trying to access an IP address that is being used

- Select the IP addresses to be used.

1. Click Start on the Windows task bar.
2. Select (All) Programs>Proficy HMI SCADA - CIMPLICITY version>CIMPLICITY Options .

   The CIMPLICITY Options dialog box opens.

   If the computer has more than one NIC, a Network tab will display.

3. Select the Network tab.
4. Select/check the IP addresses that should be/are available.
   - Verify that the card for the Viewer's network is the top card in the TCP/IP stack configuration. This will allow better communications over the Viewer-Server network.

   Use Network Connections features in the Windows Control Panel.

   Consult Microsoft documentation for details.

   Note: If both NIC cards are in what would normally be considered a class A (001.y.z.w to 126.y.z.w) or Class B network (128.y.z.w to 191.y.z.w) it is possible that the network class will supersede the subnet mask. In this case, It may be advisable to try a 10.y.z.w network on the internal, non-exposed, NIC card.

4.4. Connections on Port 32000

Port 32000 is the port that CIMPLICITY communicates through.

(On the Server)

1. Open the Command window.
2. Type `netstat –a` at the command prompt.
3. Note all connections on Port 32000.

   You should see a:

   - TCP connection to your Viewer on Port 32000 and Viewer's IP address
   - UDP connection on Port 32000 and `*:*` for the IP address.

   If the result is different do the following.

   If you do not see connections on Port 32000 or only on UDP port 32000

Make sure the

- Project is running on the server and the
- Viewer is running as a Viewer.

If you do not see connections on both UDP port 32000 and TCP port 32000

Try to un-install and reinstall the TCP/IP protocol. Consult Microsoft documentation.

(On the Viewer)

4. Open the Command window.
5. Type `netstat –a` at the command prompt.
6. Note all connections on Port 32000.

   You should see a:

   - TCP connection to your Server on Port 32000 and the Server's IP address.
   - UDP connection on Port 32000 and `*:*` for the IP address.

   If the result is different do the following/

   If you do not see connections or connections only on TCP

7. Recheck the license registration on the Server and Viewer.
8. Try to un-install and reinstall the TCP/IP protocol on the Viewer. Consult your Microsoft documentation.
   - Router listens on TCP ports 32000, 32256, 32512 and 32768 for incoming connections. Not configurable.
   - Router listens on UDP port 32000 for project announcements. Not configurable.
   - RtrPing uses TCP port 4000 by default. Configurable using REDUND_PROBE_PORT global parm. This is for server redundancy.
   - Cabling redundancy uses ports 5000 - 6000. Configurable in CIMHOSTS.TXT file.

Chapter 2. Remote Project Configuration

About Remote Project Configuration

If, when a project starts, the Point Bridge or Point Data Logger need to get points from projects on other computers running CIMPLICITY projects, you need to define remote projects.

If your computer is on a network with other CIMPLICITY computers, you can retrieve point information from projects running on the other computers in a variety of ways.

You can:

- Display CimView screens for other projects.
- Display points from other computers on CimView screens running in your project.
- Collect data from points in projects on other computers.
- Log points from projects on other computers if you are using the Database Logger option.

Remote Projects must be defined if Point Bridge or the Point Data Logger need to get point values from projects on other computers that run CIMPLICITY projects.

1. Configure a remote project.
2. Configure Point Bridge Points.
3. Configure Advanced Point Bridge and Enterprise Server Information.

1. Configure a Remote Project

1. Configure a Remote Project

Note: The Remote Registry Service must be running in order for remote node features, for example, Server Redundancy, start/stop projects CIMPLICITY Options System Sentry, Database Logger and Login Panel to work properly.

If you are configuring a remote project on Vista, you will need to:

On Vista you will need to

- Set the Remote Registry Service to automatic start mode. It is not automatic by default.
- Make sure that the following are added to the firewall:
- CIMPLICITY process

- File/printer sharing.

Step	Description
Step 1 *(page 21)*	Open a Remote project dialog box.
Step 2 *(page 23)*	Enter general specifications for a remote project.
Step 3 *(page 23)*	Enter Enterprise Specifications for a remote project.

Step 1. Open a Remote Project Dialog Box

Step 1. Open a Remote Project Dialog Box

Step	Description
Option 1.1 *(page 21)*	Create a new remote project.
Option 1.2 *(page 22)*	Open an existing Remote Project dialog box.

Option 1.1. Create a new Remote Project

1. Select **Project>Security>Advanced>Remote Projects** in the Workbench left pane.
2. Do one of the following.

A	Click File>New>Object on the Workbench menu bar.	
B	Click the New Object button on the Workbench toolbar.	
C	In the Workbench left pane:	
	Either	Or
	Double click **Remote Projects**.	a. Right-click **Remote Projects**. b. Select New on the Popup menu.
D	a. In the Workbench right pane. a. Right-click any remote project. b. Select New on the Popup menu.	
E	Press Ctrl+N on the keyboard.	

The New Project dialog box opens when you use any method.

3. Right-click **Remote Projects**.
4. Select New on the Popup menu.
5. Right-click any remote project.
6. Select New on the Popup menu.
7. Enter a name for the project in the **Project name** field.
8. Click OK.

A Remote Project dialog box opens for the new project.

Option 1.2. Open an existing Remote Project Dialog Box

1. Select **Project>Security>Advanced>Remote Projects** in the Workbench left pane.
2. Select a remote project in the Workbench right pane.
3. Do one of the following.

<table>
<tr><td>A</td><td colspan="2">Click Edit>Properties on the Workbench menu bar.</td></tr>
<tr><td>B</td><td colspan="2">Click the Properties button on the Workbench toolbar.</td></tr>
<tr><td>C</td><td colspan="2">In the Workbench left pane:
a. Right-click Remote Projects.
b. Select Properties on the Popup menu.</td></tr>
<tr><td>D</td><td colspan="2">In the Workbench right pane:</td></tr>
<tr><td></td><td>Either</td><td>Or</td></tr>
<tr><td></td><td>Double-click a remote project.</td><td>a. Right-click a remote project.
b. Select Properties on the Popup menu.</td></tr>
<tr><td>E</td><td colspan="2">Press Alt+Enter on the keyboard.</td></tr>
</table>

4. Right-click **Remote Projects**.
5. Select Properties on the Popup menu.
6. Right-click a remote project.
7. Select Properties on the Popup menu.

Step 2. Enter General Specifications for a Remote Project

rect 24, 55, 276, 89 *(page 23)*
rect 21, 94, 280, 118 *(page 23)*
rect 24, 121, 275, 147 *(page 23)*
rect 23, 148, 84, 171 *(page 23)*
rect 24, 170, 171, 191 *(page 23)*

Options on the General tab in the Remote Project dialog box are as follows.

Option	Description
User ID	Enter the CIMPLICITY User ID that will be accepted for the remote project login.
Password	Enter a password, if you want to require one, for the remote project login.
Confirm Password	If you entered a password, re-enter it here to confirm it.
Enable	• Check to enable the login. • Clear so login will not be made.
Resident Process Use Only	• Check so resident processes will automatically log in to remote projects. Users will still have to log in at the Application level. • Clear so users will not have to log in at the Application level, and they are automatically given the same privileges as the CIMPLICITY User ID for the remote login.

Step 3. Enter Enterprise Specifications for a Remote Project

If you want to use your current project as an Enterprise Server, you must define a remote project for each project in your enterprise from where you want to concentrate data or alarm information.

The Resource and Device are pre-configured on the Enterprise tab in the Remote Project dialog box.

Users who want to view point and alarm information from a remote project on an Enterprise Server must have the remote projects Resource configured in their view.

Check boxes are as follows.

Check Box	Description (when checked)

Collect points	Collects point information from the provider project. All points on the remote project that have been configured as Enterprise Points are available to the current project. Points from remote projects are identified by Remote ID and Point ID as **<** remote_id **>\<** point_id **>** for CimView and Point Control Panel windows.
Collect alarms	Collects alarm information from the provider project.

Note:

- Only one level of concentration is supported. In other words, if you are connecting to a remote project that has local and concentrated points, you will only be able to collect local points from the remote project.
- When the Enterprise Server project starts:

1. The Enterprise project synchronizes with the source projects>
2. ES_.* files are created, which is cache information.
3. The ES_.* files are updated based on Enterprise point and alarm data received from the source projects.

2. Configure Point Bridge Points

Point Bridge, which is part of the Base System, enables separate CIMPLICITY systems to exchange point data.

The systems are referred to as:

System Type	Description
Source	Collects the point data.
Destination	Runs the Point Bridge process

Review:

- When to use Point Bridge.
- Configuration Required for Point Bridge.
- Guidelines for Point Bridge setup.
- Point Bridge Point Configuration Specifications.

When to use Point Bridge

Configure points that use the Point Bridge when you:

- Have a complex system architecture where users on Viewers need to display points from Servers.

- Want to generate alarms on your Server for points on another Server.

Up *(page 24)*

ConfigurationRequired for Point Bridge

1. Configure a Remote Project for the source system on the destination system.
2. Configure the Point Bridge port.
3. Define a device for the Point Bridge port.

   Important: Make sure the Device Name matches the Remote Project name.

4. Define the points you want to retrieve from the source system.

   Important: Make sure that the Point Address matches the source project's point name.

   Guidelines for Point Bridge Setup

   - The Point ID on the destination system does not have to match the Point ID on the source system.

   The data types and number of elements of the two points have to match.

   When the source point changes value, the point value is updated on the local system.

   - The local point and the associated remote (source) point must have identical:
   - Point Class,
   - Data Type, and
   - Number of Elements.
   - Reading and writing points are supported.

   If a point is configured for WRITE access on both systems, a user on the Point Bridge system can set the value of the Point Bridge point.

   The point on the source system is then updated.

   - Dynamic configuration of Point Bridge points is also supported.
   - Points configured incorrectly will remain in an unavailable state.
   - The Point Bridge Device Communications process will generate error messages regarding configuration errors. These can be found in the project's Status Log and the PB.ERR file.

   Up *(page 24)*

   Point Bridge Point Configuration Specifications

   Point Bridge Points require the following entries in the device Point Properties dialog box on the destination server.

Create a device point on the destination server.

Note: The name does not need to match the Point ID on the source system, but it can if desired.

The Point Properties dialog box General and Device tabs include entry decisions that need to address Point Bridge requirements.

- Point Bridge Point Properties: General Tab
- Point Bridge Point Properties: Device Tab

Point Bridge Point Properties: General Tab

Select the General tab in the Point Properties dialog box.

The following entry decisions need to take Point Bridge into consideration

	Item	Description	
1	Read only	Specifies if the Point Bridge process can set the value of a point on a source system.	
		Checked	The point can only be read on the destination server.
		Clear	The point is read/write on the destination server. • A user on the Point Bridge system can set the value of the Point Bridge point. • The point on the source system is then updated. **Important:** Write access is only valid if it is configured for the point on the Point Bridge system and on the source system.
2	Change approval	If change approval is required to set points, the change approval configuration, including valid user names/passwords must be the same on both the source and destination machine. If the configuration is not the same Point Bridge will treat the point as an invalid configuration and log an error.	

Point Bridge Point Properties: Device Tab

Select the Device tab in the Point Properties dialog box.

Point Bridge specifications are as follows.

	Option	Description
1	Device ID	Node name of the source system where the Point Translation process is running.
2	Address	Point ID of the point on the source system.
3	Update Criteria	Only Unsolicited updates are supported.
4	Diagnostic Data	Diagnostic points are not available for the Point Bridge. Clear the Diagnostic Data checkbox.
5	Poll After Set	Poll After Set is not supported by Point Bridge. Clear the Poll After Set check box.

3. Advanced Point Bridge Enterprise Server Configuration

Point bridge and Enterprise Server optionally support data coercion. By default, the point types must be the same on the source and destination.

Data coercion between points of the following types is supported provided that the data value is within the range of both the source and destination:

- SINT
- INT
- DINT
- QINT
- USINT
- UINT
- UDINT
- UQINT
- REAL
- BOOL

For **Point Bridge**, the Global name is <PORT>_ALLOW_COERCE

A value of Y or T enables the feature. The default value is N

For **Enterprise Server**, the name is ES_ALLOW_COERCE

A value of Y or T enables the feature. The default value is N

Chapter 3. Remote Access

About Remote Access

Microsoft Windows Remote Access Service in Windows 2003 and Windows XP can enable CIMPLICITY Viewer software users to use modems for accessing CIMPLICITY project data on a Server computer.

Note: If the IP addresses used by the RAS server are for a different network than the one for the server on which the CIMPLICITY project is running, you may need to establish routes from the CIMPLICITY project computer to the RAS link network. You can use Windows RAS tools to do this. Consult Microsoft Windows documentation for details.

CIMPLICITY Remote Access Server Setup

1. Install the Remote Access Service (RAS) on the CIMPLICITY Server.

   Consult Microsoft Windows documentation for details.

2. Install RAS on the CIMPLICITY Viewer (client).

   Consult Microsoft Windows documentation for details.

3. Broadcast each project that should be listed by the CIMPLICITY Viewers.
   a. Open each project in the CIMPLICITY Workbench.
   b. Click Project>Properties on the Workbench menu bar.

   The Project Properties dialog box opens.

   a. Select the Options tab.
   b. Check Enable project broadcast.

4. Configure the RAS Server to accept connections.
   a. Open the CIMPLICITY Options dialog box, as follows.
   a. Click Start on the Windows task bar.
   b. Select (All) Programs>Proficy - HMI SCADA - CIMPLICITY version>CIMPLICITY Options.

   The CIMPLICITY Options dialog box opens.

   a. Select the Startup Options tab.

b. Check Accept connections.

5. Identify the Server IP addresses, as follows.
   a. Select the Hosts tab in the CIMPLICITY Options dialog box.
   b. Enter the appropriate IP address On the Hosts tab in the CIMPLICITY Options dialog box.

If the RAS Server:	Enter:
Has one or more Network Interface Cards (NIC's).	The IP address configured for each of the NIC's.
Does not have a NIC	127.0.0.1

Remote Access Client Issues

- Behavior based on Use default gateway on remote network selections.
- Use default gateway on remote network example.

Behavior based on Use default gateway on remote network selections

If the RAS client is connected to a local network with a NIC and you are using RAS, the destination host is located using the following process:

If the Use default gateway on remote network in the TCP/IP options of the RAS PhoneBook entry used to dial the remote network:

Is enabled, and the:

Destination Address is:	The packets are sent via the:
On the same subnet as the RAS Server's NIC	NIC.
Not on the same subnet as the RAS Server's NIC	RAS connection to the default gateway assigned by the RAS server.

Is not enabled, and the:

Destination Address is:	The packets are sent via the:
On the same subnet as the RAS Server's assigned RAS IP address	RAS connection.
Not on the same subnet as the RAS server's assigned RAS IP address	RAS server's NIC.

Use default gateway on remote network example

- Your LAN is divided into two Class C subnets, 1.1.1.x and 1.1.2.x,

- You configure your computer on the 1.1.1.x subnet to use RAS with the Use default gateway on remote network option enabled.

All packets that you send to the 1.1.2.x subnet will be sent via the RAS connection...not your computer's NIC.

To route packets correctly:

- Use the Windows route.exe command to add a static route to your TCP/IP route table
- The static route should instruct Windows to send packets intended for the 1.1.2.x subnet to a router on the 1.1.1.x subnet.

Consult Microsoft Windows documentation for details about the route.exe command.

CIMPLICITY RAS Support Limitations

- It will take up to 10 seconds for a list of CIMPLICITY projects to be received by the RAS Client.
- You will not be able to use a computer running CIMPLICITY Computer Cabling Redundancy as the RAS server.
- Connection speeds below 4800 Baud are not supported.
- You can have only one RAS connection at a time from a RAS Client to a RAS Server.

Note: A RAS server can handle multiple RAS clients.

Chapter 4. Proficy WebSpace Integration with CIMPLICITY

About Proficy WebSpace Integration with CIMPLICITY

Proficy WebSpace is a server-based, thin client solution that is optimized for reliable, secure, scalable remote CimView screen monitoring and interaction on 32-bit and 64-bit Windows applications.

- WebSpace/CIMPLICITY Integration: Overview
- WebSpace/CIMPLICITY Integration: Configuration

WebSpace/CIMPLICITY Integration: Overview

Note: (For clients who have upgraded from previous CIMPLICITY versions) WebSpace replaces GlobalView, providing the benefit that the same application can now be used with iFix, Plant Applications and CIMPLICITY.

- There is no loss in functionality that was available in GlobalView.
- If you have a GlobalView license and are experiencing issues with WebSpace licensing, contact your GE representative for information about converting your GlobalView license into a WebSpace license.#

WebSpace/CIMPLICITY Integration: Configuration

CIMPLICITY:

- CimView screens and CimLayout screens can be converted into HTML files and worked with through Proficy WebSpace.
- Integration with WebSpace simply includes starting/stopping the WebSpace server through CIMPLICITY and creating the CIMPLICITY HTML files.

All other configuration is configuration for the WebSpace application.

Step	Description
1 *(page 32)*	WebSpace: CIMPLICITY Configuration Location

Step	Description
2 *(page 32)*	WebSpace: Start Server through CIMPLICITY
3 *(page 33)*	WebSpace: Open Thin Client Server Admin window through CIMPLICITY
4 *(page 33)*	WebSpace: Create a CimView Web Page
5 *(page 38)*	WebSpace: Create a CimLayout Web Page
6 *(page 42)*	WebSpace: Disallow File Open in CimView

1. WebSpace: CIMPLICITY Configuration Location

The CIMPLICITY Options window includes a WebSpace section when WebSpace is available.

1. Open the CIMPLICITY Options dialog box.
2. Select the WebSpace tab.

CIMPLICITY screens can now be converted to use with WebSpace.

2. WebSpace: Start Server through CIMPLICITY

1. The Proficy WebSpace Application Publishing Service is a Windows service that:
   - Listens on a socket for connection requests.
   - Responds by connecting a request to a set of executable files that are running for a single session.
2. Executable files manage launching other applications in the session.
3. The WebSpace server monitors the executable files in the session

   Note: If the files crash, an administrator can terminate the session.

   You can do either of the following to start the WebSpace server.

   - Start the Proficy WebSpace Application Publishing Service through CIMPLICITY

- Start the Proficy WebSpace Application Publishing Service through the Services Window

Start the Proficy WebSpace Application Publishing Service through CIMPLICITY

Check or clear Start Proficy WebSpace server at boot time *(page 32)* to do either of the following.

Check	Starts the Proficy WebSpace Application Publishing Service automatically at boot time.
Clear	Does not start the Proficy WebSpace Application Publishing Service at boot time.

The WebSpace server will start each time the server reboots.

Note: Proficy WebSpace requires the Microsoft Internet Information Server (IIS). However, IIS must be started separately.

Refer to Proficy WebSpace and Microsoft Windows documentation for more details.

Start the Proficy WebSpace Application Publishing Service through the Services Window

The Proficy WebSpace Application Publishing Service can be configured to start automatically, automatically (delayed start), manually or be disabled in the Microsoft Services window like any other Microsoft service.

3. WebSpace: Open Proficy WebSpace Administration Window through CIMPLICITY

The Proficy WebSpace Administration window provides the tools to review and manage WebSpace sessions.

Click the Proficy WebSpace Admin *(page 32)* button to open the Proficy WebSpace Administration window.

Consult the WebSpace documentation that is available in the Thin Client Server Administration window for details about the Proficy WebSpace Administration and client configuration.

4. WebSpace: Create a CimView Web Page

4. WebSpace: Create a CimView Web Page

If you plan to have users logon to WebSpace through a specified URL, WebSpace simply requires that you create an entry point Web page. During the WebSpace session, the CimView Open and Overlay screens can launch other screens.

A CimView screen can be converted quickly into a Web page through the CIMPLICITY Options dialog box.

Note: Web pages that are created in WebSpace's Create Web Page dialog box use WebSpace.

- Open the Create Web Page dialog box.
- Create a Web page.

Open the Create Web Page Dialog Box

Click Create Web Page *(page 32)* on the Proficy WebSpace tab in the CIMPLICITY Options dialog box.

Result: The Create Web Page dialog box opens.

Create a Web Page

The Create a Web Page dialog box provides the following options to create a CIMPLICITY WebSpace HTML: file.

rect 0, 19, 23, 43 4.1. Web Page: Screen Selection *(page 34)*
rect -1, 54, 22, 78 4.2. Web Page: Screen Options *(page 36)*
rect 0, 196, 23, 220 4.3. Web Page: Update Rates *(page 36)*
rect 0, 260, 23, 284 4.4. Web Page: Printer Options *(page 37)*

Section	Description
4.1 *(page 34)*	WebSpace Web Page: Screen Selected
4.2 *(page 36)*	WebSpace Web Page: Screen Options
4.3 *(page 36)*	WebSpace Web Page: Update Rates
4.4 *(page 37)*	WebSpace Web Page: Printer Options

Note: Selections (except for Variables) in the Create Web Page dialog box that are made for one Web page become the default selections for the next Web page.

4.1. Web Page: Screen Selection

The name assigned to the HTML file that is created from the selected CimView screen will be included in the Logon URL.

rect -2, 28, 21, 48 *(page 35)*
rect -2, 57, 21, 77 *(page 35)*

A *(page 35)*	CimView screen
B *(page 35)*	Web page file

A	CimView screen

The selected CimView screen (*.cim or *.cimrt) will display when a user logs onto the Web site.

Do one of the following.

- Enter the path and screen that you select..
- Click Browse Screen to find and select the screen.

B	Web page file

WebSpace `HTML` files must be in the following path and folder..

C:\Program Files\Proficy\Proficy WebSpace\Web\<filename>.html

Where

- C:\Program Files\Proficy\Proficy WebSpace\Web is the location for WebSpace HTML files.
- <filename>.html is the name of the HTML file that will be created from the selected CimEdit/ CimView screen.

guide: Guideline

The full path and HTML filename can be entered easily, as follows.

1. Click Browse Page to select the directory for the file location.

   A Find Web Page browser opens.

2. Do the following.

A	Make sure the selected path is C:\Program Files\Proficy\Proficy WebSpace\Web\
B	Enter the `HTML` file name in the **File name** field.

C	Click Save.

the Find Web Page browser closes; the complete path and HTML filename are entered in the Web Page file field.

4.2. Web Page: Screen Options

1. Click Add.
   The cursor goes to the first or next line.
2. Enter the following.
3. Select a line (with our without a variable).
4. Click Delete.

4.3. Web Page: Update Rates

Wait n seconds and then update the screen every `n` seconds represent command line arguments.

Enter values and CIMPLICITY will enter the complete argument in the HTML file.

Enter the following.

Field		Command Line Argument/Description
Time without input 'til quiescent update rate.		/WaitUpdate <seconds>
		WebSpace will continue to send updates to the client at the normal rate for the number of seconds entered until the user stops using the keyboard and/or mouse.
Time between updates when quiescent.		/MinUpdate <seconds>
		When the number of seconds to wait before the user stops using the keyboard and/or mouse have been reached, WebSpace will begin the **Time between updates when quiescent** update rate and wait the number of seconds entered in the field between each update.
Time between updates during user input.		/MaxUpdate <seconds>
		Time between updates while the user is using the mouse and/or keyboard.

Important: Updates to clients require a trade-off between screen size/update frequency and server CPU usage; the more data a screen needs to transmit and more frequent the updates, the more

CPU will be required. These factors in relation to the amount of RAM installed on the server, directly affect how many clients can connect at the same time.

A way to optimize update rates and the number of clients that can connect is to

1. Determine which screens need fast updates for some users and which can update more slowly.
2. Create separate sessions to address the separate needs.
   a. Send screens that require faster updates to URL's that are different from screens requiring fewer updates.
   b. Enter different values for each screen update session; the values should address the update needs of that session only.

A Relay server can be very useful to help balance the load.

4.4. Web Page: Printer Options

The WebSpace Object options represent the parameters of the WebSpace ActiveX object browser plugin.

rect -3, 275, 23, 301 *(page 37)*
rect 5, 299, 31, 319 *(page 38)*
rect 160, 275, 186, 301 *(page 38)*

Step	Description
A *(page 37)*	Compression.
B *(page 38)*	Close browser window on exit.
C *(page 38)*	Autoconfigure printers.

A: Compression

Check to enable compression.

Default: Checked.

Compression compresses all images to a maximum of 256 colors per image.

Note: When compressed is:

- **Enabled:** Complex images may lose sharpness.
- **Disabled:** Image compression will likely result in a significant increase in bandwidth sent from the Thin Client Server.

B: Close browser window on exit

Close browser window on exit

State	Description
Checked	Closing the Program Window closes the associated browser window and ends the user's WebSpace session.
Clear	Closing the Program Window does not close the associated browser window and does not end the user's WebSpace session

C: Autoconfigure Printers

WebSpace can provide transparent access to client-side printers.

Check the option that should be allowed in your system.

Option	Description
Default printer only	(Default) Only attempt to automatically configure the default printer at startup, enabling clients to print a WebSpace screen on the default printer only.
None	Prohibits printing on the client side.
All	Attempt to automatically configure all client printers at startup, enabling clients to select and print on any available printer.

Note: If the print option is changed after a client session, it may be necessary to delete a print.ini file that was created when CimView printing used during a WebSpace session.

The file is generally located in the c:\documents and settings\<user name>\Local Settings \Application Data directory.

The next time CimView printing is used during a WebSpace session on the client machine, a new print.ini file will be created with the new settings.

5. WebSpace: Create a CimLayout Web Page

5. WebSpace: Create a CimLayout Web Page

- Open the Create CimLayout Web Page dialog box.
- Create a CimLayout Web page..

Open the Create Web Page Dialog Box

Click Create CimLayout Web Page *(page 32)*.

Result: The Create CimLayout Web Page dialog box opens.

Create a CimLayout Web Page

The Create CimLayout Web Page dialog box provides the following options to create a CIMPLICITY CimLayout Web page .

rect 0, 37, 22, 66 5.1. CimLayout Web Page: CimLayout Configuration Selected *(page 39)*
rect 0, 119, 22, 148 5.2. CimLayout Web Page: Printer Options *(page 40)*

Section	Description
5.1 *(page 39)*	CimLayout Web page: Screen selected.
5.2 *(page 40)*	CimLayout Web page: Printer options.

5.1. CimLayout Web Page: CimLayout Configuration Selected

The name assigned to the HTML file that is created from the selected CimLayout file will be included in the Logon URL.

rect -1, 30, 25, 55 *(page 39)*
rect 1, 65, 27, 90 *(page 40)*

A *(page 39)*	CimLayout screen
B *(page 40)*	Web page file

A	CimLayout file

The selected CimLayout configuration will display when a user logs onto the Web site.

Do one of the following.

- Enter the path and the CimLayout file that you select.
- Click Browse layout file to find and select the CimLayout file.

B	Web page file

WebSpace `HTML` files must be in the following path and folder..

C:\Program Files\Proficy\Proficy WebSpace\Web\<filename>.html

Where

- C:\Program Files\Proficy\Proficy WebSpace\Web is the location for WebSpace HTML files.
- <filename>.html is the name of the .html file that will be created from the selected CimEdit/ CimView screen.

guide: Guideline

The full path and HTML filename can be entered easily, as follows.

1. Click Browse Page to select the directory for the file location.

   A Find Web Page browser opens.

2. Do the following.

A	Make sure the selected path is C:\Program Files\Proficy\Proficy WebSpace\Web\
B	Enter the `HTML` file name in the **File name** field.
C	Click Save.

the Find Web Page browser closes; the complete path and HTML filename are entered in the Web Page file field.

5.2. CimLayout Web Page: Printer Options

The CimLayout Object options represent the parameters of the WebSpace ActiveX object browser plugin.

rect -3, 104, 25, 133 *(page 41)*
rect 6, 148, 34, 177 *(page 41)*
rect 173, 105, 201, 134 *(page 41)*

Step	Description
A *(page 41)*	Compression.
B *(page 41)*	Close browser window on exit.
C *(page 41)*	Autoconfigure printers.

A: Compression

Check to enable compression.

Default: Checked.

Compression compresses all images to a maximum of 256 colors per image.

Note: When compressed is:

- **Enabled:** Complex images may lose sharpness.
- **Disabled:** Image compression will likely result in a significant increase in bandwidth sent from the Thin Client Server.

B: Close browser window on exit

Close browser window on exit

State	Description
Checked	Closing the Program Window closes the associated browser window and ends the user's WebSpace session.
Clear	Closing the Program Window does not close the associated browser window and does not end the user's WebSpace session

C: Autoconfigure Printers

WebSpace can provide transparent access to client-side printers.

Check the option that should be allowed in your system.

Option	Description
Default printer only	(Default) Only attempt to automatically configure the default printer at startup, enabling clients to print a WebSpace screen on the default printer only.

Option	Description
None	Prohibits printing on the client side.
All	Attempt to automatically configure all client printers at startup, enabling clients to select and print on any available printer.

Note: If the print option is changed after a client session, it may be necessary to delete a print.ini file that was created when CimView printing used during a WebSpace session.

The file is generally located in the `c:\documents and settings\<user name>\Local Settings\Application Data` directory.

The next time CimView printing is used during a WebSpace session on the client machine, a new print.ini file will be created with the new settings.

6. WebSpace: Disallow File Open in CimView

Disallow file open in CimView provides the ability to prohibit WebSpace clients from opening CimView or CimLayout screens.

At times, this might be necessary, e.g. for security reasons.

Checking or clearing Disallow file open in CimView does the following.

Checked	WebSpace Clients:
Yes	Cannot open CimView or CimLayout screens.
No	Can open CimView or CimLayout screens.

Important: To ensure normal WebSpace functionality Disallow file open in CimView should be clear.

Default: Checked.

Chapter 5. Document Delivery

About Document Delivery

The Document Delivery system can publish data gathered from the various projects and deliver user specified information in user specified format to remote systems. The information is transferred to the remote systems using FTP, HTTP or Network files.

Important: Your system's network and operating system must support at least one of the following:

- FTP
- HTTP
- Mapped network local disk drives.

Note: Document Delivery objects support dynamic configuration. For the Document Delivery resident process, changes to the Document Delivery Base object are allowed, as well as changes to the delivery objects themselves. New delivery objects can be created dynamically. Once a delivery object has been created, the type cannot be changed.

Document Delivery Data Flow Overview

rect 0, 22, 212, 97 *(page)*
rect 210, 20, 401, 95 *(page)*
rect 242, 93, 405, 329 *(page)*
rect 11, 236, 245, 321 *(page)*
rect 19, 323, 112, 452 *(page)*
rect 20, 452, 113, 644 *(page)*
rect 113, 402, 301, 475 *(page)*
rect 207, 475, 330, 533 *(page)*
rect 346, 435, 582, 536 *(page)*
rect 426, 240, 511, 434 *(page)*
rect 511, 238, 596, 437 *(page)*

1. A file triggers a CIMPLICITY script that creates an output document with a header from the file.
2. The output document is sent to a temporary directory.
3. A CIMPLICITY script copies the output document from the temporary to the delivery directory. *(page 49)*

4. A directory thread adds the new document to a preliminary output list.
5. A control thread checks each file in the preliminary output list to see if it has the required header.
    a. If the file does not have the required header, an audit message is generated. The file is then moved to the flush directory.
    b. When an output document is found that does have a valid header, a file without the header is copied into the work directory.
6. The control function sends the filename to an Output Document list.
7. The document is delivered to a remote system.

Note: Details about delivery to the remote system are based on the existing conditions static to the delivery method *(page 45)*.

Delivery Operation

Review the operation for the:

Step	Description
1 *(page 44)*	Output document header evaluation (performed before delivery).
2 *(page 45)*	Delivery process.
3 *(page 48)*	Delivery conclusion.

1. Output document header evaluation. A delivery object has a:

1. Watcher thread that is dedicated to watching its directory using completion ports.

   When a new file is added to the directory, this thread

    a. Adds the name of the file to the preliminary output document list.
    b. Goes back to monitoring the directory.

2. Control thread that checks each file in the preliminary list (if there are files) to make sure that it has the required header.

   If the file:

    • Does have a valid header

Document Delivery:

a. Copies the file without the header into the work directory.
b. Makes a new entry for the output document list, which contains the name of the:
   • File.
   • Destination file.
a. Places the entry at the end of the list, to ensure that the output documents are processed in sequence.
   • Does not have the required header

Document Delivery:

a. Generates an audit message.
b. Moves the file to the flush directory.
a. Delivery process
   • Network or FTP
   • HTTP

Network or FTP delivery

Network or FTP delivery attempts to deliver the first file in the output document list. The method that is used is defined in the delivery object configuration.

Options are:

- Overwrite
- Fail On Existing
- Append

Overwrite

Document Delivery delivers the file unconditionally.

Destination file does exist

Document Delivery

3. Delivers the output file to the remote location with the following name.

   `DDTEMP_`<destination file name>

   Where

   `DDTEMP_` denotes a temporary file.

   <destination file name> is the name the file will be when it reaches its destination.

4. Deletes the existing file, if a file exists.

5. Changes the output file name with the **DDTEMP_** prefix to the original destination filename.

   Destination file does not exist

6. Delivers the output file to the remote location with the following name.

   `DDTEMP_`<destination file name>

   Where

   `DDTEMP_` denotes a temporary file.

   <destination file name> is the name the file will be when it reaches its destination.

7. Changes the output file name with the **DDTEMP_** prefix to the original destination filename.

   Fail On Existing

   Checks the remote location to see if the destination file already exists.

   Destination file does exist

   Document Delivery fails the delivery.

   Destination file does not exist

   Document Delivery:

8. Delivers the output file to the remote location with the following name.

   `DDTEMP_`<destination file name>

   Where

   `DDTEMP_` denotes a temporary file.

9. Changes the output file name with the **DDTEMP_** prefix to the original destination filename.

   Up *(page 45)*

   Append

   ⚠ **Important:** Append only applies to Mapped Network and FTP delivery.

   The following happens if the destination file:

   Destination file does exist

Document Delivery:

10. Copies the remote file to the working directory with the following name.

    `DDTEMP_`<destination file name>

    Where

    `DDTEMP_` denotes a temporary file.

11. Appends the append file with the body of the output document.

12. Delivers the append file to the remote location with the following name.

    `DDTEMP_`<destination file name>

    Where

    `DDTEMP_` denotes a temporary file.

13. Deletes the existing file, if a file exists.

14. Changes the destination file name with the **DDTEMP_** prefix to the original destination filename.

    Destination file does not exist

    Document Delivery:

15. Delivers the append file to the remote location with the name of the destination file pre-pended with **DDTEMP_**.

16. Changes the destination file name with the **DDTEMP_** prefix to the original destination filename.

    HTTP delivery

    HTTP attempts to deliver the first file in the output document. The method that is used is defined in the delivery object configuration.

    Options are:

    - Overwrite
    - Fail On Existing

    Overwrite

    Document Delivery delivers the file unconditionally.

Destination file does exist

Document Delivery overwrites the destination file.

Destination file does not exist

Delivers the file.

Up *(page 45)*

Fail On Existing

Document Delivery checks the remote location to see if the destination file already exists.

Destination file does exist

17. Fails the delivery
18. Re-attempts the delivery, as configured.

Destination file does not exist

Delivers the file.

a. Delivery conclusion

The following occurs based on the status of the delivery.

If in during delivery:

- Any of the delivery steps are interrupted by an error

Document Delivery stores the status of the delivery so a step does not have to be repeated.

- Delivery fails

a. Document Delivery logs an audit message.
b. If a retry count has been configured:

Document Delivery:

a. Attempts the retry after the retry interval has elapsed.
b. Keeps a count of the number of times this particular delivery has failed.
c. If the number of failures is greater than the number of configured failures before alarm:

The CIMPLICITY alarm point triggers an alarm.

a. Document Delivery continues to resend the document until all retries have been attempted.

If the retryCount parameter is set to INFINITE(-1)

Document Delivery will resend the document until it succeeds.

- Delivery succeeds

Document Delivery:

a. Logs an audit message.
b. Resets all error parameters.
c. Deletes the output document.

- Number of retries has been exceeded

Document Delivery:

a. Logs an audit message.
b. Resets all error parameters.
c. Moves the output document to the flush directory.

Note: There may be occasions when the project has been shut down before all deliveries have occurred. When the project starts up again, the directory notification will not see these files as new, so it will not do a notification. To solve this issue, an Document Delivery does an initial directory scan to process any unsent output documents. This scan will put output documents on the output document list in the order that they were created; the oldest document is sent out first.

Document Delivery Directory Structure

The Document Delivery subsystem is dependent on a particular directory structure.

The structure is as follows.

circle 138, 153, 12 *(page)*
circle 139, 105, 12 *(page)*
circle 137, 200, 12 *(page)*
circle 138, 127, 12 *(page)*
circle 138, 171, 12 *(page)*
circle 138, 221, 12 *(page)*
circle 27, 75, 14 *(page)*
circle 49, 95, 14 *(page)*
circle 49, 95, 14 *(page)*
circle 49, 139, 14 *(page)*
circle 48, 187, 14 *(page)*

Consider the following Directory Types:

1. **Base Directory**

   DocumentDelivery is located under the project directory.

2. **Delivery Directories**

Delivery directories are located under the base directory.

Each delivery object has a directory. The name is the name of the delivery object. Delivery directories are monitored for output documents.

3. **Flush Directory**

   Each delivery directory has a Flush directory.

   The flush directory stores output documents that have been flushed. The purpose is to monitor the flushed files to manually find out why the document was not sent. Output documents can be flushed in three ways:
   - **Invalid Header:** If the output document does not have the required header, then there is no way for the document to be delivered, so it is moved to the Flush directory, and an audit message is logged.
   - **Retry Count Exceeded:** If the retry count is exceeded and the output document has still not been successfully delivered, then the document is flushed, and an audit message is logged.
   - **Manual Flush:** If an output document is flushed from the ActiveX status object, then the output document is moved to the flush directory.
4. **Work Directory**

   Each delivery directory has a Work directory.

   The work directory contains the content of the output documents to be delivered without the required header. Any append files that are created and modified also use this work directory.

Configure Document Delivery Objects

Configure Document Delivery Objects

Step	Description
Step 1 *(page 51)*	Enable Document Delivery.
Step 2 *(page 52)*	Open the Document Delivery Base Properties dialog box.
Step 3 *(page 52)*	Configure Document Delivery base properties.

Step	Description
Step 4 *(page 53)*	Open the Delivery Properties dialog box.
Step 5 *(page 55)*	Configure delivery objects.

Step 1. Enable Document Delivery

Important: Document Delivery must be selected as an option when you installed CIMPLICITY. If it does not display in the options lists, re-run your CIMPLICITY DVD and select Documentation Delivery to install.

You can enable Document Delivery in:

- A new project.
- An existing project.

Enable document delivery in a new project

1. Click File>New>Project on the Workbench menu bar.

   A Create as dialog box opens.

2. Check Document Delivery in the Options box.
3. Continue to create the project.

   Result: Document Delivery will be enabled when the project is created. CIMPLICITY adds a Document Delivery folder to the Workbench left pane that contains a Document Delivery Base object and Document Delivery object.

   Enable Document Delivery in an existing project

4. Open the Project Properties dialog box.
5. Select the General tab.
6. Check Document Delivery in the Options box.

Document Delivery is enabled in the existing project. CIMPLICITY adds a Document Delivery folder to the Workbench left pane.

Step 2. Open the Document Delivery Base Properties Dialog Box

Note: There is only one object for the Document Delivery, named DeliveryBase. You cannot delete this object, or create another one. However, you can edit its properties.

1. Select **Project>Document Delivery>Document Delivery Base** in the Workbench left pane.
2. Select **DeliveryBase** in the Workbench right pane.
3. Do one of the following.

A	Click Edit>Properties on the Workbench menu bar.	
B	Click the Properties button on the Workbench toolbar.	
C	In the Workbench left pane: a. Right-click **Document Delivery Base**. b. Select Properties on the Popup menu.	
D	In the Workbench right pane:	
	Either	Or
	Double-click **DeliveryBase**.	a. Right-click **DeliveryBase**. b. Select Properties on the Popup menu.
E	Press Alt+Enter on the keyboard.	

4. Right-click **Document Delivery Base**.
5. Select Properties on the Popup menu.
6. Right-click **DeliveryBase**.
7. Select Properties on the Popup menu.

Step 3. Configure Document Delivery Base Properties

The parameters that can be edited in the Document Delivery Base Properties dialog box are as follows.

rect 23, 59, 273, 88 *(page 53)*
rect 23, 87, 273, 116 *(page 53)*
rect 23, 114, 273, 143 *(page 53)*

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Audit Level</td><td>Main attribute for Auditing. The three levels are:
<table>
<tr><th>Level</th><th>Description</th></tr>
<tr><td>NONE</td><td>No audit file</td></tr>
<tr><td>NORMAL</td><td>Regular audits of delivery</td></tr>
<tr><td>DEBUG</td><td>Detailed audit of operations</td></tr>
</table></td></tr>
<tr><td>Maximum File Size</td><td>File size of Audit log file in bytes. When the log file reaches this size, the audit file will start over logging again.</td></tr>
<tr><td>Archive</td><td>Options are:
<table>
<tr><th>Option</th><th>Details</th></tr>
<tr><td>YES</td><td>The audit file will save itself to a separate file when the file has reached it’s configured size.
  Note: The name of the archived file will be named in the following format: DocumentDelivery_Date_Count.log Where:
• <code>Date=</code>
The current date in the format <code>MMDDYYYY</code>
• <code>Count=</code>
Number that:
◦ Starts at 0
◦ Increments each time the file is archived for the current date.</td></tr>
<tr><td>NO</td><td>No archiving occurs. The audit file starts over with an empty file.</td></tr>
</table></td></tr>
</table>

Step 4. Open the Delivery Properties Dialog Box

Step 4. Open the Delivery Properties Dialog Box

Document Delivery requires at least one delivery object to communicate with a remote system.

Communication with each remote system is defined by parameters that are stored with the delivery object.

- There can be up to 40 delivery objects.
- The name of a delivery object cannot exceed 16 characters.
- The parameters for all delivery objects are stored in the CIMPLICITY proprietary files:
- psdelivery.idx and
- psdelivery.dat.

Options are:

Option	Description
Option 4.1 *(page 54)*	Create a new delivery object.
Option 4.2 *(page 55)*	Open an existing Delivery Properties dialog box.

Option 4.1. Create a new Delivery Object

1. Select **Project>Document Delivery>Document Delivery Object** in the Workbench left pane.
2. Do one of the following.

A	Click File>New>Object on the Workbench menu bar.	
B	Click the New Object button on the Workbench toolbar.	
C	In the Workbench left pane:	
	Either	Or
	Double click **Document Delivery Object**.	a. Right-click **Document Delivery Object**. b. Select New on the Popup menu.
D	a. In the Workbench right pane. a. Right-click any Document Delivery object. b. Select New on the Popup menu.	
E	Press Ctrl+N on the keyboard.	

   Result: A New Document Delivery dialog box opens when you use any method.

3. Right-click **Document Delivery Object**.
4. Select New on the Popup menu.
5. Right-click any Document Delivery object.
6. Select New on the Popup menu.
7. Enter a name for the new delivery object.
8. Click OK.

   Note: You will select the object type *(page 55)* when the Delivery Properties dialog box opens.

Option 4.2. Open an existing Delivery Properties Dialog Box

1. Select Project>Document Delivery in the Workbench left pane.
2. Select a Document Delivery object in the Workbench right pane.
3. Do one of the following.

A	Click Edit>Properties on the Workbench menu bar.	
B	Click the Properties button on the Workbench toolbar.	
C	In the Workbench left pane: a. Right-click **Document Delivery Object**. b. Select Properties on the Popup menu.	
D	In the Workbench right pane:	
	Either	Or
	Double-click a Document Delivery object.	a. Right-click a Document Delivery object. b. Select Properties on the Popup menu.
E	Press Alt+Enter on the keyboard.	

4. Right-click **Document Delivery Object**.
5. Select Properties on the Popup menu.
6. Right-click a Document Delivery object.
7. Select Properties on the Popup menu.

Step 5. Configure Delivery Objects

Step 5. Configure Delivery Objects

Step	Description
Option 5.1 *(page 56)*	Network delivery object.
Option 5.2 *(page 57)*	FTP delivery object.
Option 5.3 *(page 59)*	HTTP delivery object.

Option 5.1. Network Delivery Object

1. Enter a name for the network delivery object in the New Document Delivery dialog box.
2. Click OK.

   A Delivery Properties dialog box opens for the named delivery object.

3. Select NETWORK in the **Delivery Type** drop-down list.

   Parameters for the NETWORK delivery type are as follows.

rect 18, 44, 237, 68 *(page 56)*
rect 17, 67, 236, 91 *(page 56)*
rect 17, 87, 236, 112 *(page 56)*
rect 17, 110, 236, 135 *(page 57)*
rect 243, 45, 491, 69 *(page 56)*
rect 242, 68, 490, 92 *(page 56)*
rect 240, 90, 488, 114 *(page 57)*
rect 237, 112, 485, 136 *(page 57)*

Parameter	Value	
Delivery Type	The type of delivery object. Options are:	
	NETWORK	Mapped network deliveries (selected in this option).
Up *(page 56)*	FTP	FTP deliveries (default).
	HTTP	HTTP deliveries.
Alarm	CIMPLICITY Alarm that will be triggered if a delivery has failed the number of times specified in the Delivery Failure Count parameter. **Note:** A default alarm of $DOCUMENT_DELIVERY has been provided, but a custom alarm can be used.	
Retry Count	Number of times the delivery object will try to resend an output document before it gives up and flushes the file.	
	Range	0 through 999 or infinite.
Failures Before Alarm	The number of times the delivery object needs to fail in order to generate the alarm.	
	Range	0 through 1000 failures
	Note: A value of 0 means no alarm will be generated.	
Retry Interval	Time in seconds to wait before a retry will be attempted.	
	Range	1 through 1000 seconds

File Handling	Determines how to handle the delivery when a remote file of the same name as the one to be delivered is encountered. Options are:	
	OVERWRITE	Overwrites the existing file
Up *(page 56)*	APPEND	Attempts to append the current file to the end of the existing file
	FAIL ON EXISTING	Fails if the remote file exists
Destination	A complete path to the mapped drive.	
	Character limit	100 characters
	Note: If the destination is a mapped drive, CIMPLICITY Service Log On parameters must use an account that has access to the mapped drive, otherwise the PS Delivery will not have access to the mapped drive. The **Destination** field does not support forward slashes.	
Enable Audit	When set to YES enables, auditing for this delivery object.	
	YES	Enables audit.
	NO	There will be no audit.

Option 5.2. FTP Delivery Object

1. Enter a name for the network delivery object in the New Document Delivery dialog box.

2. Click OK.

   A Delivery Properties dialog box opens for the named delivery object.

3. Select FTP in the **Delivery Type** drop-down list.

   Parameters for the FTP delivery type are as follows.

rect 17, 44, 226, 67 *(page 58)*
rect 17, 65, 226, 88 *(page 58)*
rect 17, 87, 226, 110 *(page 58)*
rect 17, 108, 226, 131 *(page 59)*
rect 17, 129, 226, 152 *(page 59)*
rect 17, 150, 226, 169 *(page 59)*
rect 17, 167, 226, 185 *(page 59)*
rect 17, 183, 226, 211 *(page 59)*
rect 233, 46, 466, 71 *(page 58)*
rect 233, 69, 466, 88 *(page 58)*
rect 233, 86, 466, 105 *(page 58)*
rect 233, 103, 466, 122 *(page 59)*
rect 233, 120, 466, 150 *(page 59)*
rect 233, 148, 466, 167 *(page 59)*
rect 348, 170, 439, 189 *(page 59)*

Parameter	Value	
Delivery Type	The type of delivery object. Options are:	
	NETWORK	Mapped network deliveries.
	FTP	FTP deliveries (default-selected in this option).
	HTTP	HTTP deliveries.
Alarm	CIMPLICITY Alarm that will be triggered if a delivery has failed the number of times specified in the Delivery Failure Count parameter. **Note:** A default alarm of $DOCUMENT_DELIVERY has been provided, but a custom alarm can be used.	
Retry Count	Number of times the delivery object will try to resend an output document before it gives up and flushes the file.	
Up *(page 57)*	Range	0 through 999 or infinite.
Failures Before Alarm	The number of times the delivery object needs to fail in order to generate the alarm.	
	Range	0 through 1000 failures.
	Note: A value of 0 means no alarm will be generated.	
Retry Interval	Time in seconds to wait before a retry will be attempted.	
	Range	1 through 1000 seconds
File Handling	Determines how to handle the delivery when a remote file of the same name as the one to be delivered is encountered. Options are:	
	OVERWRITE	Overwrites the existing file.
Up *(page 57)*	APPEND	Attempts to append the current file to the end of the existing file.
	FAIL ON EXISTING	Fails if the remote file exists.

Destination	Name of the directory on the remote machine to which the file will be copied. The parameter must show the path from the root directory.	
	Character limit	100 characters.
	The **Destination** field supports forward slashes (UNIX).	
Enable Audit	When set to YES enables, auditing for this delivery object.	
	YES	Enables audit.
	NO	There will be no audit.
Server Name /IP Address	The FTP server name, or IP Address. The FTP prefix is not required or accepted.	
Timeout	Time out period used during communications.	
Up *(page 57)*	Range	100 to 600,000 ms (100 ms to 10 minutes).
Port	The port number to be used. **Tip:** If you do not know the port number, enter 0 and the Document Delivery process will attempt to determine the correct port number.	
Connection Timeout	Time out period used during the connection process.	
	Range	100 to 600,000 ms (100 ms to 10 minutes).
User Name	Name of the user to log in. **Note:** If empty, the user name is anonymous.	
	Maximum length	20 characters.
Password	Opens a Change Password dialog box in which you can enter a password that will be required for log in.	
	Password length	Up to 127 characters.
	Note: • There is no password on initialization. • If you forget an existing password is forgotten, you will have to delete the delivery object and recreate it. • HTTP *(page 61)* can also have passwords.	
Keep Connection Open	Options are:	
	YES	Delivery object will attempt to keep the FTP Connection open.
	NO	Delivery object will close the FTP Connection on completion of the delivery.

Option 5.3. HTTP Delivery Object

1. Enter a name for the network delivery object in the New Document Delivery dialog box.

2. Click OK.

   A Delivery Properties dialog box opens for the named delivery object.

3. Select HTTP in the **Delivery Type** drop-down list.

   Parameters for the HTTP delivery type are as follows.

rect 18, 44, 226, 67 *(page 60)*
rect 232, 44, 463, 67 *(page 60)*
rect 18, 65, 226, 88 *(page 60)*
rect 18, 86, 226, 109 *(page 61)*
rect 18, 107, 226, 130 *(page 61)*
rect 18, 128, 226, 151 *(page 61)*
rect 18, 149, 226, 172 *(page 61)*
rect 18, 170, 226, 190 *(page 61)*
rect 18, 188, 226, 211 *(page 62)*
rect 232, 65, 463, 88 *(page 60)*
rect 232, 86, 463, 109 *(page 61)*
rect 232, 107, 463, 130 *(page 61)*
rect 232, 128, 463, 151 *(page 61)*
rect 230, 188, 461, 211 *(page 62)*
rect 232, 149, 463, 172 *(page 61)*
rect 345, 171, 438, 189 *(page 61)*

Parameter	Value	
Delivery Type	The type of delivery object. Options are:	
	NETWORK	Mapped network deliveries.
	FTP	FTP deliveries (default-selected in this option).
	HTTP	HTTP deliveries.
Alarm	CIMPLICITY Alarm that will be triggered if a delivery has failed the number of times specified in the Delivery Failure Count parameter. **Note:** A default alarm of $DOCUMENT_DELIVERY has been provided, but a custom alarm can be used.	
Retry Count	Number of times the delivery object will try to resend an output document before it gives up and flushes the file.	
	Range	0 through 999 or infinite.
Failures Before Alarm	The number of times the delivery object needs to fail in order to generate the alarm.	
	Range	0 through 1000 failures.

Up *(page 60)*	**Note:** A value of 0 means no alarm will be generated.		
Retry Interval	Time in seconds to wait before a retry will be attempted.		
	Range	1 through 1000 seconds	
File Handling	Determines how to handle the delivery when a remote file of the same name as the one to be delivered is encountered. Options are:		
	OVERWRITE		Overwrites the existing file.
	FAIL ON EXISTING		Fails if the remote file exists.
Destination	The name of the directory or page on the remote machine to which the file will be copied.		
Up *(page 60)*	The **Destination** field supports forward slashes.		
Enable Audit	When set to YES enables, auditing for this delivery object.		
	YES	Enables audit.	
	NO	There will be no audit.	
Server Name /IP Address	The HTTP server name. The HTTP prefix is not required or accepted.		
	Character limit		100 characters
Timeout	Time out period used during communications.		
	Range	100 to 600,000 ms (100 ms to 10 minutes).	
Port	The port number to be used. **Tip:** If you do not know the port number, enter 0 and the Document Delivery process will attempt to determine the correct port number.		
Connection Timeout	Time out period used during the connection process.		
Up *(page 60)*	Range	100 to 600,000 ms (100 ms to 10 minutes).	
User Name	Name of the user to log in. **Note:** If empty, the user name is anonymous.		
	Maximum length		20 characters.
Password	Opens a Change Password dialog box in which you can enter a password that will be required for log in.		
	Password length		Up to 127 characters.
Up *(page 60)*	**Note:** • There is no password on initialization. • If you forget an existing password is forgotten, you will have to delete the delivery object and recreate it.		

Keep Connection Open	Options are:	
	YES	Delivery object will attempt to keep the HTTP Connection open.
	NO	Delivery object will close the HTTP Connection on completion of the delivery.
Http Command	Options are:	
	PUT	Puts a file to a directory.
	POST	Posts a file to a page.

Change Password Dialog Box

You can change the password for the following two delivery types:

- FTP
- HTTP

Configure the Document Delivery Viewer Object

Configure the Document Delivery Viewer Object

Document Delivery Viewer object is an ActiveX object that a user can use to:

- View the status of one or more delivery objects during runtime
- Delete one or more unsent output documents.

You can place and configure the on a CimEdit/CimView screen.

Note: Multiple Document Delivery Viewer objects can be used to allow the user to view the status of more than one configured delivery objects.

Steps to configure the object are as follows.

Step	Description
Step 1 *(page 63)*	Place the Document Delivery Viewer Object on a CimEdit screen.
Step 2 *(page 63)*	Open the CIMPLICITY Document Delivery Viewer Properties dialog box.

Step	Description
Step 3 *(page 64)*	Configure the Document Delivery Object configuration.
Step 4 *(page 65)*	Configure the Document Delivery Object text.
Step 5 *(page 67)*	Configure the Document Delivery Object appearance.

Step 1. Place the Document Delivery Viewer Object on a CimEdit Screen

1. Open a new CimEdit screen.
2. Do one of the following

   Method 1

   Click the OLE button on the CimEdit toolbar.

   Method 2

   a. Click Tools on the CimEdit toolbar.
   b. Select OLE object.

   An OLE bracket displays on the CimEdit screen.

3. Position the bracket where you want the object's top left corner to be placed.
4. Click the left-mouse button.

   The Insert Object dialog box opens.

5. Double-click CIMPLICITY Document Delivery Viewer.

The Document Delivery ActiveX object is placed on the screen.

Step 2. Open the CIMPLICITY Document Delivery Viewer Properties Dialog Box

Do one of the following.

Method 1

1. Right-click the Document Delivery object.
2. Select CIMPLICITY Document Delivery Viewer Object>Properties from the Popup menus.

   Method 2

3. Click Edit on the CimEdit menu bar.
4. Select CIMPLICITY Document Delivery Viewer Object>Properties from the Edit menus.

The CIMPLICITY Document Delivery Viewer Properties dialog box opens when you use either method.

Step 3. Configure the Document Delivery Object Configuration

The Configuration tab in the CIMPLICITY Document Delivery Viewer Properties dialog box contains the ActiveX Parameters that control how the Document Delivery object operates in a CimView screen.

Select the Configuration tab in the CIMPLICITY Document Delivery.

Configuration options are as follows.

rect 14, 55, 285, 77 *(page 64)*
rect 15, 75, 284, 96 *(page 64)*
rect 17, 94, 285, 114 *(page 64)*
rect 14, 112, 287, 137 *(page 65)*
rect 15, 135, 163, 155 *(page)*

Option	Description
Delivery Object	Delivery object that will be monitored
Update Interval	Interval at which the control will check if the Delivery Object has sent the control a status update. **Important:** If this interval is set to a large number, the control might miss a status update. **Format:** `hh:mm:ss.ttt`
Unresponsive After	Maximum amount of time between status updates from the delivery object that the control will consider the delivery object responsive. If the specified amount of time passes between status updates, the control will enter the Unresponsive state.

<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Flush Permissions</td><td>Roles that are allowed to flush files from the delivery object when the control is running.

Format: Comma-delimited string.

Opens the Configure Flush Permissions dialog box.

Note: Configured roles in this dialog box are automatically added to the Flush Permissions list when you click OK to close the box.

<table>
<tr><th>Role Types</th><th>Description</th></tr>
<tr><td>Available Roles</td><td>Roles that are available, but not given permission to flush unsent files.</td></tr>
<tr><td>Configured Roles</td><td>Roles that have permission to flush unsent files.</td></tr>
</table>
</td></tr>
<tr><td>Allow Flush</td><td>
<table>
<tr><th>Action</th><th>Description</th></tr>
<tr><td>Add</td><td>Adds selected roles to the configured roles list.</td></tr>
<tr><td>Remove</td><td>Removes selected roles from the configured roles list.</td></tr>
<tr><td>Checked</td><td>Enables users with configured roles to flush unsent files.</td></tr>
<tr><td>Unchecked</td><td>Disables flushing capability.</td></tr>
</table>
</td></tr>
</table>

Note: Options on the Configuration tab are writable in CimEdit; they are read-only in CimView.

Step 4. Configure the Document Delivery Object Text

All of the text displayed in the ActiveX Object may be configured to use a different font, size, height, style, character set, and color. Also, the ambient font of the system can be chosen to be used.

Select the Text tab in the CIMPLICITY Document Delivery Viewer Properties dialog box.

Configuration in the Text tab columns are as follows.

rect 17, 48, 116, 102 *(page 65)*
rect 114, 49, 177, 102 *(page 66)*
rect 175, 49, 263, 99 *(page 66)*
rect 261, 50, 347, 101 *(page 67)*

Text

1. Click the New button on the Text tab toolbar.

   A New Text Type dialog box opens.

2. Select a text type that is not on the list.

3. Click OK.

Result: The selected text object is added to the list.

You can configure the following text that displays during runtime.

1	Delivery object label
2	Status label
3	Percent Complete label
4	Destination File Label
5	Failed attempts label
6	Alarm message
7	Retries label
8	List header
9	Queued file name
10	Delivery object value
11	Status value
12	Percent complete value
13	Destination file value
14	Failed attempts value
15	Retries value

Use Ambient

Checked	Uses the ambient properties defined at the CimEdit screen level.
Clear	Enables the color palette and other text properties for custom configuration.

Font/Size

4. Select a line on the Text tab.
5. Do one of the following.

   Method 1

   Click the Font button on the Text tab toolbar.

   Method 2

   Double-click the selected line.

   The Font dialog box opens when you use either method.

Each row in the grid represents a text type that can be displayed in the ActiveX object. The following diagram shows the possible text types that can be configured in the grid.

Color

You can change the color of the selected text, if you have not checked Use Ambient.

6. Double-click the Color field for the selected text line.

   A color palette displays.

7. Click the color for the text.

The color displays in the Color field and will be the color for the selected text.

Step 5. Configure the Document Delivery Object Appearance

You can configure non-text appearance properties on the Document Delivery object on the Appearance tab in the CIMPLICITY Document Delivery Viewer Properties dialog box.

Configuration options are as follows.

rect 19, 58, 280, 82 *(page 67)*
rect 20, 80, 281, 106 *(page 67)*
rect 18, 104, 275, 128 *(page 67)*

Section	Option	Description
1	Background color	Background color of the ActiveX object.
2	File List color	Color of the unsent document list displayed by the ActiveX object when there are unsent documents currently in the delivery object.
3	Displayed Status	Status that is currently displayed in the ActiveX object. This property: • Is only editable from a CimEdit screen since the displayed status in a CimView screen is determined by the status of the ActiveX object. • Can be used to change the status of the ActiveX object to one where the text type being configured is visible, since some text types are not visible while different statuses are displayed.

Up *(page 67)*

circle 196, 163, 23 *(page 67)*
circle 176, 238, 29 *(page 67)*
circle 183, 100, 25 *(page 67)*

Monitor the Delivery Object Runtime Status

There are several different possible views for the ActiveX component based on the status of the delivery object.

- Nonexistent Delivery Object
- In Configuration
- Waiting for Document
- In Delivery
- Retrieving File
- Appending File
- Waiting for Retry
- Flushing Files
- Unresponsive

Nonexistent Delivery Object

Status	Description
Nonexistent	The delivery object specified by the delivery name parameter for the ActiveX component cannot be found

In Configuration

Status	Description
In Configuration	The delivery is currently being configured, either on initialization or by dynamic configuration

Waiting for Document

Status	Description
Waiting for Document	The specified delivery object is waiting for an output document to send.

In Delivery

Status	Description
In Delivery	The specified delivery object is attempting a delivery for the first time.

rect 6, 5, 144, 36 *(page 69)*
rect 10, 33, 141, 57 *(page 69)*
rect 8, 54, 144, 78 *(page 69)*
rect 11, 76, 144, 101 *(page 69)*
rect 5, 101, 147, 122 *(page 69)*
rect 3, 120, 147, 142 *(page 69)*
rect 10, 142, 142, 168 *(page 69)*

Parameter	Description
Delivery Object	Name of the delivery object being monitored.
Status	Status of the delivery object being monitored, as listed on this page.
Percent Complete	Percentage complete of the current delivery. **Note:** This does not include retrieving files for appending.
Destination File	Name of the remote file in delivery.
Failed Attempts	Number of failed attempts for the current output document.
Alarm after Failed Attempts	Number of failed attempts that need to happen before an alarm is sent to the alarm manager.
Retries to attempt	Number of times that a failed delivery will be retried before the delivery object will give up and flush the file.
Unsent Output Documents	List of all the unsent documents that this delivery object still needs to process. **Note:** The name of the output document does not need to be the same as the Destination Filename.

Retrieving File

rect 8, 7, 153, 31 *(page 69)*
rect 3, 32, 152, 57 *(page 69)*
rect 6, 53, 155, 76 *(page 69)*
rect 6, 75, 155, 98 *(page 69)*
rect 6, 98, 135, 121 *(page 69)*
rect 7, 118, 144, 141 *(page 69)*
rect 7, 140, 149, 165 *(page 69)*

Status	Description
Retrieving File	The file is being retrieved. If the file being delivered must be appended to an existing file, the existing file must first be retrieved from the destination.

Appending File

rect 8, 11, 146, 36 *(page 69)*
rect 8, 35, 144, 56 *(page 69)*
rect 6, 59, 144, 78 *(page 69)*
rect 7, 75, 141, 98 *(page 69)*
rect 3, 97, 142, 120 *(page 69)*
rect 7, 123, 141, 144 *(page 69)*
rect 8, 144, 142, 168 *(page 69)*

Status	Description
Appending File	File is being appended. The file being delivered must be appended to an existing file, after the file being appended is retrieved from the destination

Waiting for Retry

rect 8, 53, 144, 77 *(page 69)*
rect 8, 35, 144, 53 *(page 69)*
rect 8, 75, 142, 97 *(page 69)*
rect 10, 95, 140, 120 *(page 69)*
rect 10, 120, 140, 143 *(page 69)*
rect 8, 141, 139, 164 *(page 69)*
rect 8, 7, 144, 36 *(page 69)*

Status	Description
Waiting For Retry	Delivery object is waiting for the retry interval time to elapse in order to retry the delivery. **Note:** All of the text in the ActiveX status object is customizable. There are two different entries that are applicable. • Waiting For Retry Status Before Alarm • Waiting For Retry Status After Alarm. The color of the status text is based on these two entries and whether or not the alarm has been set.

Flushing Files

rect 8, 141, 139, 164 *(page 69)*
rect 10, 120, 140, 143 *(page 69)*
rect 10, 95, 140, 120 *(page 69)*
rect 8, 75, 142, 97 *(page 69)*
rect 8, 53, 144, 77 *(page 69)*
rect 8, 35, 144, 53 *(page 69)*
rect 8, 7, 144, 36 *(page 69)*

Status	Description
Flushing Files	Delivery Object is flushing unsent files.

Unresponsive

Status	Description
Unresponsive	ActiveX control successfully connected to the delivery object but does not receive a status update in a configured time period.

Technical Notes

Document Delivery Technical Notes

- Document Format Rules: Version 101.
- Document Delivery Version 101 XML tags: HTTP or Network.
- Document Delivery Version 101 XML Tags: FTP.
- Document Delivery overview examples.
- Document Delivery Sample Script: Version 101.
- Document Delivery legacy versions.

Output Document Format Rules: Version 101

The output document is created by scripting within the CIMPLICITY environment.

The order is: Version Number; NULL Character; Header; NULL Character; Carriage Return and Line Feed; Content

Document format rules

1. Unique Output Document Name

   Each output document must have a unique name in the directory in which it is created. Each output document is created in a specific directory. As a result, there will be instances where another file already exists in the directory. Giving each output document a unique name will prevent the overwriting of output documents.

2. Version Number (in header)

   The first characters in the output document must be an ASCII representation of the version number.

   - The version number is currently 101.

Note: If a change is made the version number will be incremented.

- The version number will be followed by a NULL character.

3. Header

The header:

- Must immediately follow the version number and NULL character.
- Will be an XML data structure that contains the destination filename and extra commands that may be required for:
  - HTTP or network delivery.
  - FTP delivery.
- The destination will be NULL terminated, followed by a carriage return and line feed.

4. Content

All content to deliver to the remote is located after the:

- Version number
- Destination file name (included in header)
- Separating carriage return
- Line feed

Note: The version number and destination file name will be removed from the document before it is sent to the remote location.

Document Delivery Rule	By	Description
1		
2		
3	Header	
4	Content	

Document Delivery Version 101 XML tags: HTTP or Network

- XML Tags: HTTP or Network
- Examples: HTTP or Network

XML Tags: HTTP or Network

XML Tag	Description		
<Header>	The main wrapper tag for the XML style header. There are no attributes for this tag		
<DestinationFilename>	Contains the remote destination filename. There are two attributes for all systems for this tag		
	`Filename`	The remote destination filename to be delivered to.	
	`Rename`	Options are:	
		`"Yes"`	(Default) The rename feature can be used.
		`"No"`	The rename feature cannot be used. **Important:** If this condition is found on an APPEND type delivery, the delivery will fail; APPEND deliveries must be able to rename.
	Note: The FTP system has two additional attributes.		

Examples: HTTP or Network

When there are no Pre and Post commands, it is not necessary to use the header tag.

These two examples work the same way.

1. No header

```
101 <DestinationFilename filename="remoteFilename.txt" rename="yes"/>
Data
```

2. Header

```
101 <Header>
    <DestinationFilename filename="remoteFilename.txt" rename="yes"/>
</Header>
```

Note: Pre and Post commands are available to FTP delivery only.

If Pre and Post commands are used for Network or HTTP deliveries the:

- Header will be considered invalid.
- Output document will be flushed.

Document Delivery Version 101 XML Tags: FTP

- XML Tags: FTP.
- FTP specific guidelines.
- Examples: FTP.

XML Tags: FTP

There is now a mechanism for doing small FTP commands before and after the actual delivery.

For instance, you might be required to manually change the directory before the delivery takes place, and then change the directory back to the root when finished.

To accomplish this, the Pre and Post commands can be used.

XML Tag	Description
<Header>	The main wrapper tag for the XML style header. There are no attributes for this tag
<DestinationFilename>	Contains the remote destination filename. There are four attributes for all systems for this tag: `Filename`: The remote destin • ation filename to be delivered to. • `Rename`: Allows the file to be renamed during the delivery process. Options are: Yes and No. `"Yes"`: (Default) The rename feature can be used. `"No"`: The rename feature cannot be used. **Important:** If this condition is found on an APPEND type delivery, the delivery will fail; APPEND deliveries must be able to rename. • `KeepConnectionOpen`: Options are Yes and No. `"Yes"`: An FTP connection will stay open. `"No"`: An FTP connection is not forced to stay open. • `overrideKeepConnectionOpen`: Options are Yes and No. `"Yes"`: Enables ability to override the open FTP connection so the connection can close. `"No"`: Does not allow an override to the open FTP connection tag.
`PreCommands`	Wrapper for all Pre commands that are to be attempted before the delivery occurs. There are no attributes for this tag.
`PreCommand`	Holds one FTP command to take place before the delivery occurs. The raw FTP command is written verbatim in the command attribute. There is one attribute for this tag. `command`: The command to be sent.
`PostCommands`	Wrapper for all the Post commands that are to be attempted after the delivery is completed. There are no attributes for this tag.
`PostCommand`	Holds one FTP command to take place after the delivery is successfully completed. The raw FTP command is written verbatim in the command attribute. There is one attribute for this tag. `command`: The command to be sent.

guide: FTP specific guidelines

- Multiple PreCommand and PostCommand tags may be used.
- The total size of the header including the version, null characters and Carriage Return Line Feed cannot exceed 4096 bytes.
- Failures in the delivery process are treated as follows.

If the following fails:	Result
Pre commands	Delivery and the post commands are not processed. Delivery is considered to be failed.
Delivery	Post commands are not processed. Delivery is considered to be failed.
Post commands	Delivery is considered to be failed.

Note: If the actual delivery of the remote file succeeds and the commands fail, subsequent delivery attempts will try to do only the pre and post commands since the delivery of the actual file has already succeeded.

- If Pre and Post commands are used for Network or HTTP deliveries, the header will be considered invalid, and the output document will be flushed.

FTP Example: using Pre and Post commands

1. Simple file

```
101 <Header>
    <DestinationFilename filename="remoteFilename.txt"/>
</Header>
Data
```

2. Files using all tags

   A

```
101 <Header>
    <PreCommands>
        <PreCommand command="SITE Stats"/>
    </PreCommands>
    <DestinationFilename filename="remoteFilename.txt" rename="No"/>
    <PostCommands>
        <PostCommand command="SITE file=yes"/>
    </PostCommands>
</Header>
```

   B

```
101 <Header>
    <PreCommands>
        <PreCommand command="CWD subdirectory"/>
    </PreCommands>
    <DestinationFilename filename="DestinationFilename.txt"
 rename="yes"/>
    <PostCommands>
        <PostCommand command="CDUP"/>
    </PostCommands>
</Header>
Data to be sent.
```

   FTP Example: Using Connection Open commands

   This example will force the connection to close after the delivery has been completed.

```
<DestinationFilename
```

```
    filename="FTP_O_RemoteFilename230.txt"
    rename="Yes"
    overrideKeepConnectionOpen="YES"
    keepConnectionOpen="No"/>
```

Document Delivery Overview Examples

The following two simple examples demonstrate how a file is delivered using the overwrite and append methods.

- Overwrite delivery method.
- Append delivery method.

Overwrite Delivery Method

- **Delivery Object:** FTPobj
- **Method:** Overwrite any existing files
- **Delivery filename:** FinalDestinationFileName.txt

The content of the file to be delivered is as follows.

`101 <DestinationFilename name=”FinalDestinationFileName.txt”/>`
Data

1. This file is saved as **OutputDocument1**.

2. **OutputDocument1** is copied to the `\Project\DocumentDelivery\FTPobj` directory.

3. The directory scanner:
   a. Finds **OutputDocument1**.
   b. Adds the name of this file to a preliminary list.
   c. Goes back to scanning.

4. **FTPobj**:
   a. Monitors the preliminary list
   b. (When a file appears) creates a file, **OutputDocument1**, in the `\Project \DocumentDelivery\FTPobj\Work` directory.

   OutputDocument1 is the original file with the header stripped out.

   The content of the file is Data

   a. Checks the remote location to see if the file FinalDestinationFileName.txt exists in the configured remote directory.
   b. Delivers the existing **OutputDocument1** document

From the `\ProjectDocumentDelivery\FTPobj\Work` directory

To the remote file DDTEMP_FinalDestinationFileName.txt.

5. If FinalDestinationFileName.txt is found on the remote system, that file is deleted.
6. DDTEMP_FinalDestinationFileName.txt is renamed to FinalDestinationFileName.txt.
7. The delivery is complete.

   Append Delivery Method

Delivery Object	FTPobj
Method	Append existing files
Delivery filename	FinalDestinationFileName.txt

   The content of the file is as follows.

`101 <DestinationFilename name="FinalDestinationFileName.txt"/>`
Data

8. This file is saved as **OutputDocument2**.
9. **OutputDocument2** is copied to the `\Project\DocumentDelivery\FTPobj` directory.
10. The directory scanner:
    a. Finds **OutputDocument2**.
    b. Adds the name of this file to a preliminary list.
    c. Goes back to scanning.
11. **FTPobj**:
    a. Monitors the preliminary list.
    b. (When a file appears) creates a file, **OutputDocument2**, in the `\Project \DocumentDelivery\FTPobj\Work` directory.

OutputDocument2 is the original file with the header stripped out.

The content of the file is More Data.

a. Checks the remote location to see if the file FinalDestinationFileName.txt exists in the configured remote directory.
b. Finds the file that was just delivered using the Overwrite *(page 76)* method.
c. Retrieves the remote FinalDestinationFileName.txt from the remote system.
d. Copies the remote file to a file DDTEMP_FinalDestinationFileName.txt in the `\Project \DocumentDelivery\FTPobj\Work` directory.

e. (In the `\Project\DocumentDelivery\FTPobj\Work` directory) copies the contents of **OutputDocument2** to the end of the DDTEMP_FinalDestinationFileName.txt file.

The content of the file is Data More Data.

a. Delivers the DDTEMP_FinalDestinationFileName.txt file

From the `\ProjectDocumentDelivery\FTPobj\Work` directory

To the remote file DDTEMP_FinalDestinationFileName.txt.

12. The current FinalDestinationFileName.txt file on the remote machine is deleted.
13. DDTEMP_FinalDestinationFileName.txt is renamed to FinalDestinationFileName.txt.
14. he delivery is complete.

Document Delivery Sample Script: Version 101

Note: The header in this sample script is the minimum required and can be used for all delivery types.

```
Const NULLCHARACTER = Chr$(0)
Const CRLF = Chr$(13) + Chr$(10)
Const QUOTE = Chr$(34)
Const DOCUMENTDELIVERYVERSION = "101"
Sub OnMouseUp(x As Long, y As Long, flags As Long)
```

'Set up the destination file name.

'Format Bottlingyyyymmdd.txt

```
Dim destinationFileName as string
destinationFileName = "Bottling"
Dim currentDate As Date
currentDate = Date
destinationFileName = destinationFileName + format( currentDate, "yyyy" )
destinationFileName = destinationFileName + format( currentDate, "mm" )
destinationFileName = destinationFileName + format( currentDate, "dd" )
destinationFileName = destinationFileName + ".txt"
```

'Set up the required Document Delivery Header

```
dim DocumentDeliveryHeader as string
DocumentDeliveryHeader = DOCUMENTDELIVERYVERSION
DocumentDeliveryHeader = DocumentDeliveryHeader + NULLCHARACTER
```

‘ Now set up the XML Header

```
DocumentDeliveryHeader = DocumentDeliverHeader + "<Header>"
DocumentDeliveryHeader = DocumentDeliverHeader + "<DestinationFilename
 filename="
DocumentDeliveryHeader = DocumentDeliverHeader + QUOTE +
 destinationFileName + QUOTE
DocumentDeliveryHeader = DocumentDeliverHeader + "/>"
DocumentDeliveryHeader = DocumentDeliverHeader + "</Header>"
DocumentDeliveryHeader = DocumentDeliveryHeader + NULLCHARACTER + CRLF
```

'Start the content string

```
Dim documentContent as string
```

'Total Bottles

```
Dim totalBottles As New point
totalBottles.Id = "\\NORTH1\TOTALBOTTLES"
totalBottles.Get
documentContent = "Total Bottles = " + cstr( totalBottles.Value )
documentContent = documentContent  + CRLF
```

'Bottles Failed

```
Dim bottlesFailed As New point
bottlesFailed.Id = "\\NORTH1\BOTTLESFAILED"
bottlesFailed.Get
documentContent = documentContent  + "Bottles Failed = " +
 cstr( bottlesFailed.Value )
documentContent = documentContent  + CRLF
```

'Gripper Errors

```
Dim gripperErrors As New point
gripperErrors.Id = "\\NORTH1\GRIPPERERRORS"
gripperErrors.Get
documentContent = documentContent  + "Gripper Errors = " +
 cstr( gripperErrors.Value )
documentContent = documentContent  + CRLF
```

'Gate Open Errors

```
Dim gateOpenErrors As New point
gateOpenErrors.Id = "\\NORTH1\GATEOPENERRORS"
gateOpenErrors.Get
documentContent = documentContent  + "Gate Open Errors = " +
 cstr( gateOpenErrors.Value )
documentContent = documentContent  + CRLF
```

'No Box Errors

```
Dim noBoxErrors As New point
noBoxErrors.Id = "\\NORTH1\NOBOXERRORS"
noBoxErrors.Get
documentContent = documentContent  + "No Box Errors = " +
 cstr( noBoxErrors.Value )
documentContent = documentContent  + CRLF
```

'Boxes Processed

```
Dim boxesProcessed As New point
boxesProcessed.Id = "\\NORTH1\BOXESPROCESSED"
boxesProcessed.Get
documentContent = documentContent  + "Boxes Processed = " +
 cstr( boxesProcessed.Value )
documentContent = documentContent  + CRLF
```

'A unique Filename is required for Document Delivery

```
dim outputDocument as string
outputDocument = "MYDelivery"
Dim uniqueIndex As New Point
uniqueIndex.Id = "\\NORTH1\MYDELIVERYINDEX"
uniqueIndex.Get
outputDocument = outputDocument + cstr(uniqueIndex.value)
```

'Increment the index for the next usage

```
uniqueIndex.value = uniqueIndex.value + 1
uniqueIndex.set
```

'Write to a temporary file

```
Dim temporaryFilename As String
temporaryFilename = "c:\" + outputDocument
Open temporaryFilename For Output Access Write Lock Read Write As #1
Print #1, DocumentDeliveryHeader
Print #1, documentContent
Close #1
```

'Copy the file to the correct directory

```
Dim root As String
root = Environ( "SITE_ROOT" )
Dim finalDestination As String
finalDestination = root + "\DocumentDelivery\MyDelivery\" + outputDocument
FileCopy temporaryFilename, finalDestination
```

'Delete the temporary file

```
Kill temporaryFilename
End Sub
```

Document Delivery Legacy Versions

Document Delivery Legacy Versions

- Output Document Format: Version 100.
- Document Delivery sample script: Version 100.

Output Document Format Rules: Version 100

CAUTION: Version 100 is still supported for systems in which it is implemented. Use version 101 to develop any new systems.

Rules for Version 100 are as follows.

Document Delivery Rule		Description
1	Unique Output Document Name	Each output document must have a unique name in the directory in which it is created. Each output document is created in a specific directory. As a result, there will be instances where another file already exists in the directory. Giving each output document a unique name will prevent the overwriting of output documents.
2	Version Number (in header)	The first characters in the output document must be an ASCII representation of the version number. The initial version number will be 100 until a change is made that will make it necessary to increment the version number. The version number will be followed by a NULL character.
3	Destination File Name	The destination file name must immediately follow the version number and NULL character. This will be the name of the file actually created or appended to at the remote location. This destination will be NULL terminated, followed by a carriage return and line feed.
4	Content	All content to deliver to the remote is located after the: • Version number, • Destination file name, • Separating carriage return and • Line feed. **Note:** The version number and destination filename will be removed from the document before it is sent to the remote location.

Document Delivery Sample Script: Version 100

Note: The header in this sample script is the minimum required and can be used for all delivery types.

```
Const NULLCHARACTER = Chr$(0)
Const CRLF = Chr$(13) + Chr$(10)
Const DOCUMENTDELIVERYVERSION = "100"
Sub OnMouseUp(x As Long, y As Long, flags As Long)
```

'Set up the destination file name.

'Format Bottlingyyyymmdd.txt

```
Dim destinationFileName as string
destinationFileName = "Bottling"
Dim currentDate As Date
currentDate = Date
destinationFileName = destinationFileName + format( currentDate, "yyyy" )
destinationFileName = destinationFileName + format( currentDate, "mm" )
destinationFileName = destinationFileName + format( currentDate, "dd" )
destinationFileName = destinationFileName + ".txt"
```

'Set up the required Document Delivery Header

```
dim DocumentDeliveryHeader as string
DocumentDeliveryHeader = DOCUMENTDELIVERYVERSION
DocumentDeliveryHeader = DocumentDeliveryHeader + NULLCHARACTER
DocumentDeliveryHeader = DocumentDeliveryHeader + destinationFileName
DocumentDeliveryHeader = DocumentDeliveryHeader + NULLCHARACTER + CRLF
```

'Start the content string

```
Dim documentContent as string
```

'Total Bottles

```
Dim totalBottles As New point
totalBottles.Id = "\\NORTH1\TOTALBOTTLES"
totalBottles.Get
documentContent = "Total Bottles = " + cstr( totalBottles.Value )
documentContent = documentContent  + CRLF
```

'Bottles Failed

```
Dim bottlesFailed As New point
bottlesFailed.Id = "\\NORTH1\BOTTLESFAILED"
bottlesFailed.Get
documentContent = documentContent  + "Bottles Failed = " +
 cstr( bottlesFailed.Value )
```

```
documentContent = documentContent  + CRLF
```

'Gripper Errors

```
Dim gripperErrors As New point
gripperErrors.Id = "\\NORTH1\GRIPPERERRORS"
gripperErrors.Get
documentContent = documentContent  + "Gripper Errors = " +
 cstr( gripperErrors.Value )
documentContent = documentContent  + CRLF
```

'Gate Open Errors

```
Dim gateOpenErrors As New point
gateOpenErrors.Id = "\\NORTH1\GATEOPENERRORS"
gateOpenErrors.Get
documentContent = documentContent  + "Gate Open Errors = " +
 cstr( gateOpenErrors.Value )
documentContent = documentContent  + CRLF
```

'No Box Errors

```
Dim noBoxErrors As New point
noBoxErrors.Id = "\\NORTH1\NOBOXERRORS"
noBoxErrors.Get
documentContent = documentContent  + "No Box Errors = " +
 cstr( noBoxErrors.Value )
documentContent = documentContent  + CRLF
```

'Boxes Processed

```
Dim boxesProcessed As New point
boxesProcessed.Id = "\\NORTH1\BOXESPROCESSED"
boxesProcessed.Get
documentContent = documentContent  + "Boxes Processed = " +
 cstr( boxesProcessed.Value )
documentContent = documentContent  + CRLF
```

'A unique Filename is required for Document Delivery

```
dim outputDocument as string
outputDocument = "MYDelivery"
Dim uniqueIndex As New Point
uniqueIndex.Id = "\\NORTH1\MYDELIVERYINDEX"
uniqueIndex.Get
outputDocument = outputDocument + cstr(uniqueIndex.value)
```

'Increment the index for the next usage

```
uniqueIndex.value = uniqueIndex.value + 1
```

```
uniqueIndex.set
```

'Write to a temporary file

```
Dim temporaryFilename As String
temporaryFilename = "c:\" + outputDocument
Open temporaryFilename For Output Access Write Lock Read Write As #1
Print #1, DocumentDeliveryHeader
Print #1, documentContent
Close #1
```

'Copy the file to the correct directory

```
Dim root As String
root = Environ( "SITE_ROOT" )
Dim finalDestination As String
finalDestination = root + "\DocumentDelivery\MyDelivery\" + outputDocument
FileCopy temporaryFilename, finalDestination
```

'Delete the temporary file

```
Kill temporaryFilename
End Sub
```

Chapter 6. Microsoft Remote Desktop and CIMPLICITY

About Microsoft Remote Desktop and CIMPLICITY

When you set up a CIMPLICITY server as a Remote Desktop server, you can use the CIMPLICITY Web site to make use of the Remote Desktop features.

Microsoft provides extensive documentation for Remote Desktops, both in Windows documentation and at the Microsoft web site.

In that documentation, Remote Desktop is defined as follows:

- Microsoft Corporation provides licenses to run Remote Desktop on a Windows Remote Desktop Server and a specified number of clients. Review Microsoft documentation for current details about Remote Desktop licenses.

- GE Digital provides licenses that enable Remote Desktop clients to work with CIMPLICITY projects.

Remote Desktop Quick Setup Guide

Remote Desktop Quick Setup Guide

Steps for the CIMPLICITY Remote Desktop setup include:

Step	Description
Step 1 *(page 86)*	Set up the Remote Desktop security.
Step 2 *(page 86)*	Configure CIMPLICITY on the Remote Desktop server.
Step 3 *(page 86)*	Use CIMPLICITY through a Remote Desktop session.

Note: Remote Desktop has a 256-color limit.

Step 1. Configure Remote Desktop Security Levels

To ensure that Remote Desktop clients have the appropriate levels of control in the Remote Desktop server, refer to your Microsoft documentation, as well as the documentation on CIMPLICITY users, to organize the necessary privilege levels.

Step 2. Configure CIMPLICITY on the Remote Desktop Server

The Remote Desktop server must be installed on a CIMPLICITY server. Microsoft Remote Desktop enables clients to work with all supported CIMPLICITY features for which a user has privileges, as follows.

- Remote Desktop clients directly access the CIMPLICITY Remote Desktop server.
- Remote Desktop clients interact directly with the CIMPLICITY Remote Desktop server. The client displays runtime data from the CIMPLICITY server and sends setpoint data to the CIMPLICITY server database

PCP = Point Control Panel

Step 3. Use CIMPLICITY through a Remote Desktop Session

Users who have the required licenses can work with CIMPLICITY through the Remote Desktop options.

CIMPLICITY can be used almost the same as if a user is sitting at the project's PC when any Remote Desktop session is being used.

There are some guidelines, mainly to insure the proper implementation of project revisions if multiple users are working with a project at the same time through Remote Desktop sessions.

Guidelines for CIMPLICITY Projects through Remote Desktop

A user can control any feature on the Remote Desktop server for which he or she has authorization.

Users can work with CIMPLICITY as if they are sitting at the Remote Desktop server.

1. You can have different CIMPLICITY Remote Desktop client sessions open simultaneously so you can work concurrently on different projects in CIMPLICITY applications.

Example: If two clients display the Workbench on each desktop and one user closes the Workbench, the second client desktop continues to display the Workbench.

2. Any changes made to CIMPLICITY during all client sessions go directly into the CIMPLICITY database.

3. All projects run in the global session on a CIMPLICITY Remote Desktop server-CIMPLICITY server. Therefore, if several people are working on a project and one person stops it, the project stops for all of them.

   Two effects are:

   a. If a client user with configuration privileges is involved in configuration, e.g. creating a new point, the dynamic configuration is no longer running. The user will have to do a configuration update in order to implement the changes.
   b. Runtime applications for the project that shuts down will either:
   - No longer display data if more than one project is running (so the router is still running) or
   - Shut down if there was only one project running (and the router has stopped).

4. When the router stops on a CIMPLICITY Remote Desktop server-CIMPLICITY client, runtime applications stop on the Remote Desktop server and for all CIMPLICITY Remote Desktop clients.

CIMPLICITY Projects through Remote Desktop Global Parameters List

You can use the following global parameters with CIMPLICITY through Remote Desktop:

Global Parameter	Use to:
GSM_TERMSERV_CACHE_SIZE	Limit the CimView cache when running on a terminal server.
TERMSERV_ALLOW_SETPOINTS	Allow or prohibit setting points from a Remote Desktop client.

CIMPLICITY Features that Work with Remote Desktop

List: CIMPLICITY Features Supported with Remote Desktop

Feature	Supported?
Action Calendar	Yes

Feature	Supported?
Alarm Management API	Yes
Alarm Sound Manager	No
Alarm Viewer	Yes
BCE (Basic Program Editor, Event Manger)	Yes
BCEUI	Yes
CimEdit/CimView	Yes
CIMPLICITY Options	Yes
CWServ	Yes
Database Server (MSDE)	Yes
DGR	No
Machine Edition integration	No
Login Panel	Yes
Marquee Driver	Yes
OPC Server (CIMPLICITY, not devcoms)	Yes
Pager	No
Point Control Panel	Yes
Point Cross Reference	Yes
Point Management API	Yes
Process Control	Yes
Recipes	Yes.
Registration/Licensing	Yes
Report Manager	No
Server Redundancy	No*
Show Users	Yes
SPC Charts	Yes
Status Log	Yes
System Sentry	Yes
Tracker (PRT User Interface, RCO Runtime User Interface)	Yes
Workbench	Yes

*Remote Desktop client failover is not supported; remote administration through Remote Desktop is supported.

CIMPLICITY Device Communications that Work with Remote Desktop

The following device communications are supported with Remote Desktop.

Device Communications	Data Collection	Remote Configuration
Allen Bradley Communications (RSLINX)	Y	Y
Allen-Bradley DF-1	Y	Y
Allen-Bradley RF-ID	NA	NA
CCM2	Y	Y
DC Toolkit	Y	Y
DDE/DDE Client	Y	Y
FloPro/FloNet Ethernet	N	N
Genius	N	N
Honeywell IPC 620	Y	Y
Johnson Controls N2 But	Y	Y
Mitsubishi Serial	N	N
Mitsubishi TCP/IP	Y	Y
MODBUS RTU	Y	Y
MODBUS RTU Standby	Y	Y
MODBUS TCP/IP	Y	Y
MODBUS TCP/IP Standby	Y	Y
N2 Serial	Y	Y
Omron Host Link	Y	Y
OMRON TCP/IP	Y	Y
OPC Client	Y	Y
SCADA Driver Client	Y	N
Series 90 Ethernet	Y	Y
Sharp TCP/IP	Y	Y
Siemens TI Serial	Y	Y
SMARTEYE Electronic Assembly	Y	Y
SNP	Y	Y
SNPX	Y	Y

Device Communications	Data Collection	Remote Configuration
SQUARE D	Y	Y
TOYOPUC TCP/IP	NA	NA
Triplex	Y	Y

NA = Not tested.

Chapter 7. CIMPLICITY Cluster Resource

About the CIMPLICITY Cluster Resource

CIMPLICITY Cluster, which is incorporated into the Microsoft Cluster technology, is available in Proficy™ HMI/SCADA CIMPLICITY® version 6.2 or higher.

A license is required to run CIMPLICITY on a cluster. If you do not have a license you can configure a project, open it from the Workbench and run it as you would any CIMPLICITY project. However, it cannot be started by the Cluster Manager and will not have the cluster benefits. The license is programmed into your hardware key.

CIMPLICITY Cluster Resource Overview

Note: The Failover Cluster Administrator is included in the Microsoft® Windows® Server 2012, Microsoft® Windows® Server 2012 R2, and Microsoft® Windows® Server 2016 operating systems.

See Microsoft documentation about cluster administration and requirements.

When you include CIMPLICITY in a cluster, the project:

- Must be on a drive that is connected to more than one node.
- Must have a dependency on that shared drive where the files are located.

The project is:

- A resource.
- Dependant on the physical disk.
- Assigned to a cluster.
- Started either when the cluster is brought online, or in the Workbench, which can be opened through the resource's Properties dialog box.

The Cluster reports the project status depending on where the project is started, as follows.

Start Project through the:	Then the cluster will:
Cluster	• Watch the project. • Indicate that it is online.

Start Project through the:	Then the cluster will:
Workbench	• Watch the project. • Indicate that the project is off line until you bring it online (through the cluster).
Workbench And Bring it online through the Cluster	• Watch the project. • Indicate that it is online.

The Cluster reports the project status depending on where the project is started/stopped, as follows.

Start Project through the:	Stop Project through the:	Then the cluster will indicate that the:
Cluster	Workbench	Project failed.
Workbench	Workbench	Project is off line.

The CIMPLICITY Router keeps track of what IP addresses are in use on the computer.

Important: You cannot use the cluster IP address as one of the IP addresses so CIMPLICITY will not listen for incoming connections on the cluster IP address.

CIMPLICITY Cluster Resource Example

- CIMPLICITY files are installed on Drive G:
- Drive **G:** is connected to four nodes.
- Node 1 currently controls **G:**
- Nodes 2, 3 and 4 will not be able to access the CIMPLICITY project until it fails over or is moved.
- The node it fails over or is moved to will then be in control of the project instead of Node 1.

Configuration on Windows Servers

- Cluster: Configuration Microsoft® Windows® Server 2012/2012 R2/2016 *(page 93)*
- Technical Reference: Cluster Configuration *(page 102)*

Important: A license is required to run CIMPLICITY on a cluster. If you do not have a license you can configure a project, open it from the Workbench and run it as you would any CIMPLICITY project. However, it cannot be started by the Cluster Manager and will not have the cluster benefits. The license is programmed into your hardware key .

Cluster: Configuration Microsoft® Windows® Server 2012/2012 R2/2016

Cluster: Configuration Microsoft® Windows® Server 2012/2012 R2/2016

Steps to configure a cluster on a Microsoft® Windows® Server 2012/2012 R2/2016 are as follows.

Step	Description
Step 1 *(page 93)*	Create a Cluster
Step 2 *(page 94)*	Configure a Role
Step 3 *(page 95)*	Add Storage to a Cluster
Step 4 *(page 95)*	Make an IP Address Available for a Cluster
Step 5 *(page 96)*	Add a CIMPLICITY Project Resource
Step 6 *(page 97)*	Configure the CIMPLICITY Project Resource Parameters
Step 7 *(page 102)*	Bring the CIMPLICITY Project Resource Online

Step 1. Create a Cluster

For Microsoft® Windows® Server 2012/2012 R2/2016: Open the Failover Cluster Manager and Create a Cluster.

Open the Failover Cluster Manager

Create a Cluster

Do one of the following:

- Click Action>Create Cluster on the Failover Cluster Manager menu bar.
- Right-click Failover Cluster Manager in the Failover Cluster Manager left-pane; select Create Cluster on the Popup menu.

Either method creates a cluster.

Step 2. Configure a Role

For Microsoft® Windows® Server 2012/2012 R2/2016: Create an Empty Role, Open a New Role Properties Dialog Box, and Configure the New Role Properties.

1. To create an Empty Role, expand the Cluster tree in the Failover Cluster Manager left-pane.
2. Do either of the following.

A	Click Action>Create Empty Role on the Failover Cluster Manager menu bar.
B	Right-click Failover Cluster Manager in the Failover Cluster Manager left-pane; select Create Cluster on the Popup menu.

   A new role is create in the cluster.

3. Open a New Role Properties Dialog Box by select the new role in the Failover Cluster Manager window center-pane.
4. Make sure the role is stopped.
5. Do either of the following.

A	Right-click the new role; select Properties on the Popup menu.
B	Click New Role>Properties on the Failover Cluster Manager window right-pane.

   A New Role Properties dialog box opens when you use either method.

   Configure the New Role Properties

   The New Role Properties dialog box provides the following options.

   - General Tab
   - Failover Tab

   General Tab

Select the General tab in the New Role Properties dialog box. Options are as follows:

A	Name	Identifies the role in the cluster.	
B	Preferred Owners	Machines that may be available for the selected role.	
		Checked	The machine will be available
		Clear	The machine will not be available.
		Consult Microsoft documentation for details about preferred owners.	

Failover Tab

Failover values must be entered. However, they are dependent on your system needs.

There are no specific recommendations or requirements for a cluster that uses a CIMPLICITY project resource.

Step 3. Add Storage to a Cluster

For Microsoft® Windows® Server 2012/2012 R2/2016, add storage to a cluster.

1. Select the new role item in the Failover Cluster Manager window center-pane.
2. Do either of the following.

A	Right-click the new role; select Add Storage on the Popup menu.
B	Click New Role>Add Storage on the Failover Cluster Manager window right-pane.

   A Storage browser opens listing the available storage disks

3. Check the disks that should be available for storage.
4. Click OK.
   All of the selected disks will be available; one disk will be selected as a dependency for the CIMPLICITY project resource

Step 4. Make an IP Address Available for a Cluster

For Microsoft® Windows® Server 2012/2012 R2/2016, make an IP Address Available for a Cluster. Open the New Resource Wizard, Configure the New Resource Wizard, and then Bring the Resource Online.

1. Select the new role item in the Failover Cluster Manager window center-pane.
2. Do either of the following.

A	Right-click the new role; select Add Resource>Client Access Point on the Popup menu.
B	Click New Role>Add Resource>Client Access Point on the Failover Cluster Manager window right-pane.

   A New Resource Wizard opens.

3. From the Client Access Point Screen, enter and check the following.

1	Name	Name that identifies the network
2	Address	The IP address for each selected (checked) network.

4. Click Next.
   A Confirmation screen opens.
5. Check the new resource details.
6. Click Next to confirm the configuration.

   The New Resource Wizard configures the access point; a Summary screen opens.

   A Summary screen reports the configuration details.

7. Click Finish.
   Details are listed in the Failover Cluster Manager middle-pane.
8. Bring the server/IP address online before you configure the CIMPLICITY resource in order to make sure they will be available for that resource. Bring the resources online, as follows:

   a. Right-click an IP Address resource; select Bring Online on the Popup menu.

   b. Do the same for all networks that should be brought online.

The selected networks are online.

Step 5. Add a CIMPLICITY Project Resource

For Microsoft® Windows® Server 2012/2012 R2/2016, add a CIMPLICITY Project Resource.

1. Right-click a new role in the Failover Cluster Manager middle-pane.

2. Select the following on the Popup menu.

A	Add Resource
B	More Resources
C	CIMPLICITY Project

Result: A new CIMPLICITY project resource is added to the Other Resources list in the Failover Cluster Management window middle-Pane.

The Default Name is New CIMPLICITY Project.

Step 6. Configure the CIMPLICITY Project Resource Parameters

Even if you used the Microsoft High Efficiency Wizard to configure most of the CIMPLICITY cluster, you can customize the configuration for the CIMPLICITY resource cluster in the CIMPLICITY Project Resource Name Properties dialog box. These steps are for Microsoft® Windows® Server 2012/2012 R2/2016.

Step	Description
A *(page 97)*	Open the <CIMPLICITY Project Resource Name> Project Properties Dialog Box
B *(page 97)*	Configure the <CIMPLICITY Project Resource Name> Project Properties Dialog Box

A. Open the <CIMPLICITY Project Resource Name> Project Properties Dialog Box

Right-click a CIMPLICITY project resource; select Properties on the Popup menu.

Result: A <CIMPLICITY Project Resource Name> Project Properties dialog box opens.

B. Configure the <CIMPLICITY Project Resource Name> Project Properties Dialog Box

The following tabs are available for other Microsoft cluster resources. However, values that are entered for many cluster configuration options are based on system requirements for a CIMPLICITY cluster, as follows

Review Microsoft documentation for detailed descriptions of cluster resource configuration, in general.

- General Tab
- Dependencies Tab
- Policies Tab
- Advanced Policies Tab
- Parameters Tab (CIMPLICITY project)

General Tab

General specifications based on CIMPLICITY cluster requirements are as follows.

Item	Description
Resource Name	Name that clearly identifies the CIMPLICITY resource. • Any name is acceptable for the resource. • The title bar will display a changed name after the dialog box is closed and re-opened. Default: New CIMPLICITY Project Maximum Length
Resource type	(Read-only) CIMPLICITY Project
State	(Read-only) A resource needs to be brought online for runtime performance. **Note:** During initial configuration, the CIMPLICITY resource has not yet been brought online. States are as follows: • Offline state - The resource will not be enabled during runtime. • Online state - The resource is enabled for runtime.

Dependencies Tab

The CIMPLICITY project resource dependencies (e.g. disk drives, IP address) are listed on the Dependencies tab.

You can select as many available resource dependencies as needed from a drop down list.

Important: Required dependencies are as follows.

Section	Object	Description
1	Network Name	Network name assigned to the CIMPLICITY project resource.
2	Shared Disk	The shared disk on which the CIMPLICITY project resource resides.

Section	Object	Description
3	IP Address	An IP address that is assigned to the CIMPLICITY project resource. **Note:** When the project is made dependent on the IP address resource then all of the components in the cluster will move as a group. This will make the cluster more stable than if an IP address was not assigned.

Important: Important:Click Apply before you select another tab or close the <CIMPLICITY Project Resource Name> Properties dialog box. This is required in order to confirm your selections. If you do not click Apply, the selections will not be applied.

Policies Tab (restart)

Policy specifications based on CIMPLICITY cluster requirements are as follows.

rect 0, 308, 30, 338 *(page 101)*
rect -1, 229, 29, 259 *(page 101)*
rect 0, 190, 30, 220 *(page 100)*
rect 0, 84, 30, 114 *(page 100)*

Section	Respond to resource failure
1	Select either of the following. • If resource fails, do not restart. The failover will take over without attempting any restart of the process • If resource fails, attempt restart on current node. Requires entries based on the selected CIMPLICITY project resource requirements. • Period of restarts `(mm:ss)`: The number of seconds the resource should wait for the project resource to attempt to restart. The resource will fail after the entered time. **Note:** Include the entered **Pending timeout** value (below) in your calculation to help insure that the desired number of restart attempts will occur. • Maximum restarts in the specified period: The maximum number of restart attempts that will be performed in the specified restart time period. **Note:** If the resource does not start, the Cluster service will take the actions specified on the Policies tab.
2	If restart is unsuccessful, fail over all resources in the service or application. **Enable (**checked**).** **Important:** Failing over all resources is required for a CIMPLICITY project resource.

Section	Respond to resource failure
3	If all the restart attempts fail, begin restarting again after the specified period (hh:mm) Optional for a CIMPLICITY project resource.
Pending timeout	
4	Pending timeout (mm:ss) The time the resource takes before the Cluster service puts the resource from Online to Offline and into the Failed state. **Important:** Change the estimated time required for the CIMPLICITY project resource to start if it is different from the Cluster service default. Make sure there is enough time for the CIMPLICITY project to start. Default: 03:00

Warning: Make sure you allow enough time for all CIMPLICITY processes to shut down and restart. Although it is highly recommended that you only include one CIMPLICITY project *(page 102)* in the cluster, if your system requires more than one project, the time needs to be increased accordingly. The projects shut down consecutively, not simultaneously. If enough time is not allowed some processes could still be in the process of shutting down when a restart is initiated.

Advanced Policies Tab

The Advanced Policies tab enables you to select the computers (nodes) that can host the CIMPLICITY project resource.

Status	The node:
Checked	Can host the CIMPLICITY project resource.
Clear	Cannot host the CIMPLICITY project resource.

Parameters Tab (CIMPLICITY project)

rect 2, 50, 379, 305 CIMPLICITY Cluster Configuration Tool for 64-Bit Machines *(page 103)*

A CIMPLICITY tab in the <CIMPLICITY Project Resource Name> Project Properties dialog box is not available on 64-bit machines.

Instead, a Cluster Configuration tool for 64-bit machines/operating systems *(page 103)* is available to configure the CIMPLICITY server properties.

Step 7. Bring the CIMPLICITY Project Resource Online

The CIMPLICITY project resource can be brought online as soon as the parameters are successfully entered in the CIMPLICITY Cluster Configuration Tool for 64-Bit Machines *(page 103)*. These steps are for Microsoft® Windows® Server 2012/2012 R2/2016.

1. Open *(page 93)* the Failover Cluster Manager.
2. Do the following.

A	Right-click the CIMPLICITY project resource that was successfully configured in the Cluster64ServerConfig *(page 103)* dialog box.
B	Select Bring Online on the Popup menu.

The CIMPLICITY project resource is online, associated with IP resources and will be protected in a cluster configuration.

Technical Reference: Cluster Configuration

Technical Reference: Cluster Configuration

Cluster: Best Practices

Cluster failures that cause a failover to occur include:

- Computer (hardware) failure.
- Manual move. (Move this service or application to another node on the CIMPLICITY project resource Popup menu).
- Process health subsystem issue.

In order to take full advantage of cluster benefits, it is recommended that:

- Run a single project only run in a cluster.

- Include an IP address as a dependency so the all of the components in the cluster will move as a group (e.g. The IP address will move with the project).

Points to Consider if running multiple projects on a cluster node

- The single side of the cluster has to have the capability to support all the projects.
- In a cluster configuration the router is a single point of failure; if the router fails all projects in the cluster fail.
- In a single drive set up as a shared drive where you host the projects, the projects are shut down in order on one node before the drive is transferred to the other node in reverse order.

If you have multiple drives set up to host each individual project, then you have dependencies on each individual drive set up for that specific project. The router dying will effectively take them down in order.

- Including an IP address as a dependency is required when there are multiple projects.
- It is essential that you allow enough time for all CIMPLICITY processes to shut down and restart (<CIMPLICITY Project Resource Name> Properties dialog box>Policies tab).

Although it is highly recommended that you only include one CIMPLICITY project in the cluster, if your system requires more than one project, the time needs to be increased accordingly. The projects shut down consecutively, not simultaneously. If enough time is not allowed some processes could still be in the process of shutting down when a restart is initiated.

CIMPLICITY Cluster Configuration Tool for 64-Bit Machines

The Cluster Configuration Tool for 64-Bit Machines tool enables you to perform the CIMPLICITY configuration hat is available in the <CIMPLICITY Project Resource Name> Project Properties dialog box>CIMPLICITY tab for 32-bit machines on 64-bit machines/ operating systems.

The tool enables you to

- Specify the CIMPLICITY project that will be in the CIMPLICITY cluster.
- Access and work in the project through the CIMPLICITY Workbench.

Cluster64ServerConfig Tool: Location

Double-click ..\<CIMPLICITY installation>\extras\Cluster64ServerConfig.exe

A Cluster64ServerConfig dialog box opens.

Cluster64ServerConfig Tool: Configuration

The following information will provide the information so a CIMPLICITY project can be brought online as a cluster resource.

rect 0, 212, 27, 235 *(page 105)*
rect -1, 122, 26, 145 *(page 104)*
rect -1, 31, 26, 54 *(page 104)*
rect 363, 144, 428, 167 *(page 104)*
rect 362, 175, 428, 195 *(page 105)*
rect 360, 230, 429, 253 *(page 105)*

Step	Description
1 *(page 104)*	Cluster Information
2 *(page 104)*	Parameters
3 *(page 105)*	Status

1: Cluster Information

Cluster information includes the following.

- **Cluster Name:** Name of the cluster server selected in the Cluster Failover Manager.
- **Resource Name:** CIMPLICITY Resource ID that was selected in the Cluster Failover Manager.

2: Parameters

The project can be selected and opened in the **Parameters** section.

- Project/Browse Button
- Workbench Button

Project/Browse Button

The CIMPLICITY project for which the cluster is created.

The project entry includes the ..\path\<project name>

Important: The project must be on the shared drive that was selected as a resource dependency.

Do either of the following.

Manual Entry:

Enter the project path and name in the **Project** field.

Browse Button:

1. Click the Browse button to the right of the **Project** field. An Open dialog box opens.
2. Select the connected drive (that was selected as a resource dependency).

   Note: You can access the drive only if you are working on the node that is controlling the resource.

3. Select the project that should be used for the resource.
4. Click Open.

Result: The project name and path will be entered in the Project field.

Workbench Button

Important: You can open the Workbench only if you are working on the node that is controlling the resource.

Work in the Workbench as follows.

Workbench off line

Stop the project the same way you do when it is not in a cluster. The project is taken off line. Configure the project while it is stopped (off line).

Note: The Workbench remains open after the Cluster64ServerConfig dialog box is closed; you can continue configuration.

Workbench online

Start the project the same way you do when it is not in a cluster.

3: Status

The Cluster64ServerConfig tool enables you to see immediately if the entries are valid by clicking the Set Parameter button and viewing the **Status** box.

Set Parameter Button

Click the Set Parameter button to save the CIMPLICITY project name (parameter) to the CIMPLICITY Project Resource.

Status Box

When the Set Parameter button is clicked, results display in the **Status** box.

SUCCESS

If all entries are valid the Status box displays SUCCESS and the Resource Parameter List.

You can confirm the CIMPLICITY project path is set as a cluster Resource in the Failover Cluster Manager.

When the resource is brought online, the cluster is configured.

ERROR

Errors can include any of the following.

- Any entries (Cluster Name, Resource Name or Project Path) are empty.
- Cluster name is invalid.
- Resource name is invalid.
- Project path is invalid.

Chapter 8. Server Redundancy

About Server Redundancy

Congratulations, you've chosen to use CIMPLICITY Server Redundancy as part of your Mission Critical Application. You should completely review and understand this file before getting started with your application.

⚠ **Important:** Projects configured with Server Redundancy will not start without a valid Server Redundancy license. This includes starting in Demo mode.

Topics to review include:

- Server redundancy overview, including:
  - Hardware requirements
  - Application requirements
  - Automatic and manual redundancy
- Redundancy configuration.
- The Redundancy object.
- Recovery procedures.
- Cabling redundancy.
- Network and socket status.
- Supported communication interfaces.
- Configuration parameters.
- Status log messages.
- Problems and solutions.

Levels of Redundancy

- Overview
- PLC redundancy
- Cabling redundancy
- Server redundancy
- Computer network redundancy

Overview

The principle of redundancy in automated systems provides for switchover of functionality to a backup component in case of failure of a primary component. The switchover is considered

automatic if no operator intervention is required. Redundancy applies to both hardware and software, and implies minimal loss of continuity during the transfer of control between primary (active) and redundant (standby) components. Redundant systems reduce single points of failure, preventing loss of functionality.

For cell control systems, the major levels of redundancy include:

- PLC.
- Cabling (PLC LAN or serial connections to server).
- Computer server redundancy.
- Computer networks.

Each level of redundancy provides a failover system that allows continuous system activity with minimal loss of data. The following sections briefly describe each level.

PLC Redundancy

PLC redundancy lets control transfer from a primary programmable controller to a redundant one in case of failure.

When the primary PLC comes back on line, control can be transferred from the redundant PLC back to the primary with minimal loss of data.

The redundancy can be synchronous or independent. Synchronous systems coordinate control and handling of data between CPUs of the active and standby units, while in independent systems each PLC acts like an active unit and is not constrained by the others.

Some CIMPLICITY communication options support PLC redundancy.

Cabling Redundancy

Cabling redundancy *(page 158)* involves separate physical connections to the same device.

The devices can be on a LAN (GENIUS, MAP, etc.) or may require serial connections (SNP, CCM, etc.). Redundant cabling provides an alternate communication path to the device in case of primary path failure. The implementation of cable redundancy with respect to host monitoring/control systems differs with the device protocol involved.

Some CIMPLICITY communication options support cabling redundancy.

Server Redundancy

Server redundancy *(page 113)* involves a primary factory monitoring server and a secondary "Hot Standby" server.

The secondary server is essentially a mirror image of the primary server, running alternate monitoring/control processes and applications. Data collection is performed via independent or shared network paths to the same devices, depending on the protocol. The characteristics of the selected communications protocol(s) determine the details of the configuration.

Upon detection of failure of the primary server, the secondary server can assume control of data collection, alarm functions, applications, and allow user access with minimal loss of continuity. When the primary server comes back on line, control can be transferred back, and the secondary server will resume its backup role.

Computer Network Redundancy

Computer cabling redundancy is similar to cabling redundancy, except it covers computer to computer communications rather than computer to programmable controller. Computer cabling redundancy provides an alternate network path in case of failure of the primary network.

Redundancy Types Supported by CIMPLICITY

CIMPLICITY software supports two types of redundancy:

- Server Redundancy
- Computer Cabling Redundancy

Server Redundancy

Server Redundancy *(page 113)* is fully integrated with CIMPLICITY software's base system functionality, enhancing its already powerful monitoring capability in a full range of computer integrated manufacturing environments.

Computer Cabling Redundancy

CIMPLICITY Computer Cabling Redundancy *(page 158)* provides network redundancy between CIMPLICITY Servers and Viewers. The CIMPLICITY Ethernet traffic travels over both networks in parallel, thus the loss of a single network causes no loss of communications.

Server Redundancy Overview

Server Redundancy Overview

Simply enabling server redundancy for your project provides you with a wealth of redundancy features. However, server redundancy is only a part of your system. The other key parts of your system are your Project, PLCs and the communications network. Combined together these pieces form a mission critical application. Therefore, the application is only as robust as its weakest link. While server redundancy provides many built in features, it cannot repair a faulty network or fix incorrectly written logic. Server redundancy depends on you, the Control Engineer to build a robust environment to enable server redundancy to perform its job.

Review these topics for an overview of the decisions you need to make while designing your mission critical application:

- Server redundancy hardware requirements
- Server redundancy application requirements
- Server redundancy configuration overview
- Automatic redundancy operation overview
- Manual redundancy overview

Server Redundancy Hardware Requirements

Server Redundancy Hardware Requirements

Because the secondary server *(page 113)* in a redundant pair will be set up to run exactly the same functions (except for configuration functions) as the primary server, the secondary server in a redundant pair must be identical to the primary server; that is, the disk, memory, and input/output peripherals should be identical.

Cabling *(page 158)* to devices may place the primary and redundant servers on the same or different cables. The type of cabling used will depend on the requirements of the device. Communications interface software supported by CIMPLICITY Server Redundancy attempts to minimize network traffic to and from the secondary server.

You can connect a device to redundant servers via different cables.

Or, you can connect a device to redundant servers on the same cable.

Review hardware requirements for the:

- Computer
- Network

Computer Requirements for Server Redundancy

1. Steady-State CPU Utilization of Primary, Secondary and viewers is less than 40%.
2. Steady-State Memory Utilization does not require page faulting.

   Note: Using the Windows Performance Monitor observe, the Memory / Pages/Sec Counter. This value should be zero.

3. Use equipment rated for the ambient temperature of your environment.

   Server Redundancy requires that the primary and secondary servers and viewers run a CIMPLICITY supported operating system.

Network Requirements for Server Redundancy

1. Network must be reliable and properly configured.

   Additionally, the following recommendations should be implemented.

2. Primary and secondary servers connected into the same intelligent network switch or hub.

   Note: A large volume of network traffic occurs between the primary and secondary computers. These two computers should be plugged into a network switch that will isolate the inter-server communications from the rest of the network.

3. Steady-State Memory Utilization should be less than 10%.

   Note: Using the Windows Performance Monitor observe, the Memory / Pages/Sec Counter. This value should be 0.

4. Ping times between primary and secondary servers must be less than 10ms, between viewers and servers less than 30ms.
5. Use equipment rated for the ambient temperature of your environment.
6. The servers should not use DHCP unless the leases never expire.

   Additionally, the following recommendations should be implemented.

7. Primary and secondary servers connected into the same intelligent network switch or hub

   Note: A large volume of network traffic occurs between the primary and secondary computers. These two computers should be plugged into a network switch that will isolate the inter-server communications from the rest of the network.

8. Consider using 100mbs Ethernet between the primary and secondary computers.
9. Consider isolating Server to PLC Traffic on a private network segment.

   Server redundancy requires a reliable network, if network reliability is an issue you should consider implementing cabling redundancy between the servers and viewers.

Server Redundancy Application Requirements

Server redundancy provides automatic synchronization of Point and Alarm Databases. Server redundancy provides automatic switch over of CimView application using Points and Alarms. Before you start building your application you should review the section in this documentation entitled "Limitations of Server Redundancy", to verify that the features of CIMPLICITY that you intend on using are supported in server redundancy.:.

CIMPLICITY will run your application as you design it. CIMPLICITY cannot automatically fix your project if you design it incorrectly. Therefore, it is important that you design your project to be mission critical from the ground up. Also, it is imperative that you test your application in a server redundant environment with viewers during the development stage. Only with a properly configured project can you switch on server redundancy and have it work flawlessly.

We, at GE Intelligent Platforms, have designed many redundant systems using CIMPLICITY. We understand the methodology and design techniques needed to build a robust system. Therefore, we do recommend contacting your salesperson to obtain several days of design consultation before you start your first project, and several days of on-site support during deployment.

Scripting Requirements for Server Redundancy

The single biggest issue in building a server redundancy system is your user defined scripts. During failover, point values may be unavailable for a short time. Scripts must be written to properly handle these intermittent periods and to exit cleanly. Scripts that depend on cleanly exiting must be coded to trap the errors that can occur when a point goes unavailable. You must test your scripts during fail over to verify they operate correctly.

Use of Primary / Secondary Computers

The purpose of your primary and secondary computers is to read and process data from your devices, distribute it to viewers, and to remain synchronized. They need available CPU bandwidth to handle exception conditions in your process. If you have viewers in your system, the primary and secondary servers should not run user interface applications such as CimView. The secondary server is not a "spare" computer to be used to perform other chores like word processing, etc. It is a hot backup, dedicated to providing redundancy for your mission critical application.

Important: The primary computer should use a UNC path (e.g. \\COMPUTER\SHARE \) to connect to the secondary computer. It is through this path that a qualified user (a user with administrative privileges) can start and stop the standby.

Note: A UNC path is recommended. A mapped drive may also be used; however, a mapped drive may not be a valid configuration in all situations.

Database Logging Requirements

If you are planning on using database logging, you should certainly read the information in this document on how to use logging within a server redundancy project. Additionally, in a mission critical application, the use of Microsoft Access (As-Is product) as a database is not supported. Instead, Microsoft SQL Server, Oracle, or other supported database server must be used. If you plan on logging a large volume of data you may want to consider locating the database servers on separate computers within the same LAN / switch as the primary and secondary. Remember the total CPU utilization, including the database server, must be less than 40%.

Network Configuration Requirements

In addition to having a solid physical network, server redundancy requires specific network software configuration to be performed on every computer in the system. Since specific configuration is required on every computer you cannot just "plug" another viewer into the network and expect it to work. The network configuration must be updated on the viewer and related computers.

Time Synchronization Requirements

The times on the Primary and Secondary computer must be synchronized. Additionally, if using trending on viewer computers, the times on the viewers must be synchronized with the servers. It is your responsibility to ensure that the computer times are synchronized. There are a variety of commercial products available to maintain time synchronization between computers. If you choose to automatically synchronize your clocks do so at any time other than midnight.

Deployment

Important: : Server redundancy does not support CIMPLICITY deployment .

Server Redundancy Configuration Overview

CIMPLICITY software's Base System Functionality fully integrates Automatic Server Redundancy *(page 109)* . This functionality transfers control from a primary to a secondary server when the primary goes down and, as a result, the connection between the primary and secondary is severed.

Redundant features are integrated into Point Management, Device Communications, User Registration and Alarm Management. The focus of redundancy in CIMPLICITY software centers on:

- Data collection
- Applications driven by these data
- Alarms
- Users accessing these applications

CIMPLICITY also offers the capability for manual redundancy. Manual Server Redundancy *(page 121)* lets control be transferred from a primary to a secondary server, even if the primary is active and the two servers are connected. Transfer capability includes:

- Point management, including data collection
- Entire project control

For CIMPLICITY Server Redundancy, there are two configured computers–the primary server and the secondary server.

A Primary Server is the Server that normally takes the primary role in a redundant configuration. Each Primary Server has one Secondary Server.

A Secondary Server is essentially a mirror image of the Primary Server. It runs the same version of the software as the Primary Server and communicates to the same devices. When the Primary Server fails, the Secondary Server assumes control of the appropriate functions that normally run on the Primary Server. A Secondary Server cannot be a primary configuration node, and does not support any configuration functions.

Server Redundancy Principles of Operation Summary

Important: A user must be logged on with administrative privileges when mapping the drive the standby will be running on. If the user does not have administrative privileges the project will not start on the standby.

In a normal state:

- The primary is in control or is the active server.
- The secondary is the standby server.
- The primary keeps the secondary Alarm, Point and User information synchronized.
- Viewers collect data from the primary computer.

When the primary fails:

- The primary is off line.
- The secondary becomes the active server.
- Viewers collect data from the secondary computer.

When the project on the primary is restarted *(page 153)* :

- The primary obtains Alarm and User information from the secondary and automatically takes over these functions.
- The secondary continues to provide and collect point data for the viewers and the primary for synchronization.

Example

In the following case:

- The primary and secondary are running in a server redundant pair,
- There is a temporary network interruption between the two that exceeds the heartbeat timeouts and retries for the two nodes,
- The primary is restarted,

both the primary and secondary will assert themselves as active.

When the network recovery occurs, the two servers will negotiate to decide which will be the active and which will be the standby. Thus, the primary as the standby, and the secondary will be the active. You cannot choose which of the two nodes in the redundant pair will be the active after a dual active recovery; it will always be the secondary.

After a system manager resets the primary:

- The primary collects point data and takes over point management as well as all other project functions.
- The secondary returns to standby mode.

Automatic Redundancy Operation Overview

Automatic Redundancy Operation Overview

Server Redundancy is configured from within the Workbench on the primary computer. The primary computer most frequently uses a UNC path (e.g. \\COMPUTER\SHARE\) to connect to a secondary computer. The Workbench will automatically distribute the configuration data to the secondary and can control startup / shutdown of the pair.

Principles of automatic server redundancy operation include:

- Server Redundancy principles of operation summary.
- Limitations: Automatic Server redundancy.
- Server redundancy data collection.
- Setpoint use in Server redundancy.

- Database logging in Server redundancy.
- Alarm management behavior in Server redundancy.
- User registration in Server redundancy.
- CimView behavior in Server redundancy.
- Failover period in a Server.

Limitations: Automatic Server Redundancy

Limitations: Automatic Server Redundancy

There are some limitations to automatic server redundancy functionality and failure. Manual server redundancy is a solution for some of these limitations.

Review limitations on:

- Automatic server redundancy functionality.
- Automatic server redundancy failure recovery.

Limitations: Server Redundancy Functionality

1. You may not use the:
    - Multiple Projects feature on the redundant servers
    - Enterprise Server capability

2. The following are not supported:
    - Recipes
    - SPC
    - Tracker

3. Viewers have the following limitations:
    - Fail over is not supported for Viewers in the following cases:
    - BCEUI displays
    - CimView screens with embedded Recipe objects
    - CimView screens with embedded SPC objects
    - Computers that use a Remote Access Server (RAS) or a Wide Area Network (WAN) connection
    - Show Users displays
    - Viewers must have local copies of CimView screens to operate following fail over.

4. The primary server in redundancy must be a development server. (This is a licensing requirement.)

5. If you are accessing at logged data when the primary server fails, you will have to switch to the secondary data source to continue accessing the logged data for Trending.

6. For Trending, point-buffering information is lost on fail over.
7. Configuration changes that cannot be made dynamically require the entire project to be shut down on both computers then be updated and restarted.
8. Dynamic configuration changes can only be made when both computers are running.

   Note: You will have to configure the CIMPLICITY Windows service *(page 144)* if you want to create Logging tables dynamically triggered by an event.

   This includes using an event to trigger scripts that turn on dynamic configuration using object model methods and properties, e.g. CimProject.DynamicMode (property) and CimSystem.OpenSystem (method) .

9. During fail over, device values are not read and setpoints are not written.

Limitations: Automatic Server Redundancy Failure Recovery

CIMPLICITY Server Redundancy will not cover the following failures. Application development for manual server redundancy can frequently circumvent these limitations: Loss of data due to failure of a single component involved in data collection.

If a cable or LAN interface fails, CIMPLICITY software detects the problem, but it will not automatically start collecting data on the secondary server. Under these circumstances, a user may choose to shut down the primary server to allow the secondary server to take over.

When both servers are acting as the Primary server [dual active condition], the Primary Server will go to standby mode and allow the secondary server to take over as active.

Loss of the communications link between CIMPLICITY primary and secondary servers while the primary server is still running.

If the link is lost, both servers will act as the primary server. The secondary server will need to be shut down, and the network repaired. CIMPLICITY software can then be restarted on the secondary server.

Server Redundancy Data Collection

1. Processes point updates from:
   - Device Communication and the Virtual Point Process on the primary server
   - All manual and automatic control functions
2. Sends updates to the secondary Point Manager.

If device communications processes are running on the primary server, the corresponding processes also run on the secondary server.

While the primary server is the active server, the device communication modules on the secondary server operate in standby mode to minimize the impact of redundant data collection on the communications LAN or the programmable controller.

When the primary server terminates, the:

3. Secondary Point Manager automatically begins receiving its updates from
    - The Device Communications and Virtual Point Process on the secondary server
    - All manual and automatic control functions.

4. Device Communications on the secondary server:
    - Establishes full communications with the devices and scans all point values.
    - Reports all point data to the Point Manager.

Important: Applications affected by duplicated point values are not supported.

When the primary server is restarted, resynchronization takes place:

5. The primary server immediately updates user registration and alarm data from the secondary server while it automatically takes over these functions.

6. A CIMPLICITY System manager issues a manual command for the primary server to take over point management and device communication.

7. The primary server:
    - Collects point data from the secondary server
    - Takes control of point management and device communication

8. The secondary server returns to standby mode.

Setpoint Use in Server Redundancy

Users can make setpoint requests on either the primary or secondary server via:

- Point Control Panel
- CimView
- Automatic Control Functions (Event Manager, Custom Programs)

While the primary server is running, all setpoints from the secondary server except those from the Automatic Control Function will be routed to the primary computer. All setpoint originating from Automatic Control Functions on the secondary will be discarded when the primary is in control.

Let's consider the case of the Event Manager. The Event Manager runs on both the primary and secondary computers. Events are triggered on both the primary and secondary computers. All setpoint requests invoked from the action or script tied to the event will be ignored on the standby computer. In other words, your scripts execute in tandem on both computers, but the output to the points is processed only on the active computer.

A Custom Program would be a PTMAP API program written by you that executes as a resident process within CIMPLICITY. Setpoints originating from this program will work the same as the Event Manager.

Database Logging in Server Redundancy

When the primary server is in control, both the primary and the secondary server log alarm and point data into their separate databases. As a result, if the primary fails the secondary computer can continue to log data without loss of information.

When you bring a server back on line after a failure, a **datamerge.exe** utility:

1. Executes a merge from the primary to the secondary server.
2. Executes a merge from the secondary to the primary server.

   See Recovery Procedures in this documentation for more information.

   The ability to conduct an accurate merge begins with your configuration.

   Note: Guidelines for Redundant Logged Database Identification

   When you set up your redundant logged database configuration, you have to make sure that both the primary and secondary servers know where to log their own data. You also have to make sure that the primary server knows where the secondary server is logging data, in case it needs to access the secondary logged database after a failure.

   When setting up redundant logged databases:

3. Set up the same database on the primary and secondary servers so you will have two actual databases that, under normal operation, will be identical.
4. Give the database on each redundant server:
   - The same name as the database on the other server
   - A different name Data Source Name (DSN) from the corresponding DSN on the other server
5. Set up the primary server to point to:
   - Its own database

- The database on the secondary server

6. Set up the secondary server to point to
   - Its own database.
   - The database on the primary server.

   See Server Redundancy Configuration Procedures *(page 126)* in this documentation for configuration details.

   Important: Viewer applications, such as Trending, that use logged data from a server will not fail over to the database on the redundant server.

Alarm Management Behavior in Server Redundancy

The Alarm Manager on the primary server receives its updates from CIMPLICITY services on both the primary and secondary servers. CIMPLICITY applications that generate alarms (Point Management, Event Manageretc.) will not generate alarms when the corresponding application is running on the primary server.

Exceptions to this rule are:

- Device communications alarms.
- Process down alarms.
- Node lost alarms.

User Registration in Server Redundancy

A runtime database for users is maintained by User Registration on the primary server. This information is passed to User Registration on the secondary server.

CimView Behavior in Server Redundancy

CimView applications running on the primary server or viewers receive point updates from the primary point manager. CimView applications running on secondary server receives updates from the point manager running on the secondary server. All setpoints are routed to the primary point manager. When the primary server is lost, CimView applications on viewers automatically begin receiving updates from the secondary point manager.

Important: Trend Controls on CimView screens that use logged data will not fail over to the database on the redundant server.

Failover Period in a Server

CIMPLICITY Interprocess Communications has a built-in probing mechanism, independent of TCP/IP's network probing mechanism. This was introduced so that you can configure a smaller failover period than the TCP/IP default timeout period of 2 hours. This expedites detection of a failed node for Server Redundancy. The default time out is 15 seconds.

Manual Redundancy Overview

Manual Redundancy Overview

Although automatic server redundancy is an essential feature of CIMPLICITY, it requires total failure of the primary server for the secondary server to take over. There are specific failures when you need the secondary server to take over a function or the entire project, even when the primary server has not failed. Therefore server redundancy provides an application interface to allow you to trigger a failover when a specific criteria is reached.

There may be a failure involving the primary server with the:

- Software, when for some reason, the:
- Data collection stops
- Project goes down, even though the server continues to function
- Device communication, when the:
- Device connection to Point Management (PTM) is severed
- All devices, Alarm Manger (AM), User Registration (UR) and Point Management (PTM) applications lose contact with the processes

Principles of manual server redundancy operation include:

- Functions to address specific failures.
- Tools for device failure.
- Point requirements for manual Server redundancy.
- Manual point management transfer (including data collection).
- Manual project transfer forced.

Functions to Address Specific Failures

1. Point management transfer, including data collection
2. Entire project fail over

   For device failure:

3. Point management transfer

4. Entire project fail over

The functions reside in the Redundancy.dll and can be called by any programming language, like the Basic Control Engine, that is capable of calling a DLL entry point.

The functions are:

COR_BOOLEAN failover_project(COR_STATUS *retstat)

Causes the local project to shutdown.

COR_BOOLEAN failover_data_collection(COR_STATUS *retstat)

Causes the current standby computer to become the current active computer.

COR_BOOLEAN redundant_is_redundant()

Tells if this is a redundant project.

int redundant_local_index()

Returns the index of the global point element that has the status of the local device.

Returns	If on the
0	Primary
1	Secondary.

int redundant_remote_index()

Returns the index of the global point element that has the status of the remote device

Returns	If on the
0	Primary
1	Secondary.

You, the system manager, will configure a specific global point and provide the logic to determine when a changeover will occur. Basically the logic can be whatever you want, as long as it is running as part of the project.

Note: Aids that are in your CIMPLICITY directory, if you installed the server redundancy option include:

- Mon_failure.c, a sample program to review as a working example. It is located at:

…\Program Files\Proficy\Proficy CIMPLICITY\api\redundant_api\mon_failure.c

- Redundancy.h, a "C" header file that contains the prototypes for the function. It is located at:

…\Program Files\Proficy\Proficy CIMPLICITY\include\inc_path\redundancy.h

Tools for Device Failure

The devcom toolkit provides the current status of a device connection to Point Management (PTM). Whenever the status of the connection changes the devcom will send a message to Point Management.

Point Management will set a global point based on the status of the device connection.

If there is a failure in the:

- Devcom: All the devices for the devcom are marked unavailable
- Remote PTM: All of the remote devices are marked unavailable
- Local PTM: The application fails over to the remote PTM

The remote PTM marks the local devices as unavailable

A global BOOLEAN array point of two (2) elements indicates the status of the device connection.

The value of...	Indicates that the devcom...
1	Is communicating with the device
0	Is not communicating with the device

Point Requirements for Manual Server Redundancy

1. Have the same name as the name of the device.
2. Be a Boolean point.
3. Have two (2) elements (for example, the status of the device on the primary server and the status of the device on the secondary server).
4. Be a global point.

Manual Point Management Transfer (Including Data Collection)

In a normal state the primary server carries out several processes that can be classified as point management.

Point management includes:

- Data collection

- Virtual point processing
- Sending information to CimView screens

If the primary server stops collecting data from one device, but is still running and communicating with the secondary computer, there is no automatic fail over.

Under these circumstances or for whatever reasons you specify, you can manually transfer point management from the primary to the secondary server.

After the transfer the:

- Primary server maintains control of processes such as:

Software

- Database logging
- Alarm viewer
- Base control engine

Devcom

- Alarm Manager (AM)
- User Registration (UR)
- Point Management (PTM)
- Secondary server takes over point management

todo: To manually transfer point management from the primary to secondary server:

1. Create a specific Boolean point with the same name as the device being monitored.
2. Call this function:

   failover_data_collection()

3. Specify what actions should occur if the point changes from 1 to 0 through a Basic script.

   Example

   Your primary server is connected to a PLC for a conveyor belt called **CB_PLC** and a CimView screen.

   You have configured a global Boolean point called **CB_PLC** that:

   - Monitors the status of the device on the primary server
   - Is on standby on the secondary server
   - Alerts the system manager, if it changes from 1 to 0

The primary server stops collecting data from the CB_PLC device.

The system manager is alerted and switches data collection to the secondary server.

The secondary server takes over point management function.

Manual Project Transfer Forced

In a normal state the primary server is the active server.

The active server controls all the processes in the project.

If the primary server loses contact with one device, but is still running and communicating with the secondary server, there is no automatic fail over.

Under these circumstances or, for whatever reasons you specify, you can manually force a fail over from the primary to the secondary server.

todo: To manually force a project transfer:

1. Create a specific Boolean point with the same name as the device being monitored.
2. Call the function:

   failover_project ()

3. Specify what actions should occur if the point changes from **1** to **0** through a Basic script.

   Example

   Your primary server is connected to a PLC for a conveyor belt called **CB_PLC** and a CimView screen.

   You have configured a global Boolean point called **CB_PLC** that:

   - Monitors the status of the device on the primary server
   - Is on standby on the secondary server
   - Alerts the system manager, if it changes from 1 to 0

   The primary server loses connection with the CB_PLC device.

   The system manager is alerted and fails over the entire project from the primary to the secondary server.

   The secondary server takes over the primary server role.

Server Redundancy Configuration Procedures

Server Redundancy Configuration Procedures

Before you begin configuration, make sure that the same version of CIMPLICITY software is installed and licensed on both servers of each redundant pair. In addition, you must install all required application options, protocols and databases software on both computers.

Review:

- Base System configuration.
- Vista, Windows 7 and Windows Server 2008 extra configuration.
- Database logging configuration.

Base System Configuration

Base System Configuration

Configure the base system in the following order:

Step	Description
1 *(page 126)*	A project.
2 *(page 127)*	A network and verify configuration.
3 *(page 129)*	Device communications.

Important: You need to install the redundancy option on all Viewers.

Note: Global points are obsolete as of CIMPLICITY 5.0. Instead use the points that are created in the redundancy object.

1. Configure a Project for Server Redundancy

1. Select Project on the Workbench menu bar.

2. Select Settings.

   The Project Properties dialog box appears.

3. Select the General tab.

4. Check **Server Redundancy** in the Options box.

5. Select the Redundancy tab.

6. Enter the following information in the Redundancy tab:

Computer name	Enter the name of the secondary Server.
Project path	Enter the directory on the secondary Server where the CIMPLICITY project will be stored. • (Recommended) A UNC path, e.g. \\SERVER2\Redund. **Note:** UNC filenames are supported. • A mapped drive on the primary server. **Note:** A mapped drive may not be a valid configuration in some situations.

   Configuration files and screens are copied from the primary server to the Project path whenever a Configuration Update is performed.

   Note: Make sure you configure the logging setup on both the primary and secondary server through the Database Logger in the CIMPLICITY Workbench.

2. Configure Networks for Server Redundancy

2. Configure Networks for Server Redundancy

The second step when configuring a base system for server redundancy is to configure and verify the network.

Configuration includes host names.

Important: SR requires that all computers (primary, secondary and viewer) must have their names and IP addresses configured and these names must match the actual computer names. You may configure the host names in DNS, WINS or in the local host file on each computer, depending on the networking resources available at your site. SR will not function correctly if this information is not configured. If you do not understand network configuration you should obtain the services of someone that does.

Once the configuration is complete, the following tests should be run.

1. From the primary computer ping primary, secondary and all viewers by name and by address.

2. From the secondary computer ping primary, secondary and all viewers by name and by address.
3. From each viewer, ping primary and secondary by name and address.
4. Verify computer names of each computer match.

Note: Keep alives are automatically configured on a:

- Server when redundancy is installed and
- Viewer when Viewer redundancy is installed.

Ping Example

```
C:\WINNT\system32>ping albsagp2
Pinging albsagp2 [3.26.4.215] with 32 bytes of data:
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
C:\WINNT\system32>ping -a 3.26.4.215
Pinging ALBSAGP2 [3.26.4.215] with 32 bytes of data:
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
Reply from 3.26.4.215: bytes=32 time<10ms TTL=128
C:\WINNT\system32>set computername
COMPUTERNAME=ALBSAGP2
C:\WINNT\system32>
```

To verify names using the above example as a reference.

To Verify Names	Referencing the Above Example
Ping by Name	**ping albsagp2 first translates albsagp2 to an IP** Address and then verifies communication to the computer. **albsagp2** has an IP address of 3.26.4.215. The time required to Ping must be less than 10ms between primary and secondary and less than 30ms between viewers. This step verifies that the network software can convert a hostname to an IP Address.
Ping by Address	Type **ping –a 3.26.4.215 .** The output of **ping albsagp2** provides the IP Address. This step verifies that the network software can convert the IP Address back to the same node name as entered in the first step. If you obtain a different IP Address back this may indicate that you have duplicate entries for the IP Address in you network lookup tables. This must be corrected before continuing.
Continue Pinging	In this example we just ping one computer. You would continue to ping the other computers (secondary, viewers, etc)
Check Computer Names	Type set **computername** to return the current setting. This final step on each computer is determines if **the system's computer name is the same as the computer name configured in the** network. This setting must match the name returned in the above two tests. If not this must be corrected by either changing your computername or changing the network software configuration before continuing.

3. Configure Device Communications for Server Redundancy

3. Configure Device Communications for Server Redundancy

Specific Device Communications configuration such a driver or interface card configuration will need to be configured and tested on the secondary before starting redundancy. Consult the appropriate device communications manual for additional details.

Unsolicited Data

The Device Communications module receives and processes unsolicited data reported from factory devices.

Unsolicited data must be directed to the secondary server in addition to the primary server, so it can be processed by the Device Communications/Point Manager on the secondary server when the primary server fails.

4. Configure Global Points for Server Redundancy

The next step is to configure virtual points to track redundant server status during system operation. The points have the following requirements:

- Naming convention is:
- **ACTIVE_PTM_RP** for the primary server
- **STANDBY_PTM_RP** for the secondary server
- Type is virtual
- Class is Digital
- Calculation for the point is None (default).

A point will take on a value of:

- **1** if the server it represents is currently operating as the primary server
- **0** if the server is the secondary server

The current Primary Point Manager will only change the values. This implies that point updates to the global points will occur when:

- There is a redundant server failure
- Redundant servers are synchronized at startup
- An orderly transition from secondary to primary server occurs.

Important: If you are using point lines in Trending that automatically look for the data source, you must configure **ACTIVE_PTM_RP** and **STANDBY_PTM_RP** . These are the points that Trending needs to failover to the secondary server if the primary is down.

Windows Server 2008 R2 Extra Configuration

Following are the procedures to setup Server redundancy and starting/stopping of remote projects on Windows Server 2008.

Important: You must be a user in the administrators group to set up server redundancy on a remote machine that is part of a Workgroup.

-
- Server Redundancy Setup
-
- On the primary and secondary servers
-
- On the secondary server
-
- On the primary server
- Remote project start from the Workbench

Server Redundancy Setup

On the Primary and Secondary Servers

Do the following on both the Primary and Secondary servers

1. Follow the CIMPLICITY documentation requirements for Server Redundancy to set up the hardware and network, including
    - Hardware.
    - Computer.
    - Network.
2. Create the same user on both servers.

    The user must:

    - Be in the Administrator group
    - Have the same user name on both servers.
    - Have the same password on both servers.
3. Configure the machines to be in the same workgroup.
4. Do the following in the Windows Firewall.

a. Select Start>Control Panel>Windows Firewall.
b. Select the Exceptions tab.
c. Check the following.
- File and Printer Sharing
- Netlogon Service

5. Click OK.

6. Configure the network(s) between the computers as Private.

On the Secondary Server

Note: If you want to start the project on the primary server from the secondary server, you also have to do these procedures on the primary server,

7. Open the Control Panel>Network and Sharing Center.

8. Make sure the following are enabled.
- File sharing
- Password protected sharing

9. Edit the Registry to access a remote **WinRM** service in a workgroup, as follows.

Note: If the account is a local computer member of the Administrators group, the User Account Control (UAC) does not allow access to the **WinRM** service.

Warning: Making changes to the registry is very dangerous and should be done with care.

a. Run regedit.exe or regedt32.exe.
b. Expand the following folders in the Registry Editor left-pane.
a. HKEY_LOCAL_MACHINE
b. SOFTWARE
c. Microsoft
d. Windows
e. CurrentVersion
f. Policies
g. System
a. Right-click **System** in the left pane; select New DWORD (32-bit) Value on the Popup menu.
a. Enter `LocalAccountTokenFilterPolicy` in the new field that displays in the Registry Editor right-pane.
b. Press Enter.
c. Double click `LocalAccountTokenFilterPolicy`
d. Enter 1in the **Value Data** field.
e. Click OK.
f. Close the Registry Editor.

10. Set Remote Registry startup to Automatic, as follows.
    a. Open the Control Panel>Administrative Tools>Services.
    b. Right-click Remote Registry; select Properties.
    c. Select Automatic in the **Startup type** list field.

    On the Primary Server

    Configure project with server redundancy and use a UNC path name for the Project path under the Redundancy tab on the Project Properties.

    Remote Project Start from the Workbench

    The following procedure enables selected users to start a remote project from the Workbench.

    On the Computer that will start a project on a remote computer.

11. Map a drive from remote computer to the local computer to access the project.

12. Grant full access for each administrative user who will perform administrative tasks, including configuration update, dynamic configuration, as follows.
    a. Open a command window.
    b. Enter the following.

    net share /grant:Username1,FULL [/Grant:Username2>,FULL] [/Grant:UsernameN>,FULL]

    Where

Sharename=drive:path	Drive and path of the share
Username	You can enter as few as one name to as many names as should have this privilege. • Username1 is the first name entered for selected users. • Username2 is the second user name for selected users. • UsernameN is the user name with N corresponding to the sequence number for selected users.
FULL	Grants full access.

13. Continue setup using the same procedures that you use to set up *(page 130)* server redundancy.

Database Logging Configuration

Database Logging Configuration

When you set up the same database on the primary and secondary server (so you will have two actual databases that, under normal operation, will be identical) you need to identify them for data logging.

- Redundancy configuration overview.
- Redundancy configuration steps.

Redundancy Configuration Overview

In a redundant configuration:

- The primary and secondary servers log data in parallel to their own database.
- The primary server logs to the database on the primary SQL Server.
- The secondary server logs to the database on the secondary SQL Server.

The only time the primary or secondary server will connect to the other node's database is when the data on the two servers needs to be merged.

- The CIMPLICITY server redundancy operation requires that the:
- Primary server be aware of the secondary SQL Server
- Secondary server be aware of the primary SQL Server

- P = Primary Server
- S = Secondary Server

Redundancy Configuration Steps

In logged database redundancy, configure CIMPLICITY server redundancy, on both the primary and secondary server, through Windows Control Panel..

Step	Description
Step 1 *(page 133)*	Configure the Windows ODBC Data Source Administrator
Step 2 *(page 142)*	Identify the Database Logger data source in a CIMPLICITY project.

Step 1. Configure the Windows ODBC Data Source Administrator

Step 1. Configure the Windows ODBC Data Source Administrator

Note: Viewer applications, such as Trending, that use logged data from a primary server will not fail over to the database on the redundant server.

CAUTION: On the primary server, make sure you have specified the redundant server in the Project Properties *(page 126)* dialog box.

Step	Description
Step 1.1 *(page 134)*	Primary Server: Configure Windows ODBC data sources.
Step 1.2 *(page 138)*	Secondary Server: Configure Windows ODBC data sources.

Step 1.1. Primary Server: Configure Windows ODBC Data Sources

Step 1.1. Primary Server: Configure Windows ODBC Data Sources

Step	Description
Step 1.1.1 *(page 134)*	Display the System DSN Tab.
Step 1.1.2 *(page 135)*	Select the driver for a new data source.
Step 1.1.3 *(page 135)*	Configure the primary data source.
Step 1.1.4 *(page 136)*	Configure the secondary data source.

Step 1.1.1. Display the System DSN Tab

1. Do the following for a:

   32-bit system

   a. Open the Windows desktop on the **primary** Server.
   b. Open the **Administrative Tools> Datasources (ODBC)**> ODBC Data Source Administrator dialog box.

   64-bit system

   • Use the ODBCAD32 utility to:
   a. Open the ODBC data Source Administrator.
   b. Populate the System DSN tab.

2. Select the System DSN tab.

3. Click Add.

The Create New Data Source window opens.

Note: If the data source already exists do the following.

1	Click Configure on the System DSN tab.
2 (page 135)	Continue to configure the Primary Data Source on the Primary Server

Step 1.1.2. Select the Driver for a New Data Source

Do the following.

1. Select the appropriate driver in the **Create New Data Source** list.
2. Click Finish.

Result: An ODBC SQL Server dialog box opens in which you begin to set up the data source.

Step 1.1.3. Configure the Primary Data Source on the Primary Server

- Primary Server/primary data source overview.
- Primary Server/primary data source configuration.

Primary Server/Primary Data Source Overview

The SQL Server storing data from the primary server is the primary data source.

P	= Primary Server
S	= Secondary Server

Primary Server/Primary Data Source Configuration

1. Enter specifications in the Create a new Data Source to SQL Server dialog box as follows:

	Field	Description	Example
A	Name	Unique for the primary server data source.	PRIMARY SQL SERVER
B	Description	(Optional) Description of the data source.	

C	Server	The SQL Server for the primary server. **Note:** Select the SQL Server from the drop down menu.	SQLSERVER1
D	Click **Next**.		

A dialog box opens for authentication information.

2. Do the following.

A	Check: With SQL Server authentication using a login ID and password entered by the user.
B	Enter the following. Login ID: Valid database username. Password: Password associated with the username.
C	Click **Next**.

A dialog box opens for database information.

3. Do the following.

A	Check: Change the default database to the dropdown list of databases that are connected to the selected data source is enabled.
	The dropdown list of databases that are connected to the selected data source is enabled.
B	Select a default database. **Important:** The default database should have the same name as the database ID for the seconadry server.
C	Click Next.

A dialog box opens for language information.

4. Do the following.

A	Check: Perform translation for character data.
B	Click **Finish**.

An ODBC Microsoft SQL Server Setup screen displays the details of your configuration.

5. Click Test Data Source.

The data source connection and settings are tested.

6. Click **OK** if the specifications are correct.

The SQL Server data source (e.g. PRIMARY SQL SERVER) is created and displays in the Data Source list on the System DSN tab.

Step 1.1.4. Configure the Secondary Data Source for the Primary Server

- Primary Server/secondary data source overview.
- Primary Server/secondary data source configuration.

Primary Server/Secondary Data Source Overview

The SQL Server storing data from the secondary server is the secondary data source.

The primary server must be aware of this data source. However, the only time the primary or secondary server will connect to the other node's database is when the data on the two servers needs to be merged.

P	= Primary Server
S	= Secondary Server

Primary Server/Secondary Data Source Configuration

1. Enter specifications in the Create a new Data Source to SQL Server dialog box as follows:

	Field	Description	Example
A	Name	Unique for the secondary server data source.	SECONDARY SQL SERVER
B	Description	(Optional) Description of the data source.	
C	Server	The SQL Server for the secondary server. **Note:** Select the SQL Server from the drop down menu.	SQLSERVER2
D	Click **Next**.		

   A dialog box opens for authentication information.

2. Do the following.

A	Check	With SQL Server authentication using a login ID and password entered by the user.
B	Enter the following.	
	Login ID	Valid database username.
	Password	Password associated with the username.
C	Click **Next**.	

   A dialog box opens for database information.

3. Do the following.

A	Check	Change the default database to:

	The dropdown list of databases that are connected to the selected data source is enabled.	
B	Select	A default database **Important:** The default database should have the same name as the database ID for the primary server.
C	Click Next.	

A dialog box opens for language information.

4. Do the following.

A	Check	Perform translation for character data.
B	Click **Finish**.	

An ODBC Microsoft SQL Server Setup screen displays the details of your configuration.

5. Click Test Data Source.

The data source connection and settings are tested.

6. Click **OK** if the specifications are correct.

The SQL Server data source (e.g. SECONDARY SQL SERVER) is created and displays in the Data Source list on the System DSN tab.

Step 1.2. Secondary Server: Configure Windows ODBC Data Sources

Step 1.2. Secondary Server: Configure Windows ODBC Data Sources

Step	Description
Step 1.2.1 *(page 138)*	Display the System DSN Tab.
Step 1.2.2 *(page 139)*	Select the driver for a new data source.
Step 1.2.3 *(page 139)*	Configure the primary data source.
Step 1.2.4 *(page 141)*	Configure the secondary data source.

Step 1.2.1. Display the System DSN Tab

1. Do the following for a:

   32-bit system

   a. Open the Windows desktop on the **primary** Server.
   b. Open the **Administrative Tools> Datasources (ODBC)**> ODBC Data Source Administrator dialog box.

   64-bit system

   • Use the ODBCAD32 utility to:
   a. Open the ODBC data Source Administrator.
   b. Populate the System DSN tab.

2. Select the System DSN tab.

3. Click Add.

The Create New Data Source window opens.

Note: If the data source already exists do the following.

1	Click Configure on the System DSN tab.
2 *(page 139)*	Continue to configure the primary Data Source on the secondary Server.

Step 1.2.2. Select the Driver for a New Data Source

Do the following.

1. Select the appropriate driver in the **Create New Data Source** list.
2. Click Finish.

Result: An ODBC SQL Server dialog box opens in which you begin to set up the data source.

Step 1.2.3. Configure the Primary Data Source on the Secondary Server

- Secondary Server/primary data source overview.
- Secondary Server/primary data source configuration.

Secondary Server/Primary Data Source Overview

The SQL Server storing data from the primary server is the primary data source.

The secondary server must be aware of this data source. However, the only time the primary or secondary server will connect to the other node's database is when the data on the two servers needs to be merged.

P	= Primary Server
S	= Secondary Server

Secondary Server/Primary Data Source Configuration

1. Enter specifications in the Create a new Data Source to SQL Server dialog box as follows:

	Field	Description	Example
A	Name	Unique for the primary server data source.	PRIMARY SQL SERVER
B	Description	(Optional) Description of the data source.	
C	Server	The SQL Server for the primary server. **Note:** Select the SQL Server from the drop down menu.	SQLSERVER1
D	Click **Next**.		

A dialog box opens for authentication information.

2. Do the following.

A	Check	With SQL Server authentication using a login ID and password entered by the user.
B	Enter the following.	
	Login ID	Valid database username.
	Password	Password associated with the username.
C	Click **Next**.	

A dialog box opens for database information.

3. Do the following.

A	Check	Change the default database to:
	The dropdown list of databases that are connected to the selected data source is enabled.	
B	Select	A default database **Important:** The default database should have the same name as the database ID for the secondary server.
C	Click Next.	

A dialog box opens for language information.

4. Do the following.

A	Check	Perform translation for character data.
B	Click **Finish**.	

An ODBC Microsoft SQL Server Setup screen displays the details of your configuration.

5. Click Test Data Source.

The data source connection and settings are tested.

6. Click **OK** if the specifications are correct.

The SQL Server data source (e.g. PRIMARY SQL SERVER) is created and displays in the Data Source list on the System DSN tab.

Step 1.2.4. Configure the Secondary Data Source for the Secondary Server

- Secondary Server/secondary data source overview.
- Secondary Server/secondary data source configuration.

Secondary Server/Secondary Data Source Overview

The SQL Server storing data from the secondary server is the secondary data source.

P	= Primary Server
S	= Secondary Server

Secondary Server/Secondary Data Source Configuration

1. Enter specifications in the Create a new Data Source to SQL Server dialog box as follows:

	Field	Description	Example
A	Name	Unique for the secondary server data source.	SECONDARY SQL SERVER
B	Description	(Optional) Description of the data source.	
C	Server	The SQL Server for the secondary server. **Note:** Select the SQL Server from the drop down menu.	SQLSERVER2
D	Click **Next**.		

A dialog box opens for authentication information.

2. Do the following.

A	Check: With SQL Server authentication using a login ID and password entered by the user.
B	Enter the following: • Login ID: Valid user name. • Password: Password associated with the user name.
C	Click **Next**.

A dialog box opens for database information.

3. Do the following.

A	Check	Change the default database to the dropdown list of databases that are connected to the selected data source is enabled.
B	Select	A default database **Important:** The default database should have the same name as the database ID for the primary server.
C	Click Next.	

A dialog box opens for language information.

4. Do the following.

A	Check	Perform translation for character data.
B	Click **Finish**.	

An ODBC Microsoft SQL Server Setup screen displays the details of your configuration.

5. Click Test Data Source.

The data source connection and settings are tested.

6. Click **OK** if the specifications are correct.

The SQL Server data source (e.g. SECONDARY SQL SERVER) is created and displays in the Data Source list on the System DSN tab.

Step 2. Identify the Database Logger Data Source in a CIMPLICITY Project

Step 2. Identify the Database Logger Data Source in a CIMPLICITY Project

1. Click Project on the Workbench menu bar.
2. Select Properties.

   The Project Properties dialog box opens.

3. Select the Settings tab.
4. Select Database Logger.
5. Click **Settings**.

   The Logging Properties dialog box opens:

6. Select the Point Connection - Primary tab.
7. Fill in the fields in the Logging Properties dialog box as follows.

Field	Description
ODBC data source	The data source for the server from the drop down list of available data sources. **Tip:** Click the **ODBC Data Source** button to the right of the **ODBC data source** field to open the ODBC Data Source Administrator dialog box. You can then see the drivers configured for each data source and make any necessary changes or additions.
Database user	(Required if you are connecting to a SQL Server) user who has the privilege to connect to the selected database driver.
Password	(Required if you are connecting to a SQL Server) needed to connect to the selected database driver.
Reconnect wait period	Amount of time the Database Logger waits between reconnect attempts when the connection to the database is lost in the. Enter a value between 0 seconds (continuous retries) and 24 hours. The default is 30 seconds.
Enable Store and Forward	Checked enables Store and Forward. After you enable the feature, check one of the following: • Unlimited: Database Logger stores an unlimited number of records while its connection to the database is down. The number of records actually stored is determined by the amount of time the connection is lost and by the amount of free disk space you have. • Max number of stored records: Database Logger stores a specified number of records when its connection to the database is down. Enter a number between 1 and 4,294,967,295.

8. Repeat these steps for the other three tabs so you will have configured all four tabs:

Tab		Server
Point Connection	Primary	Primary (Primary)
Point Connection	Secondary	Second (Secondary)
Alarm Connection	Primary	Primary

Alarm Connection	Secondary	Second

9. Click OK or select the Parameters tab.

CIMPLICITY validates your entries. If the Data Logger is unable to connect to the selected database, validation fails.

Important: On each tab, make sure that you select the correct data source for the computer (Primary / Secondary) that the tab represents.

2.1. Configure a Service for Creating Logging Tables Dynamically Triggered by an Event

Note: A script can be run to create logging tables dynamically triggered by an event.

However, for this to work in a redundant environment, a CIMPLICITY service has to be configured to log on as an account that has access to both the primary and the secondary computer.

CIMPLICITY uses file access calls to determine if the project is running on the secondary.

If a service is not configured this will fail because the test is being done by the CIMPLICITY service. An error message will indicate Project must be running on secondary computer to use dynamic configuration. even if the project is running on both servers.

1. Open the Windows Control Panel.
2. Click **Administrative Tools> Services**.

   The Services dialog box opens.

3. Double-click the CIMPLICITY HMI Service.

   The CIMPLICITY Service Properties dialog box opens.

4. Select the Log On tab.
5. Fill in the fields as follows.

Field	Description
This account	Enabled when checked. CIMPLICITY user ID that can access CIMPLICITY projects on both servers.

Password	User password to access CIMPLICITY.
Confirm password	Repeat of password.

6. Click OK.

The CIMPLICITY service can now log on to both computers. Therefore, a script can now be used to dynamically create logging tables which is triggered by an event.

2.2. Guidelines and Notes about Specific Data Sources

CIMPLICITY SQL Server Logging

A Microsoft SQL Server data source that logs data to an on-node SQL Server database. You must install SQL Server (sold separately) to use this data source.

If you are connecting to a SQL Server, you may be prompted for a database name during validation.

Oracle Database

You may see the ODBC data source that you created for Oracle.

You may be prompted for a Server ID during validation. Enter the Alias Name for the Oracle database in this field.

Redundancy Object

Redundancy Object

When you activate the server redundancy option in the Workbench, CIMPLICITY automatically installs a Redundancy object, which is an object of the redundancy class.

The Redundancy object enables you to easily:

- View whether or not the primary and/or secondary server is running,
- Switch the active role from one server to the other when you need to take the current active offline,
- Switch the active back when the original active is brought back on line and
- Configures a set of point to use in your application.

This capability enables you to efficiently switch control back and forth while ensuring that data is not lost.

Note: The class object and screen are automatically created when Redundancy is enabled.

Section	Description
1	Class objects
2	Automatically created Redundancy object.
3	Automatically created GefRedundancy CimView screen.

Redundancy Object Components

1. GefRedundancy class.
2. Redundancy object.
3. CIMPLICITY Redundancy object points as follows:

REDUNDANCY.PRI_ACTIVE	Primary computer is active. (i.e. CIMPLICITY is running on the primary computer.)
REDUNDANCY.RESTORE_PRIMARY	When set to **1**(by a button on the Redundancy CimView screen), the active is switched to the primary computer.
REDUNDANCY.SEC_ACTIVE	Secondary computer is active. (i.e. CIMPLICITY is running on the secondary computer.)
REDUNDANCY.SWITCH_TO_SEC	When set to **1**(by a button on the Redundancy CimView screen), the active is switched to the secondary computer.

4. GefRedundancy CimEdit/CimView screen.

   Note: You do not have to do any configuration for the redundancy class and object. CIMPLICITY does it all for you.

Redundancy Object Use

Redundancy Object Use

The Redundancy object is straightforward to use.

Steps to use the Redundancy object are:

Step	Description
Step 1 *(page 147)*	Display the Redundancy CimView screen.
Step 2 *(page 147)*	Monitor the servers through the Redundancy screen.
Step 3 *(page 147)*	Switch the active role between redundant computers.

Step 1. Display the Redundancy CimView Screen

1. Make sure the project is running on the local computer.
2. Do the following.

A	Select **Project>Objects** in the Workbench left pane.
B	Right-click the Redundancy object in the Workbench right pane.
C	Select Quick View from the Popup menu.

The Redundancy CimView screen appears.

Step 2. Monitor the Servers through the Redundancy Screen

Once the Redundancy CimView screen displays for the project, you can review which:

- Computers are running.
- Computer is the active.
- Computer is the standby.

1. PRIMARY1 is the active.
2. SECOND1 is the standby.

Step 3. Switch the Active Role between Redundant Computers

Switch the active role from one computer to the other.

The secondary computer the active and the primary computer is offline.

1. View the Redundancy CimView screen the secondary computer.

The secondary computer displays as the active; the primary computer displays as offline.

2. Bring the primary computer back online.
3. Restart the CIMPLICITY project.
4. View the Redundancy object on the primary computer.

   The primary computer displays as the standby and online.

5. **Switch** roles.

   The primary computer is the active; the secondary computer is the standby.

Recovery Procedures

Recovery Procedures

- Normal operating procedures.
- Primary Server failure.
- Failure exceptions for automatic Server redundancy.

Normal Operating Procedures

Normal Operating Procedures

Normal operating procedures in server redundancy involve:

- Run and stop redundant CIMPLICITY projects.
- Start a project from the secondary server.
- Configure the project to start at boot.

Start and Stop Redundant CIMPLICITY Projects

When all hardware is working correctly, CIMPLICITY software can be started and shut down using the **Run** and **Stop** tools in the CIMPLICITY Workbench on the primary server.

Important: Under most circumstances you should use the Workbench to start redundant projects. This is because the Workbench:

1. Allows you to start up both systems in one coordinated action. (If you use CIMPLICITY Options to startup on the primary, you need to wait for the primary to finish its startup and then start the secondary.)

2. Will update the standby node configuration as required before starting the project, making sure that the active and standby nodes are synchronized.

   A rare exception to normal startup occurs if, there is a catastrophic event that forces both computers to shut down, e.g. the power failed. In this situation, the last active server must be the first restarted to ensure data integrity in the following areas:

   - Manual mode data and values.
   - Saved point values.

   If, the secondary server was the only computer running before shutting down, it should start first, as the active Data that was collected before the shutdown can then be failed over to the primary server before it is reinstated as the active.

   Tip: If one project is running and one is stopped both the **Run** and **Stop** buttons on the Workbench toolbar are active . Click either button to determine which project is running.

   Start a Redundant Project

3. Do one of the following:

   Method 1

   a. Click Project on the Workbench menu bar.
   b. Select Run.

   Method 2

   Click the **Run** button on the Workbench toolbar.

   The Redundant project stop dialog box opens when you use either method.

   The buttons are dimmed for servers that are running.

4. Select Run.

   The Start Redundant Project dialog box opens.

5. Select one of the following:

Option	Starts the project:
Primary & Secondary	On both the primary and secondary servers
Primary only	Only on the primary server

Secondary only	Only on the secondary server

6. Click Start to start the project.

   Stop a Redundant Project

7. Do one of the following:

   Method 1

   a. Click Project on the Workbench menu bar.
   b. Select Stop.

   Method 2

   Click the **Stop** button on the Workbench toolbar.

   The Stop Redundant Project dialog box opens.

   Note: The buttons are disabled for servers that are not running.

8. Select one of the following:

Option	Stops the project:
Primary & Secondary	On both the primary and secondary servers
Primary only	Only on the primary server
Secondary only	Only on the secondary server

9. Click Stop to stop the project.

Start the Project from the Secondary Server

1. Open the CIMPLICITY Options dialog box.
2. Select the project file from the project directory on the secondary computer
3. Click **Start**.

Configure the Project to Start at Boot

The project can be configured to start on both the primary and secondary computers when they power up.

Important: Make sure that the projects do not start at the same time if they are configured to start on both the primary and secondary computers when they power up. Failure to ensure this can result in both computers considering themselves the active server.

To provide a mechanism for dealing with a **Power On** situation where both computers boot at the same time you can configure a parameter to delay the secondary computer's startup until the primary is complete.

To delay the secondary computer's startup until the primary is running, configure the following global parameter in the project:

STARTUP_STARTUP_TIMOUT|1|< TIME-MINUTES >

Where

Time-minutes is the amount of time it takes for the project to start on the primary computer plus an additional minute.

This value you empirically determine by measuring the startup time on the active server.

Note: It is recommended that you use the Workbench to start redundant projects.

Primary Server Failure

Primary Server Failure

When the server fails in automatic server redundancy, the secondary server automatically takes control.

Review:

- System operation during failover.
- Methods to detect the cause of primary Server failure.
- Reset the primary Server after recovery.
- Re-synchronize (Datamerge) database logging files.

System Operation during Fail Over

When the primary server of a redundant pair fails, the secondary server goes from standby to active mode to insure that all essential areas in the project continue to operate.

Areas include:

- Device communications and point management.
- Alarm management.
- User logons.
- Runtime interfaces.

Device Communications and Point Management

Device Communications on the secondary server begins actively polling for data and passes point data to the Point Manager on the secondary server, which now becomes the primary Point Manager. Any viewer process that was connected to the original primary Point Manager will automatically switch over to the new primary Point Manager, which will now assume all supervisory and control functions.

Note: Between the time that the primary server is lost and before the secondary server takes over, users may notice an interruption in system performance. During this time, point values will not be updated, setpoints and alarm acknowledgments will not complete, and users will not be able to log on or off.

Alarm Management

The Alarm Manager on the secondary server becomes the primary Alarm Manager. No alarm data is lost because the Alarm Manager on the primary server continually updated the alarm list for the Alarm Manager on the secondary server. The new primary Alarm Manager will now process all alarm updates and provide alarm information to all interested processes.

User Logons

The User Registration process on the secondary server becomes the primary User Registration process. No user registration data is lost because User Registration on the primary server continually updated the user list for User Registration on the secondary server. The new primary User Registration will now process all user logins and logouts and provide information on user views.

Runtime Interfaces

When the primary server fails, all CimView, Alarm Viewer and Point Control Panel sessions running on that server are lost.

- CIMPLICITY user interface accessed from console on the primary server.

When the primary server fails, this console is no longer usable. The user will have to move to the console on the secondary server, log in, and access the CIMPLICITY user interface from there.

- CIMPLICITY user interface accessed from a Viewer.

The user interfaces will fail over to the new active.

Methods to Detect the Cause of Primary Server Failure

Server failure may be due to:

- CIMPLICITY software shutdown on either server
- Network failure between the primary and secondary servers
- Loss of server due to loss of power or equipment failure

Server failure is detected by the IPC Router, which is the communications process that runs on each server.

1. The Router sets up links to each server in the system and sends messages to each node at a set interval.

   The probe interval is defined in REDUND_PROBE_DELAY that is set to 3000 millisecond by default.

2. If no reply is received from the server for a set number of tries (defined in REDUND_PROBE_COUNT) the server is then declared to have failed.

3. The Router sends a Partner Dead message to any processes that have outstanding messages to processes on that server.

   Server failure may be detected on a primary server, secondary server or Viewer.

   - When a secondary server fails, functionality is not lost, because all functions are also running on the primary server.
   - When a primary server fails, the secondary server initiates procedures to take over redundant CIMPLICITY functions.

Reset the Primary Server after Recovery

After a primary server has failed and recovered, processes on the secondary server need to be told that the primary sever is now available. The Redundancy object provides a straightforward way to reset the primary server. CIMPLICITY automatically displays this object when the redundancy feature is activated.

todo: To reset the primary server on CIMPLICITY:

1. Make sure that CIMPLICITY software is running on the primary and secondary servers.

2. Start a project's Workbench on the primary server.

3. Select the **Objects** icon in the Workbench left pane.

4. Right-click the Redundancy object.

5. Select Quick View from the popup menu.

The Redundancy screen appears displaying the primary computer as the standby and the secondary as the active.

6. Click **Switch**.

The primary server is reset to be the active.

Note: The Alarm Manager and User Registration on the primary and secondary servers will automatically resynchronize themselves to their primary and secondary roles when the primary server initially comes on line.

The following will occur as part of the reset:

- Device communications modules on the secondary server will stop collecting data and return to standby mode.
- The Point Manager on the secondary server resumes its secondary role.
- All viewer applications will automatically resynchronize to the primary Point Manager.

The Point Manager on the primary server will resume its primary role, and will initiate device communications modules on the primary server to start collecting data.

Re-synchronize (Datamerge) Database Logging Files

- Re-synchronize (Datamerge) overview.
- Datamerge command
- Datamerge command example
- Datamerge process

Re-synchronize (Datamerge) Overview

The Database Logger uses ODBC database tables to store historical data. For Server redundancy, the same database tables are created on the primary and secondary servers.

Time synchronization works as follows:

One of the redundant servers goes down.

Two ptnr_<timestamp>.log file are generated in the project's \log directory.

- One on the primary server
- One on the secondary server

With <timestamp> as the date and time the file was created the ptnr_<timestamp>.log records on the:

Server	Clocked time

Secondary	When one of the servers went down and came back up.
Primary	When one of the servers went down and came back up.

Note: If there is no data with the exact time stamp that is at the exact End time stated in the destination database , the Datamerge utility will:

- Search for the next latest logged data in the destination and
- Merge to that point.

Datamerge will automatically use the found next latest logged data as the END time even though an End time was specified at the command line or via the ptnr* files.

These files are then used to synchronize the databases.

Datamerge Command

On the primary server:

1. Open the project's Workbench.
2. Click Tools>Command Prompt on the Workbench menu bar.

   The Command window opens.

3. Continue with the datamerge command as follows.

A	**Note:** The command prompt should display the path to the selected project. a. Type the following. `datamerge /?` a. Press Enter.	
B	A description of the DATAMERGE.EXE command displays in the Command window.	
C	Enter the datamerge.exe command with parameters to merge specific times. The command format is: `DATAMERGE.EXE SOURCE DEST TIME1 TIME2` Where	
	SOURCE	Source server [PRIMARY | SECONDARY]
	DEST	Name of the destination server [SECONDARY | PRIMARY]
	TIME1	Start time for merging data [`dd-Mmm-yyyy hh:mm:ss`]
	TIME2	End time *(page 155)* for merging data [`dd-Mmm-yyyy hh:mm:ss`]

Important:

- The month (only the month) has the first letter capitalized.
- PRIMARY and SECONDARY must be capitalized.
- Use either PRIMARY or SECONDARY (as KEYWORDS) for the SOURCE and DEST servers.
- If no arguments are supplied, no merge occurs.

CAUTION: `Datamerge` must include command line options for both the primary and secondary servers if either the primary or the secondary server is restarted without the other.

Examples

- The Primary Only or Secondary Only server is checked on the Start Redundant Project *(page 149)* dialog box and the Start button is clicked.
- The Primary Only or Secondary Only server is checked on the Stop Redundant Project *(page 150)* dialog box and the Stop button is clicked.
- Data is logged to one server and not the other.

`TIME1` and `TIME2` must be exactly the same for both.

Datamerge Command Example

Following are examples of the datamerge.exe command.

```
datamerge PRIMARY SECONDARY 23-Jan-2007 21:00:00 25-Jan-2005 14:00:00
```

or

```
datamerge SECONDARY PRIMARY 23-Jan-2007 21:00:00 25-Jan-2005 14:00:00
```

Datamerge Execution Process

Following is how datamerge.exe is executed from the primary to the secondary server.

As it executes, the **datamerge.exe** utility:

4. Type the following.
   `datamerge /?`
5. Press Enter.
6. Reads the **ptnr_** < timestamp > **.log** files on the primary and secondary servers.
7. Determines from the **ptnr_** < timestamp > **.log** files in the primary server's **\log** directory what data needs to be merged from the primary server's database to the secondary server's database.
8. Executes the merge from primary to secondary.
9. Determines from the **ptnr_** < timestamp > **.log** files in the secondary **\log** directory what data needs to be merged from the secondary server's database to the primary server's database
10. Executes the merge from secondary to primary.

    A **db_merge.log** file is generated to report the success or failure of the merge.

Note:

- The process from the secondary to primary server merges the files from the secondary server to the primary server first.
- When you run the datamerge.exe utility with specific start and end times, the ptnr_timestamp.log files on the secondary and primary servers are not used.

Failure Exceptions for Automatic Server Redundancy

Failure Exceptions for Automatic Server Redundancy

There are two categories of failure exceptions that CIMPLICITY automatic Server Redundancy will not handle:

- Process failures.
- Network failures.

Process Failures

A process failure occurs when a single process on a server fails. If this occurs on a primary server, the recovery method depends on which process failed.

- If the Alarm Manager or User Registration on the primary server fails, control automatically passes the corresponding process on the secondary server. If the process on the primary server is restarted (via cpc), control will automatically pass back to the process on the primary server.
- If a device communications process on the primary server fails, control will not pass to the corresponding process on the secondary server, and point data will be lost.
- If the Point Manager on the primary server fails, control automatically passes to the Point Manager on the secondary server.

Recover from process failure in automatic redundancy

1. Click Project on the Workbench menu bar.
2. Select one of the following.

Stop	Shuts down the project on the primary server.
Run	Restarts the project on the primary server.

3. When the project is running, reset the primary server to be the active.

   Circumvent limitations of process failure in automatic redundancy

Use either of two manual redundancy functions to anticipate and deal with this possibility.

Network Failures

1. On each secondary server, use **Stop** to shut down the CIMPLICITY project.
2. Repair the network.
3. After the network is restored, use **Start** to bring the project on the secondary server back online.

Note: The above procedure assumes that the CIMPLICITY project is still running on the primary server. If the project has been shut down, then use the normal startup procedures to restart the project on both the primary and secondary server.

Computer Cabling Redundancy

Computer Cabling Redundancy

You can use CIMPLICITY Computer Cabling Redundancy to create a redundant cabling configuration for your CIMPLICITY for Windows Servers and Viewers.

In a network with CIMPLICITY Computer Cabling Redundancy, you can have two types of computers; computers with single IP addresses and computers with dual IP addresses.

- Computers with dual addresses–can continue to communicate with other computers with dual IP addresses even if one of the Ethernet network connections is lost.
- A computer with a single IP address– can communicate with a computer with dual IP addresses. If the Ethernet connection that the computer with the single IP address is using is lost, then all communication to the dual IP address computer will be lost.

When a CIMPLICITY project detects that it is sending a message to a computer that has Computer Cabling Redundancy, the project sends duplicate messages to each IP address. CIMPLICITY software on the receiving computer processes the first message and deletes the second one.

Review:

- Cabling redundancy operation rules.
- Cabling redundancy limitations.
- Cabling redundancy hardware requirements.
- Cabling redundancy supported network configurations.

- Cabling redundancy configuration procedures.

Cabling Redundancy Operation Rules

The following rules describe the overall operation of a computer with Computer Cabling Redundancy:

- When a loss of a network is detected Ethernet traffic continues on the other network.
- When the network is repaired the computers will re-establish communication automatically, within 45 seconds.
- If both networks are lost then CIMPLICITY software acts as if communication was lost to the project. CIMPLICITY software will re-establish the connections to the other computer when the network is repaired.

Cabling Redundancy Limitations

There are some limitations when using computer cabling redundancy that you need to be aware of before you implement it.

Functionality Limits

The following functionality limitations apply for Computer Cabling Redundancy:

- If you are viewing logged data remotely with the Trend Control and the Ethernet cable that ODBC is currently using is lost then the Trend Control will stop updating the logged data.
- If you are logging to a remote database and the Ethernet cable that ODBC is currently using is lost then some data may not be logged.
- Non-CIMPLICITY applications might experience interruptions as they fail over to the remaining Ethernet cable.
- It is not possible to have two Ethernet cards installed and only use one of them for CIMPLICITY software.

Failure Recovery Limitations

CIMPLICITY Computer Cabling Redundancy will not cover the following failures:

- Loss of both network connections.
- Rapid, alternating loss of both network connections, where the time period is less than 45 seconds.

Cabling Redundancy Hardware Requirements

CIMPLICITY Computer Cabling Redundancy supports two network interface cards (NIC) in each computer.

Each card must be configured for a different

- IP network and
- Physical Network

Different IP Networks for Cabling Redundancy

There are three different classes of IP Networks.

- A,
- B and
- C

If you configure one card for an **A** class IP network and the other for a **C** class, by default they will be connected to different IP networks.

You can configure each card to the same class IP network. However, you have to make sure that they are, in fact, different IP networks.

Example

You have a Class **C** IP address **192.68.1.135.**

Where

192.68.1 specifies the network that is a Class C status.

135 is a host (specific PC) in the network.

You can configure the second card with a Class C IP address by simply changing the number that is unique to the address.

The Class C IP address may now be **192.68.2.243.**

Where

192.68.2 specifies a second network that is a Class C status.

243 is a host (specific PC) in the second network.

You now have two distinct IP networks.

Note: You can use any subnet mask for two IP addresses. However, you cannot use a single subnet mask to differentiate the networks. You have to use a different IP network number for each IP address, disregarding the subnet mask.

Different Physical Networks for Cabling Redundancy

Each of two network interface cards must be connected to its own physical network. This is needed to provide for a backup network in the case that one of the networks is lost.

Cabling Redundancy Supported Network Configurations

The supported network configurations are:

- Non-redundant server to non-redundant viewer.
- Redundant server to non-redundant viewer.
- Non-redundant server to redundant viewer.
- Redundant server to redundant viewer.
- Non-redundant server to non-redundant server.
- Redundant server to redundant server.

Cabling Redundancy Configuration Procedures

Note: Make entries on the Hosts tab and Network tab in the CIMPLICITY Options dialog box. Entries on the Hosts tab are written to the CimHosts.txt file. When CimHosts.txt is edited through the Hosts tab, comments that have been written:

- In the body of the file or at the end are deleted.
- At the beginning of the file are not deleted.

Failover Rate for Cabling Redundancy

The default value for detecting that a network connection has been lost is 20 seconds. Due to the seamless nature of CIMPLICITY Computer Cabling Redundancy, this should be a reasonable default. It can be modified, depending on the needs of the application. The loss detection rate should never be modified to less than 3 seconds.

The failover period is defined as:

PING_INTERVAL * (PING_COUNT + 1)

The two parameters, **PING_INTERVAL** and **PING_COUNT** are defined in the **cimhosts.txt** file.

The formats for these parameters are:

#PING_INTERVAL < seconds >

#PING_COUNT < count >

where

<seconds > is the number of seconds between probe attempts and <count> is the number of probes to make.

Example

Sample entries for these parameters would be:

#PING_INTERVAL 2

#PING_COUNT 10

A related parameter is **CONNECT_TIMEOUT**. This is the number of seconds to wait after forming the TCP/IP connection for the initial data to arrive from the other computers. The default value is 10 seconds. There should be no reason to change this value.

Diagnostic Output for Cabling Redundancy

The CIMPLICITY Computer Cabling Redundancy option can generate diagnostic output that you can use to track down problems with the functioning of the cabling redundancy system. To generate diagnostic output, enter a valid value for the DEBUG parameter in the **cimhosts.txt** file.

The format for this parameter is:

#DEBUG < flags >

where

< flags > is a value used to control what types of diagnostic output to generate.

You can add any of the following values together to form the < flags > value:

Value	Output
1	Print errors
2	Print infrequent calls
4	Print all Winsock calls
8	Print all transactions

TCP/IP Port for Cabling Redundancy

To support CIMPLICITY Computer Cabling Redundancy, you need to use a set of TCP/IP ports in the range 5000 to 6000. Depending on other communication software you are running you might have to alter this range. This is controlled by the **START_PORT_RANGE** and **NUMBER_OF_PORTS** parameters in the **cimhosts.txt** file.

The formats for these parameters are:

#START_PORT_RANGE < port >

#NUMBER_OF_PORTS < count >

where

< port > is the TCP/IP port to start with and < count > is the number of ports to use.

Example

Sample entries for these parameters would be:

#START_PORT_RANGE 5000

#NUMBER_OF_PORTS 1000

Monitoring Network and Socket Status

Computer Cabling Redundancy Monitoring

The Computer Cabling Redundancy Monitoring API allows you to determine how the network connections at your site will be monitored. You can write BASIC scripts or C programs that can set points, generate alarms or pop up dialogs to inform the operator that a network connection has been lost. The following APIs are available:

- The IP Status API can be used to monitor when a connection to an IP address has been lost or formed.
- The Socket Status API provides more detail about each of the connections in use by CIMPLICITY.

Review:

- IP Status API.

- IP Status API functions.

The APIs are callable from any programming language that can call exported C functions from a DLL. This includes the CIMPLICITY BASIC Control Engine.

The functions reside in the **redwinsock.dll** DLL.

In the **api\redundant_api** directory under the CIMPLICITY root directory are two samples programs that demonstrate the APIs.

- The BASIC script **ip_status.bcl** generates and resets alarms as IP connections are lost or formed.
- The C program **sock_status.cpp** will print out the status information for all the sockets currently in use.

In addition, a compiled version of the **sock_status.cpp** program is in the **exe** directory.

Review:

- Socket Status API.
- Socket Status API Functions.

IP Status API

Use the Computer Cabling Redundancy IP Status API to monitor the state of connections to IP addresses.

The following BASIC script generates or clears an alarm depending upon the state of the connection to an IP address.

```
Declare Function InitSocketChange CDecl Lib "redwinsock.dll" _
  () As Long
Declare Function WaitSocketChange CDecl Lib "redwinsock.dll" _
 (ByVal data As Long, ByVal timeout As Long) As Long
Declare Function CloseSocketChange CDecl Lib "redwinsock.dll" _
 (ByVal data As Long) As Boolean
Declare Function GetNextSocketChange CDecl Lib "redwinsock.dll" _
 (ByVal data As Long, ByVal node As String, ByRef ipAddress As Long, _
  ByRef state As Long) As Long
Declare Function CvtIPAddress(x As Long) As String
Sub Main()
  Dim data As Long
  im node As String
  Dim ipAddress As Long
  Dim state As Long
  Dim n256 As Long
  n256 = 256
  data = InitSocketChange()
```

```
  If data <> 0 Then
    While 1
       = WaitSocketChange(data, 5000)
      If i = 1 Then
        ode = Space(256)
        While GetNextSocketChange(data, node, ipAddress, state)
          ipStr$ = CvtIPAddress(ipAddress)
          message$ = "IP address " & ipStr$
          refId$ = ipAddress
          If state = 2 Then
            AlarmUpdate "", "IPALARM", "$SYSTEM", AM_RESET_M, _
                        message$, "", refId$
          Else
            AlarmGenerate "", "IPALARM", "$SYSTEM", message$, _
                        "", refId$, True
          End If
          node = Space(256)
        Wend
      End If
    Wend
    i = CloseSocketChange(data)
  End If
End Sub
Function CvtIPAddress(x As Long) As String
  mod1 = ipAddress Mod n256
  If mod1 < 0 Then mod1 = 256 + mod1
  mod2 = ipAddress / n256 Mod n256
  f mod2 < 0 Then mod2 = 256 + mod2
  mod3 = ipAddress / (n256 * n256) Mod n256
  If mod3 < 0 Then mod3 = 256 + mod3
  mod4 = ipaddress / (n256 * n256 * n256) Mod n256
  If mod4 < 0 Then mod4 = 256 + mod4
  CvtIPAddress = mod1 & "." & mod2 & "." & mod3 & "." & mod4
End Function
```

IP Status API Functions

IP Status API Functions

The IP Status API supports the following functions.

IP Status API functions include:

- InitSocketChange
- WaitSocketChange
- CloseSocketChange
- GetNextSocketChange

InitSocketChange

Function	Description
Syntax	**void *InitSocketChange();**
Description	This function initializes the API to provide changes in IP statuses.
Comments	This function returns a pointer used in other functions to identify this request, or NULL if the initialization failed.

```
Dim data As Long
data = InitSocketChange()
```

WaitSocketChange

Function	Description
Syntax	**DWORD WaitSocketChange (void *arg, DWORD timeout);**
Description	This function waits for the next change in an IP status to occur.
Comments	The **arg** pointer is the pointer returned by the **InitSocketChange** function. The **timeout** parameter is the length of time in milliseconds to wait for a change to occur. If the value is –1 then the function will wait forever. This function returns a 1 if a status has changed or 0 if the function timed out.

```
Dim data As Long
i = WaitSocketChange(data, 5000)
```

CloseSocketChange

Function	Description
Syntax	**void CloseSocketChange (void *arg);**
Description	This function does a cleanup of the state change notification.
Comments	The **arg** pointer is the pointer returned by the **InitSocketChange** function. This function does not have a return status.

```
Dim data As Long
i = CloseSocketChange(data)
```

GetNextSocketChange

<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>Syntax</td><td>DWORD GetNextSocketChange (void *arg, TCHAR *node, DWORD *ipAddress, DWORD *state);</td></tr>
<tr><td>Description</td><td>This function gets the information about the next IP status change.</td></tr>
<tr><td>Comments</td><td>The arg pointer is the pointer returned by the InitSocketChange function. The node buffer is used to hold the name of the node for this IP address. It should be 255 characters or larger. The ipAddress is the IP address that has had a status change. The state is the new state of the IP address. The states are:
<table>
<tr><th>Value</th><th>State</th></tr>
<tr><td>0</td><td>Not connecting</td></tr>
<tr><td>1</td><td>Connecting</td></tr>
<tr><td>2</td><td>Connected</td></tr>
<tr><td>3</td><td>Deleted</td></tr>
<tr><td>4</td><td>Unknown</td></tr>
</table>
This function returns 1 if it is successful, or 0 if there are no more changes.</td></tr>
</table>

Dim data As Long

```
Dim node As String
Dim ipAddress As Long
Dim state As Long
Node = Space(256)
While GetNextSocketChange (data,node,ipAddress,state)
Wend
```

Socket Status API

You can use the Computer Cabling Redundancy Socket Status API to monitor the state of all the sockets currently in use by CIMPLICITY Computer Cabling Redundancy.

The following C program prints out the status of each of the sockets.

```
#include <string.h>
#include <inc_path/cor.h>
#include <inc_path/redwinsock.h>
void print_sockaddr_in(struct sockaddr_in *addr);
TCHAR *socketUse[] =
{
  _T("None"),
  _T("Listen"),
  _T("Connect"),
  T("Accept"),
};
TCHAR *socketState[] =
```

```
{
  _T("None"),
  _T("Connecting"),
  _T("Connected"),
};
int main()
{
  HANDLE dwChangeHandle;
  dwChangeHandle = FindFirstSocketChangeNotification();
  if(dwChangeHandle == INVALID_HANDLE_VALUE)
    return 0;
  HANDLE objectArray[1];
  WORD objectCount = 1;
  objectArray[0] = dwChangeHandle;
  DWORD dwWaitStatus = WAIT_OBJECT_0;
  while(1)
  {
    time_t ltime = time(NULL);
    printf("\n%s", ctime(&ltime));
    switch(dwWaitStatus)
    {
    case WAIT_OBJECT_0:
      struct SocketFindData findSocketData;
      if(FindFirstSocket(dwChangeHandle, &findSocketData))
      {
        do
        {
          if(findSocketData.opened)
          {
            printf("%-10s ", findSocketData.prcName);
            printf("%-10s ", findSocketData.imageName);
            printf("%-6s ", findSocketData.opened ? "Opened" :
                                                    "Closed");
            printf("%-7s ", socketUse[findSocketData.socketUse]);
            rintf("%-9s ", findSocketData.isRedundant ? "Redundant" :
                                                        "");
            printf("%-9s ", findSocketData.connectionCompleted ?
                              "Completed" : "");
            printf("%-10s ", findSocketData.hostName);
            printf("\n");

            unsigned int i;
            for(i = 0; i < findSocketData.connectionCount; i++)
            {
               if(findSocketData.sockets[i].isOpen)
               {
                   printf("\t%2d: %-10s ", i,
                   socketState[findSocketData.sockets[i].socketState]);
                printf("%-9s ", findSocketData.sockets[i].hasException
                                  ? "Exception" : "");
                if(findSocketData.socketUse == SOCKET_ACCEPT
                || findSocketData.socketUse == SOCKET_CONNECT)
                 {
```

```
                  print_sockaddr_in(
                      &findSocketData.sockets[i].partnerAddr);
                 }
                else if(findSocketData.socketUse == SOCKET_LISTEN)
                 {
                  print_sockaddr_in(
                     &findSocketData.sockets[i].localAddr);
                 }
                 printf("\n");
               }
             }
           }
         } while(FindNextSocket(dwChangeHandle, &findSocketData));
         printf("\n");
         FindCloseSocket(dwChangeHandle);
       }
         break;
     default:
       FindCloseSocketChangeNotification(dwChangeHandle);
       return 0;
     }
     dwWaitStatus = WaitForMultipleObjects(objectCount,
                                           objectArray,
                                           FALSE,
                                           INFINITE);
     if(FindNextSocketChangeNotification(dwChangeHandle) == FALSE)
     {
       FindCloseSocketChangeNotification(dwChangeHandle);
       return 0;
     }
   }
   return 0;
}
void print_sockaddr_in(struct sockaddr_in *addr)
{
   printf(_T(" %d.%d.%d.%d %u "),
          addr->sin_addr.s_net,
          addr->sin_addr.s_host,
          addr->sin_addr.s_lh,
          addr->sin_addr.s_impno,
          ntohs(addr->sin_port));
}
```

Socket Status API Functions

Socket Status API Functions

The Socket Status API supports the following functions.

Socket Status API functions include:

- FindFirstSocketChangeNotification
- FindNextSocketChangeNotification
- FindCloseSocketChangeNotification
- FindFirstSocket
- FindNextSocket
- FindCloseSocket

FindFirstSocketChangeNotification

Function	Syntax
Syntax	**HANDLE FindFirstSocketChangeNotification();**
Description	This function initializes notification that the socket data has changed.
Comments	There are no input or output arguments. This function returns the handle used in the find socket routines, or **INVALID_HANDLE_VALUE** on failure.

```
HANDLE dwChangeHandle;
dwChangeHandle = FindFirstSocketChangeNotification();
if(dwChangeHandle == INVALID_HANDLE_VALUE)
    return 0;
```

FindNextSocketChangeNotification

Function	Description
Syntax	**BOOL FindNextSocketChangeNotification (HANDLE changeHandle);**
Description	This function prepares to receive the next socket change notification.
Comments	The **changeHandle** input argument is the handle returned by the **FindFirstSocketChangeNotification** function. There are no output arguments. This function returns TRUE if successful, or FALSE on failure.

```
HANDLE dwChangeHandle;
if(FindNextSocketChangeNotification(dwChangeHandle)==FALSE)
{
   FindCloseSocketChangeNotification(dwChangeHandle);
   return 0;
}
```

FindCloseSocketChangeNotification

Function	Description
Syntax	**BOOL FindCloseSocketChangeNotification (HANDLE changeHandle);**

Function	Description
Description	This function closes the socket change notification.
Comments	The **changeHandle** input argument is the handle returned by the **FindFirstSocketChangeNotification** function. There are no output arguments. This function returns TRUE if successful, or FALSE on failure.

```
HANDLE dwChangeHandle;
FindCloseSocketChangeNotification(dwChangeHandle);
```

FindFirstSocket

Function	Description
Syntax	**BOOL FindFirstSocket (HANDLE changeHandle, Struct SocketFindData *findSocketData);**
Description	This function finds the first socket data.
Comments	The **changeHandle** input argument is the handle returned by the **FindFirstSocketChangeNotification** function. The **findSocketData** output argument contains all the information about the found socket. This structure is defined in **<inc_path/toolkit.h>.** This function returns TRUE if a socket is found, or FALSE if there are no more sockets.

```
HANDLE dwChangeHandle;
struct SocketFindData findSocketData;
if(FindFirstSocket(dwChangeHandle, &findSocketData))
  ;
```

FindNextSocket

Function	Description
Syntax	**BOOL FindNextSocket (HANDLE changeHandle, Struct SocketFindData *findSocketData);**
Description	This function finds the succeeding socket data.
Comments	The **changeHandle** input argument is the handle returned by the **FindFirstSocketChangeNotification** function. The **SocketFindData** output argument contains all the information about the found socket. This structure is defined in **< inc_path/toolkit.h>.** This function returns TRUE if a socket is found, or FALSE if there are no more sockets.

```
HANDLE dwChangeHandle;
struct SocketFindData findSocketData;
do
{
} while(FindNextSocket(dwChangeHandle, &findSocketData));
```

FindCloseSocket

Function	Description
Syntax	**BOOL FindCloseSocket (HANDLE changeHandle);,**
Description	This function finishes looking for socket data.
Comments	The **changeHandle** input argument is the handle returned by the **FindFirstSocketChangeNotification** function. This function returns TRUE if successful, or FALSE if an error occurred.

```
HANDLE dwChangeHandle;
FindCloseSocket(dwChangeHandle);
```

Supported Communication Interfaces

Supported Communication Interfaces

This appendix documents the redundant communication interfaces supported by Server Redundancy.

The supported redundant communication interfaces are:

- CCM2
- Genius
- SNPX
- Allen-Bradley Communications
- DDE Client
- Modbus RTU
- Modbus TCP/IP
- OPC Client
- Point Bridge
- Series 90 TCP/IP Triplex Device Communications

In addition, CIMPLICITY supports the development of Server Redundant Device Communication Toolkit drivers. Use the heartbeat function in the enabler to determine if the secondary server can communicate with its configured devices.

CCM2 Communications

Server redundant computer configurations that use the CCM2 Communications option must be configured as in the following diagram:

The primary and secondary servers must have independent cable paths to the PLC. The PLC must have two serial ports available for CCM2 communications. Both of the PLC's serial ports must be configured with the same CPU ID.

The recommended configuration is to use the CMM module in CCM2 mode. Both ports on the CMM module can be configured for CCM2 communications.

Genius Communications

Server redundant computer configurations that use the Genius Communications option must be configured as in the following diagram:

The primary and secondary servers must have different PCIM addresses. Unsolicited datagrams sent from the PLC must be sent to both the primary and secondary servers.

SNPX Communications

Server redundant computer configurations that use the SNPX Communications option must be configured as in the following diagram:

The primary and secondary servers must have independent cable paths to the PLC. The PLC must have two serial ports available for SNPX communications. Both of the PLC's serial ports must be configured with the same CPU ID.

Allen-Bradley Communications

Server redundant computer configurations that use the Allen-Bradley Communications option must be configured as in the following diagram:

Communications can be done over an Ethernet network or over a Data Highway Plus network using Allen-Bradley 1784 KTX cards. Unsolicited data sent from the PLC must be sent to both the primary and secondary servers.

DDE Client Communications

Server redundant computer configurations that use the DDE Client Communications option must be configured in one of the following ways:

- The NetDDE Server must be installed and configured identically on both the primary and secondary servers.

- The NetDDE Server must be installed and configured on a computer other than the primary and secondary servers.

Modbus RTU Communications

Server redundant computer configurations that use the Modbus RTU Communications option must be configured as in the following diagram:

The programmable controller needs to respond through both of its serial cards at the same standby address at the same time. However, the standby node is only heart beating. It is not reading the full data set. The active node reads the full data set.

Important: Because both RTU cards have the same ID they have to be in separate networks.

Modbus TCP/IP

Server redundant computer configurations that use the Modbus TCP/IP Communications option must be configured as in the following diagram:Communications

The primary and secondary servers are on the same Ethernet LAN connected to a single Ethernet card in the programmable controller.

OPC Client

If your OPC Server is configured to support server redundant configurations, OPC Client will also support server redundant configurations.

Point Bridge

Point Bridge communications is fully supported in server redundant computer configurations.

Series 90 TCP/IP Triplex Device Communications

Series 90 TCP/IP Triplex Device communications is fully supported in server redundant computer configurations.

Server Redundancy Configuration Parameters

Server redundancy configuration parameters include:

Parameter	Description
REDUND_LINK_SLEEP	Wait time before the Router creates a link to the standby node.
REDUND_PROBE_COUNT	Number of missed probes before a failover.
REDUND_PROBE_DELAY	Gap between sending probes.
REDUND_PROBE_PORT	TCP/IP port number to implement the probe mechanism.
SECONDARY_STARTUP_TIMEOUT	Starting time delay for the secondary server project, when both the project on both the primary and secondary start at boot.

Computer Cabling Redundancy Status Log Error Messages

Computer Cabling Redundancy Status Error Messages

The following is a list of errors you may encounter in the CIMPLICITY Status Log relating to Computer Cabling Redundancy.

Error messages cover:

- Binding failures.
- Connection failures.
- Socket failures.
- Missed communications.

Binding Failures

This error should only occur if a non-CIMPLICITY communication program is using IP ports in the same range as the Computer Cabling Redundancy option.

Failed to bind to port

CIMPLICITY attempts to use the same IP port on both network interface cards (NIC). If this error occurs try changing the range of IP ports that Computer Cabling Redundancy is using.

Connection Failures

These errors occur when the Computer Cabling Redundancy option starts a connection with another computer:

Failed to receive connection ID: invalid ID

Failed to complete connection: <IP Address>

Failed to receive connection ID: invalid ID

Failed to receive connection ID: timeout

Failed to receive connection ID: exception

Failed to receive connection ID: recv failed

When the option starts a connection with another computer it exchanges some identifying information. If this information is not received, the connection will fail.

This is typically caused by an incorrect configuration. If the local computer believes that the remote computer is supporting Computer Cabling Redundancy but the remote computer is not configured for Computer Cabling Redundancy, then this error will occur. This error can also occur if the network is broken during the connection setup time.

Socket Failures

This error occurs when the local socket is waiting for data:

Error performing select on socket

This error should never occur.

Missed Communications

This error occurs when network traffic is not being received from the remote computer:

Missed hearing from partner: < IP Address>

If this error occurs and the network is functioning you should check the **PING_INTERVAL** and **PING_COUNT** parameters in the **cimhosts.txt** file. You may need to increase them depending on the load on your computers or network.

Problems and Solutions

1. Open the project's Workbench.

2. Click Tools on the menu bar.

3. Select Command prompt…
   Type **datamerge.exe** in the Command Prompt window.

4. Start the project.

5. Open the project's Workbench.

6. Click Tools on the menu bar.

7. Select Command prompt…
   Type **datamerge.exe** in the Command Prompt window.