

CICS[®] Transaction Gateway Configuration



Configuring CICS Transaction Gateway for Terminal Servlets (WebSphere Application Server for Windows NT Version 3.5)

CICS® Transaction Gateway Configuration



Configuring CICS Transaction Gateway for Terminal Servlets (WebSphere Application Server for Windows NT Version 3.5)

Contents

Chapter 1. Overview	1	Chapter 6. Configuring WebSphere Application Server for the Terminal Servlet.	11
Chapter 2. Software checklist	3	Configuring and loading the TerminalServlet	11
Chapter 3. Installing the software	5	Configuring the parameters	18
Installing the workstation software	5	Setting up WebSphere Application Server to use a local CICS Transaction Gateway	20
Installing the Gateway Server software	5	Chapter 7. Testing using the supplied samples	21
JDK	5	Chapter 8. Troubleshooting	25
WebSphere Application Server	5	WebSphere Application Server log files	25
CICS Transaction Gateway	6	TerminalServlet logging	25
Chapter 4. Configuring the Gateway Server software	7	CICS Universal Client troubleshooting	26
Configuring CICS Transaction Gateway	7	CICS Universal Client log	26
Chapter 5. Starting and stopping the Gateway Server software	9	CICS Transaction Gateway tracing	26
Starting CICS Transaction Gateway	9	CICS Universal Client tracing	27
Stopping CICS Transaction Gateway	9	Appendix. Trademarks	29
Starting and stopping HTTP Server	9		
Starting and stopping WebSphere Admin server	9		

Chapter 1. Overview

The sample configuration in Figure 1 on page 2 shows how you can use Java servlets from a workstation to converse with a remote CICS across HTTP. The browser on the workstation uses only HTML pages.

The servlets are managed by WebSphere Application Server and run by the HTTP Server plug-in for WebSphere Application Server on the gateway machine.

The servlets issue ECI or EPI requests to the CICS Transaction Gateway, which makes the relevant calls to the coupled CICS Universal Client. The CICS Universal Client then passes the calls to the remote CICS server, using a TCP/IP connection.

Although the sample configuration uses a TCP/IP connection to the remote CICS server, you can configure your CICS Universal Client to communicate with any CICS server that supports ECI or EPI, for example, CICS Transaction Server for OS/390. For a list of CICS servers that CICS Universal Client can connect to, refer to your *CICS Universal Client Administration* book.

The servlet used in this configuration is the Terminal Servlet provided with CICS Transaction Gateway. The Terminal Servlet allows you to access CICS 3270 applications using a Web browser. You create Web pages that invoke the Terminal Servlet to start a CICS transaction, display screen information sent from a CICS server, and send screens back to CICS.

For full information on the Terminal Servlet, including how it maps 3270 screens to HTML, refer to your *CICS Transaction Gateway Administration* book.

In this configuration document we cover the following topics:

- “Chapter 2. Software checklist” on page 3
- “Chapter 3. Installing the software” on page 5
- “Chapter 4. Configuring the Gateway Server software” on page 7
- “Chapter 5. Starting and stopping the Gateway Server software” on page 9
- “Chapter 6. Configuring WebSphere Application Server for the Terminal Servlet” on page 11
- “Chapter 7. Testing using the supplied samples” on page 21
- “Chapter 8. Troubleshooting” on page 25

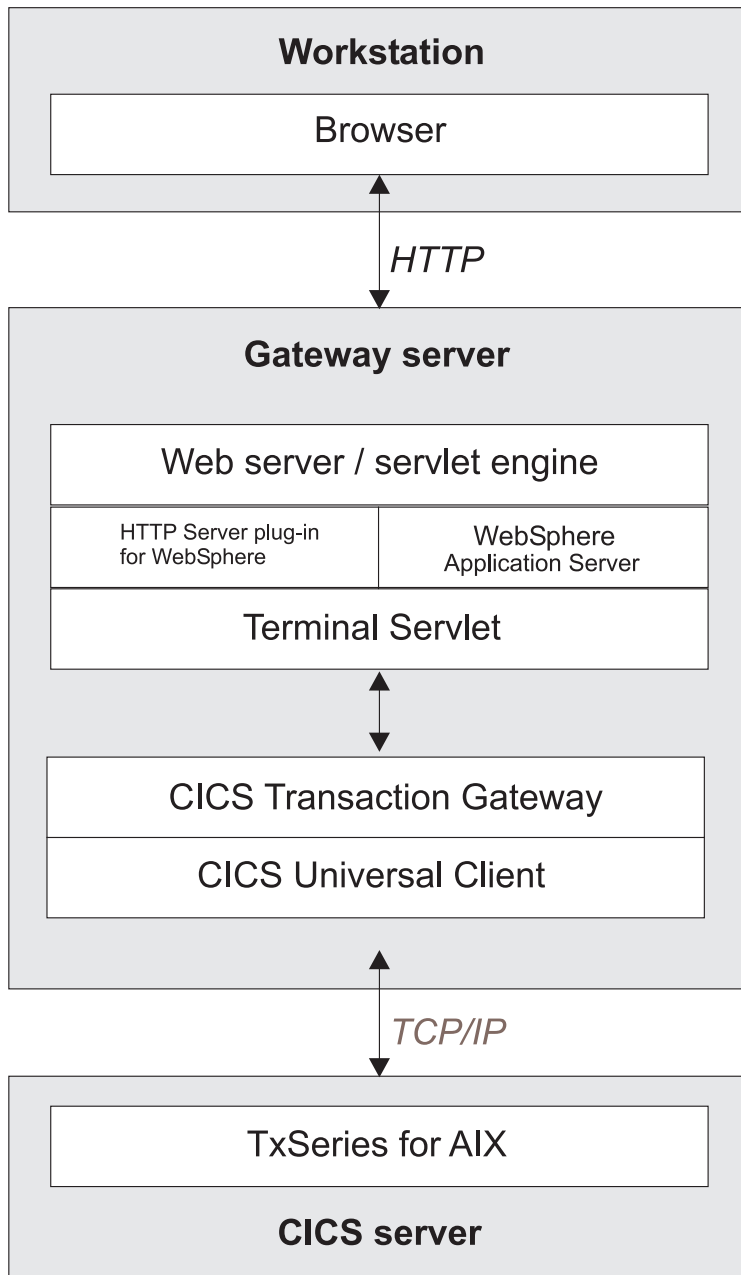


Figure 1. CICS Transaction Gateway using servlets across HTTP

Chapter 2. Software checklist

The levels of software we used in the sample configuration are not necessarily the latest levels available. Check the relevant products for levels of compatible software.

We used the following software on the workstation:

- Windows NT Workstation Version 4.0 (service level 6A)
- Netscape Navigator for Windows NT Version 4.07

We used the following software on the gateway host:

- Windows NT Workstation Version 4.0 (service level 6A)
- Java Development Kit (JDK) Version 1.2.2 for Windows NT (included with WebSphere Application Server)
- IBM HTTP Server Version 1.3.12 (included with WebSphere Application Server)
- WebSphere Application Server Version 3.5 for Windows NT
- CICS Transaction Gateway Version 3.1.2 (includes CICS Universal Client Version 3.1.2)

We used the following software on the server host:

- TXSeries for AIX Version 4.3, which supports CICS access through TCP/IP
- CICS Transaction Gateway supplied EPIC transaction. See “Chapter 7. Testing using the supplied samples” on page 21 for details of how to prepare this transaction for use.

Software checklist

Chapter 3. Installing the software

Although this section describes the installation of all necessary software for this CICS Transaction Gateway scenario, the installation cannot be considered complete until you perform the configuration steps described in “Chapter 4. Configuring the Gateway Server software” on page 7.

Installing the workstation software

This sample configuration assumes that Windows NT Version 4.0 is already installed and service level 6A has been applied. It also assumes Netscape Navigator for NT Version 4.07 or later is installed.

No special configuration is needed for the workstation.

Installing the Gateway Server software

You should install the software in the order given in this section, to avoid dependency problems.

JDK

You can install a JDK as part of the installation process for WebSphere, as described in the following section.

You can verify that the JDK is installed and working with the `java -version` command.

WebSphere Application Server

To install WebSphere Application Server Advanced Edition and HTTP Server for Windows NT:

1. Insert the installation CD into your CD-ROM drive.
2. Run the setup program `\nt\setup.exe`.
3. Select the appropriate language and then **Next**.
4. Accept the license agreement, and then **Next**.
5. On the Install Options page, select **Quick Install** and then **Next**. This option installs InstantDB, rather than DB2, and this is sufficient for use with WebSphere Application Server.
6. On the Security Options page, fill in the user ID, password, and confirming password to use for the application server and select **Next**. You must have administrative privileges.
7. Specify the destination directory, then select **Next**.

Installing the software

8. Select **Next** again to begin the installation.

You can find more information about installation of WebSphere Application Server and testing the installation at this URL:

<http://www-4.ibm.com/software/webservers/appserv/doc/v35/ae/infocenter/>

CICS Transaction Gateway

To install CICS Transaction Gateway:

1. Make sure you are logged on with Administrator privileges.
2. Follow the installation instructions in the *CICS Transaction Gateway Administration* book.

Make sure that you perform a full installation of CICS Transaction Gateway, to ensure that all necessary Terminal Servlet files are installed.

Chapter 4. Configuring the Gateway Server software

Start with the configuration steps in this section only after completing all installation steps described in “Chapter 3. Installing the software” on page 5.

Configuring CICS Transaction Gateway

You use the CICS Transaction Gateway’s configuration tool to define the CICS servers and make the servers reachable by the CICS Transaction Gateway. The configuration tool generates the CTG.INI file, which is located in the \bin directory. The CICS Transaction Gateway uses the CICS Universal Client and the CTG.INI file to establish a connection to a CICS server.

For information on using the configuration tool, refer to your *CICS Transaction Gateway Administration* book.

You may need to change the following **Server** configuration settings (see Figure 2 on page 8):

Server name

An arbitrary name for a particular CICS server.

Description

An arbitrary description for the CICS server.

Network protocol

A protocol for communication with the CICS server. We use TCPIP in our sample.

Hostname or IP address

The TCP/IP address of the CICS server.

Port The TCP/IP port number of the CICS server. Set this to zero to use the default TCP/IP port number of the CICS server.

The *CICS Universal Client Administration* book and the configuration tool’s online help provide descriptions of the configuration settings for CICS Universal Client.

Configuring the Gateway Server software

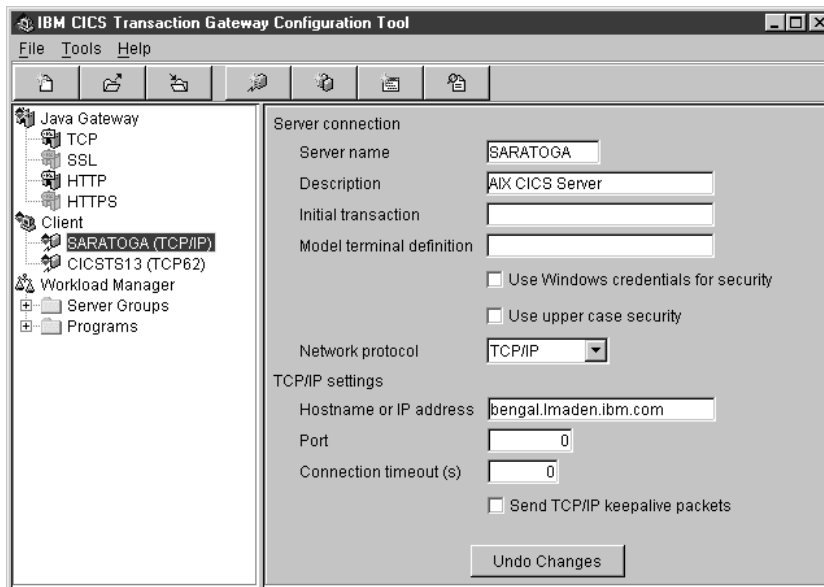


Figure 2. Configuration tool settings for using servlets

Figure 3 shows an excerpt from the resultant CTG.INI file that would make the CICS AIX server known to the CICS Transaction Gateway.

```
SECTION CLIENT = *  
  :::  
ENDSECTION  
  :::  
SECTION SERVER = SARATOGA  
  DESCRIPTION=AIX CICS Server  
  :::  
  PROTOCOL=TCPIP  
  NETNAME=bengal.lmaden.ibm.com  
  PORT=0  
ENDSECTION  
  :::  
SECTION DRIVER = TCPIP  
  DRIVERNAME=CCLIBMIP  
ENDSECTION
```

Figure 3. CTG.INI file definitions

Chapter 5. Starting and stopping the Gateway Server software

This section describes how to start and stop the software products installed and configured in the preceding sections.

Starting CICS Transaction Gateway

To start CICS Transaction Gateway, issue the following commands:

1. `cicscli -s`
2. `ctgstart`

Stopping CICS Transaction Gateway

Enter the character `q` in the CICS Transaction Gateway startup session console to terminate the program.

Starting and stopping HTTP Server

To start or stop HTTP Server, open the Control Panel and select **Services**. Select **IBM HTTP Server** and then select **Start** or **Stop** as required.

You can also start HTTP Server from the **Start** menu.

Starting and stopping WebSphere Admin server

To start or stop the WebSphere Admin server, open the Control Panel and select **Services**. Select **IBM WS AdminServer** and then select **Start** or **Stop** as required.

Chapter 6. Configuring WebSphere Application Server for the Terminal Servlet

This section describes how to use the WebSphere Application Server to prepare the sample servlet, TerminalServlet, supplied with the CICS Transaction Gateway.

Configuring and loading the TerminalServlet

To configure and load the TerminalServlet:

1. Ensure that the WebSphere service is started. Open the Control Panel and select **Services**. Start the service by selecting **IBM WS AdminServer** and then selecting **Start**.
2. Run the WebSphere Advanced Administrative Console, which is how you manage WebSphere Application Server. From the Start menu, select **Programs, IBM WebSphere, Application Server 3.5**, then **Administrator's Console**, see Figure 4.

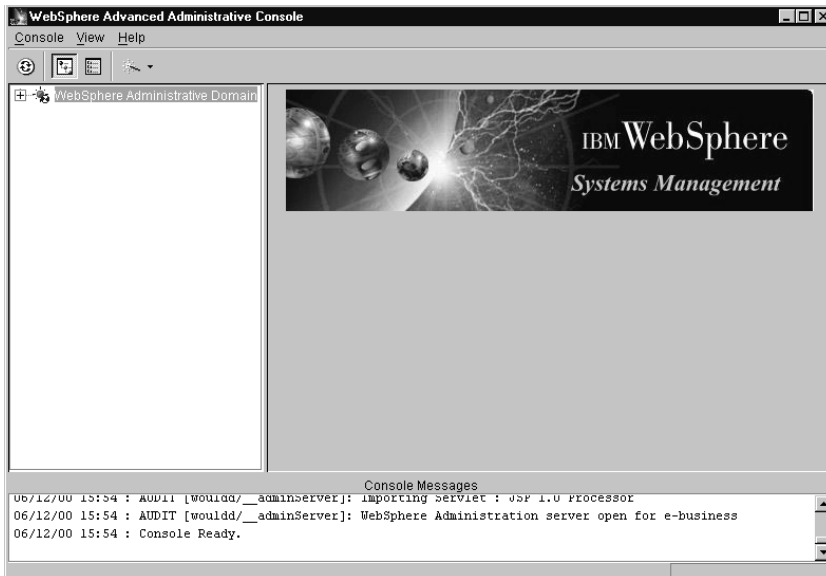


Figure 4. WebSphere Advanced Administrative Console

Configuring WebSphere Application Server for the Terminal Servlet

3. In the Administrative Console, select the down arrow next to the wand icon (wizards) and select the wizard for **Create Application Server**, see Figure 5.

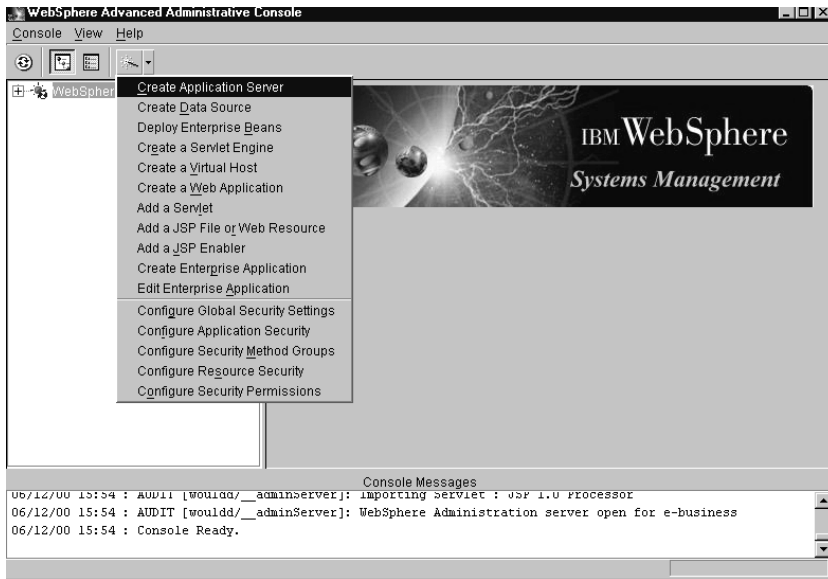
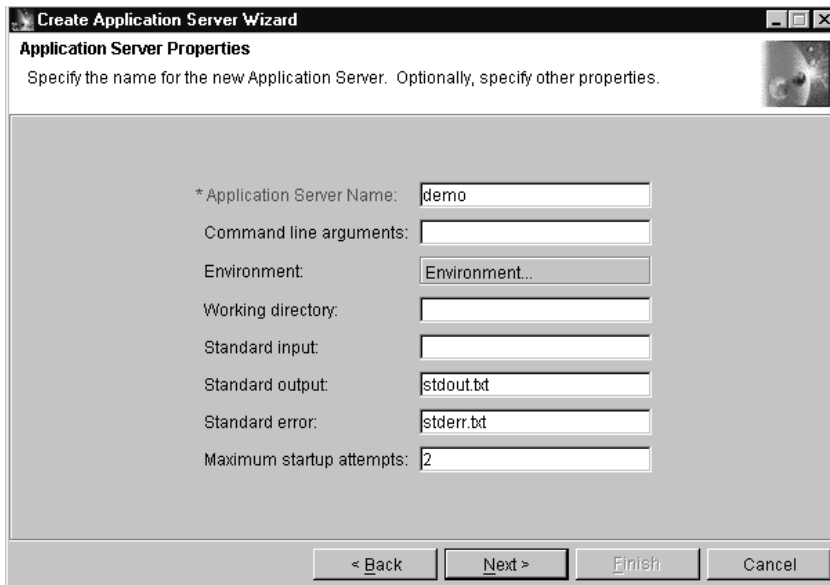


Figure 5. WebSphere Advanced Administrative Console

4. Deselect **Enterprise Beans** (unless you intend to use Enterprise Java Beans).
5. Select **Next**.
6. Enter the **Application Server Name** of your choice, for example, demo, (see Figure 6 on page 13), and then select **Next**.

Configuring WebSphere Application Server for the Terminal Servlet



The screenshot shows a window titled "Create Application Server Wizard" with a sub-tab "Application Server Properties". The window contains the following fields and controls:

- Instruction: "Specify the name for the new Application Server. Optionally, specify other properties."
- * Application Server Name:
- Command line arguments:
- Environment:
- Working directory:
- Standard input:
- Standard output:
- Standard error:
- Maximum startup attempts:
- Navigation buttons: "< Back", "Next >", "Finish", and "Cancel".

Figure 6. Application Server Properties

7. In the next panel, select the **Do not start the server automatically after creating it** check box and then **Next**.
8. In the tree, expand **Nodes** and select the only available one (the local machine), (see Figure 7 on page 14), and select **Next**.

Configuring WebSphere Application Server for the Terminal Servlet

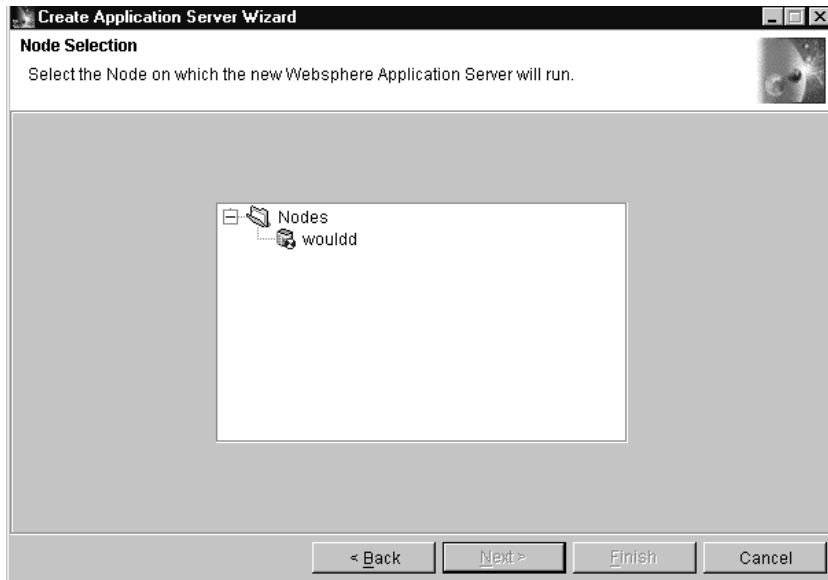


Figure 7. Node Selection

9. In the **Virtual Host Name** field enter a sensible name, or expand **Virtual Hosts** and select **default_host**. Select **Next**.
10. Accept the default **Servlet Engine Name** and select **Next**.
11. Enter a **Web Application Name** or accept the default presented, in this case, **demoWebApp**. In the **Web Application Path** field, enter **/servlet**.
12. Select the **Advanced** tab.
13. In the **Classpath** field, enter the locations for the **ctgclient.jar** and **ctgserver.jar**, (see Figure 8 on page 15), and select **Next**.

Configuring WebSphere Application Server for the Terminal Servlet

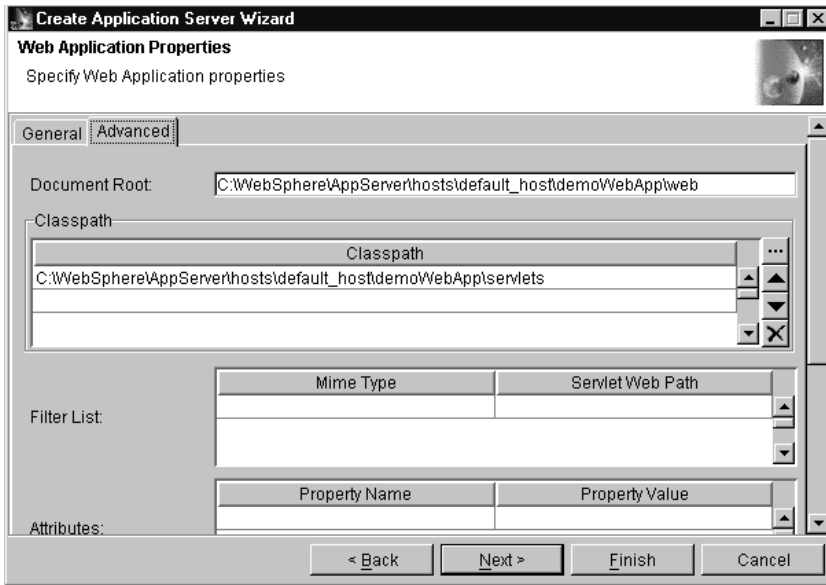


Figure 8. Web Application Properties

14. Select **Enable File Servlet** and deselect **Serve Servlets by Classname**, then select **Finish**, see Figure 9.

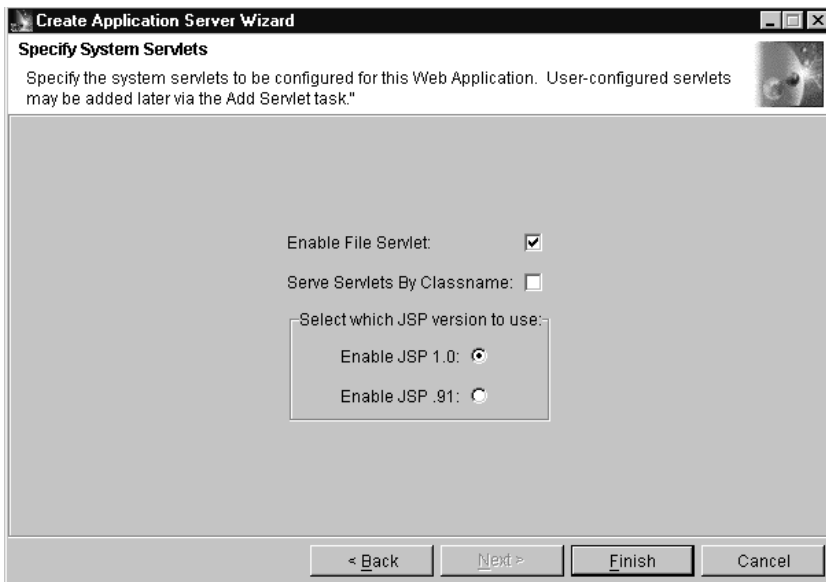


Figure 9. Specify System Servlets

Configuring WebSphere Application Server for the Terminal Servlet

15. In the tree view, find the Application Server, Servlet Engine, and Web application you created. It should contain three default servlets, **Error Reporting Facility**, **File Serving Enabler**, and **JSP 1.0 Processor**. See Figure 10.

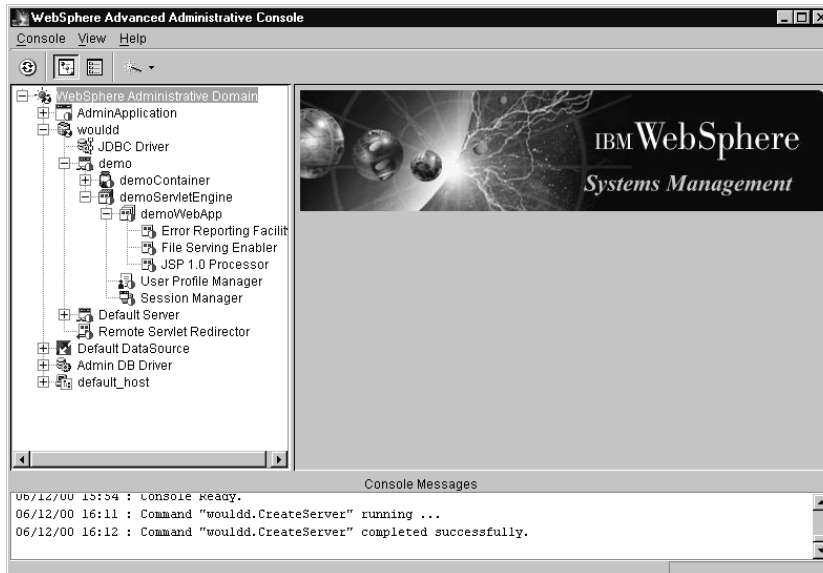


Figure 10. WebSphere Advanced Administrative Console - Expanded Tree View

16. Right-click on the name of your Web application, and select **Create** and then **Servlet** from the menu to display the Create Servlet dialog box.
17. Enter Terminal Servlet in **Servlet Name**, and `com.ibm.ctg.servlet.TerminalServlet` in the **Servlet Class Name** field.
18. In **Servlet Web Path List** you are presented with the beginning of a URL, for example, `/servlet/webapp/DemoApp/`. Select the **Add** button, and add to the URL so that it contains `/servlet/webapp/demoApp/TerminalServlet`. Select **OK** (or **Create** on some versions), see Figure 11 on page 17.

Configuring WebSphere Application Server for the Terminal Servlet

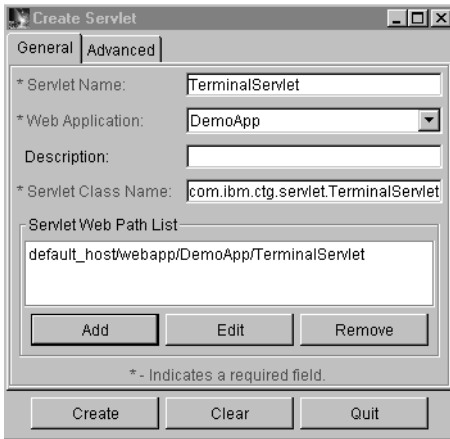


Figure 11. Create Servlet dialog box - General

19. Select the **Advanced** tab, and enter `Servlet@propertyFile` in the **Init Parameters** field, and enter the value for the location of your `Servlet.properties` file. Set **Load at Startup** to true, see Figure 12.

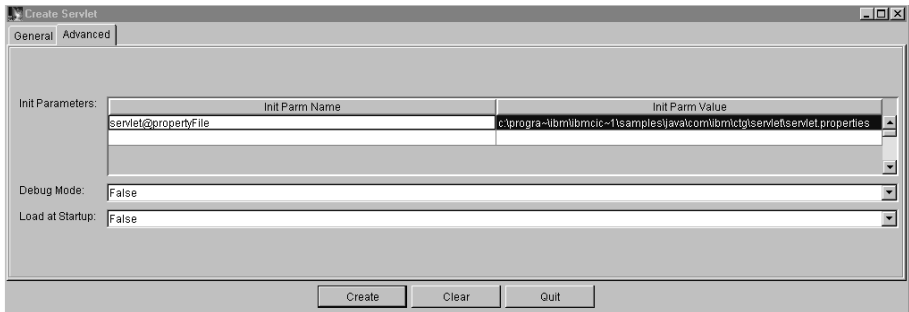


Figure 12. Create Servlet dialog box - Advanced

20. Select **OK** and the servlet should appear in the list under your Web application, Select the **Play** button to start it, see Figure 13 on page 18.

Configuring WebSphere Application Server for the Terminal Servlet

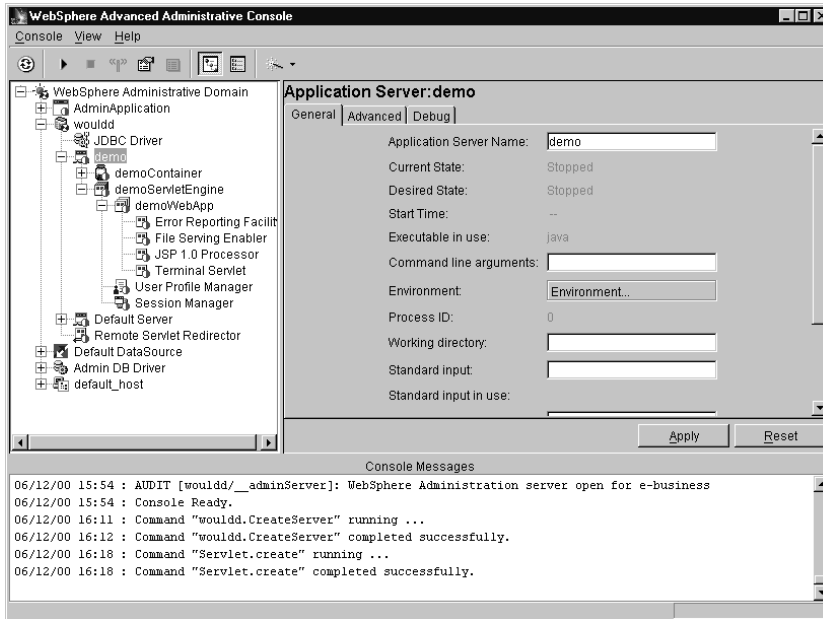


Figure 13. Application Server settings

21. Select the root node of your newly created application server and select the **Play** button to start it.
22. While starting it should load the terminal servlet and this can now be reached at the URL:

`http://local_machine/servlet/webapp/DemoWebApp/TerminalServlet`

Configuring the parameters

In the previous section you specified the `servlet.properties` file to be used with the `TerminalServlet`. The properties file should contain the following properties and values for use with the samples provided:

page@default

Example: `file://epissam1.shtml`

This parameter specifies the default server-side include file that is used for returning CICS HTML to the browser.

Page mapping properties can be set either to a URL, or to point to a template file. This must start with `file://`, and if you set the **servlet@templateDir** property, you can set template filenames relative to a template directory, as in our example.

page@demo.MAPINQ1ScreenHandler

Example:

Configuring WebSphere Application Server for the Terminal Servlet

`file://ctg/samples/java/com/ibm/ctg/servlet/epissam3.html`

where *ctg* is the directory where CICS Transaction Gateway is installed. This parameter specifies the HTML page that is used as a template for mapping CICS 3270 map fields to HTML tags.

page@disconnected

Example: `file://epissam.html`

This parameter specifies the HTML file that will be displayed when the user clicks on the disconnect button from a CICS HTML page. File `epissam.html` contains the URL links that run the CICS Transaction Gateway servlet samples.

pool@gatewayURL

Example: `local:`

This parameter specifies the location of the CICS Transaction Gateway. For our scenario, the CICS Transaction Gateway is located on the same gateway server host as the WebSphere Application Server, so we use the value `local:`.

pool@handlerPath

Example: `/ctg/classes/sample.jar`

This parameter specifies the archive containing the Java classes used by the samples.

pool@serverName

Example: `SARATOGA`

This parameter value corresponds to the arbitrary server name defined using the configuration tool, see “Configuring CICS Transaction Gateway” on page 7. Use the CICS server name you specified there.

servlet@extendedLogging

Example: `false`

Set this parameter to true to enable logging. See “Chapter 8. Troubleshooting” on page 25 for more information about using this parameter.

servlet@propertyFile

Example:

`/ctg/samples/java/com/ibm/ctg/servlet/servlet.properties`

This parameter specifies the file containing default parameters for the TerminalServlet. The parameters described in this list are also described in the `servlet.properties` file.

Configuring WebSphere Application Server for the Terminal Servlet

`servlet@templateDir`

Example: `/ctg/samples/java/com/ibm/ctg/servlet`

This parameter specifies the directory where the HTML pages and server-side includes are located.

Full information on Terminal Servlet parameters is given in the *CICS Transaction Gateway Administration* book.

Setting up WebSphere Application Server to use a local CICS Transaction Gateway

To set up WebSphere Application Server to use a local CICS Transaction Gateway:

1. Start the Administrators Console
2. Go to your Application Server settings page, see Figure 13 on page 18.
3. Modify **Command line arguments**, to read, for example:

```
-mx128m -classpath  
c:\Program Files\IBM\CICS Transaction Gateway\classes\ctgclient.jar;  
c:\Program Files\IBM\CICS Transaction Gateway\classes\ctgserver.jar
```

This must all be on one line. The location of the .jar files will vary depending on where you installed CICS Transaction Gateway.

4. Now click once on the **Environment** box, and add the PATH environment variable, for example:
PATH=c:\Program Files\IBM\CICS Transaction Gateway\bin
5. Select **OK** and then **Apply**. If you forget to Apply, nothing will happen!

Chapter 7. Testing using the supplied samples

After you have installed and configured all relevant products for the sample configuration, we recommend that you:

Test your configuration, using the samples provided with the CICS Transaction Gateway. The samples include the EPIC transaction, which you must place on your remote CICS server. The CICS Universal Client includes several files needed for the sample CICS server programs. These files are located in the Server subdirectory of the samples directory and include:

- Epiinq.bms
- Epiinq.ccs

Copy the files to your CICS server and use them to build the CICS EPIC transaction. Refer to the documentation supplied with your CICS server for information about building CICS transactions.

To display the CICS Transaction Gateway Terminal Servlet Samples page, point your browser to a URL like the following:

```
http://azov.almaden.ibm.com/servlet/TerminalServlet?request=disconnect
```

Substitute the host name of your own gateway server host in the URL.

Figure 14 on page 22 shows the CICS Transaction Gateway Terminal Servlet Samples page. The page contains URL links to the samples supplied by the CICS Transaction Gateway and instructions on using the samples.

Testing using the supplied samples

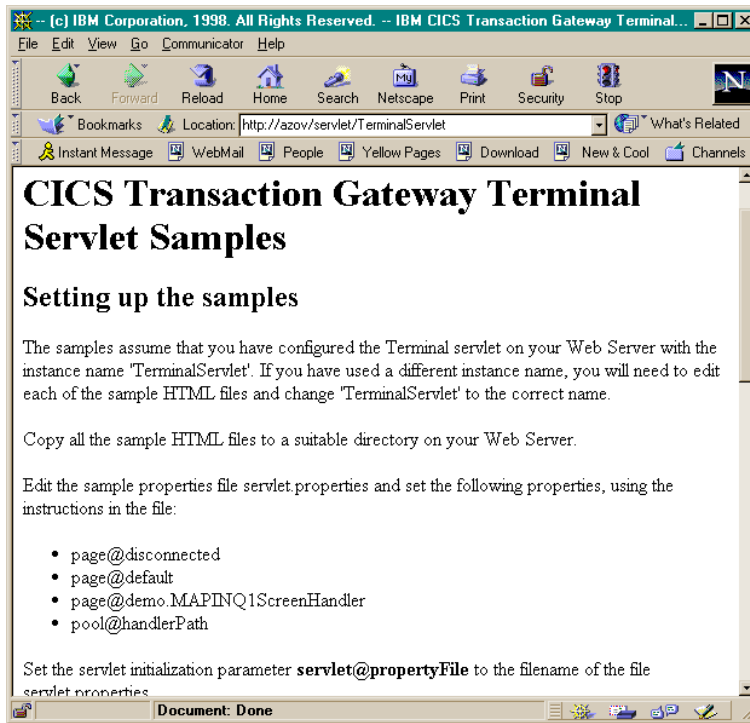


Figure 14. CICS Transaction Gateway Terminal Servlet Samples page

Select the link **Basic Terminal Emulation** on the CICS Transaction Gateway Terminal Servlet Samples page to display the CICS Transaction Gateway Terminal Emulation page. Figure 15 on page 23 shows the CICS Transaction Gateway Terminal Emulation page displaying the CICS CECI transaction.

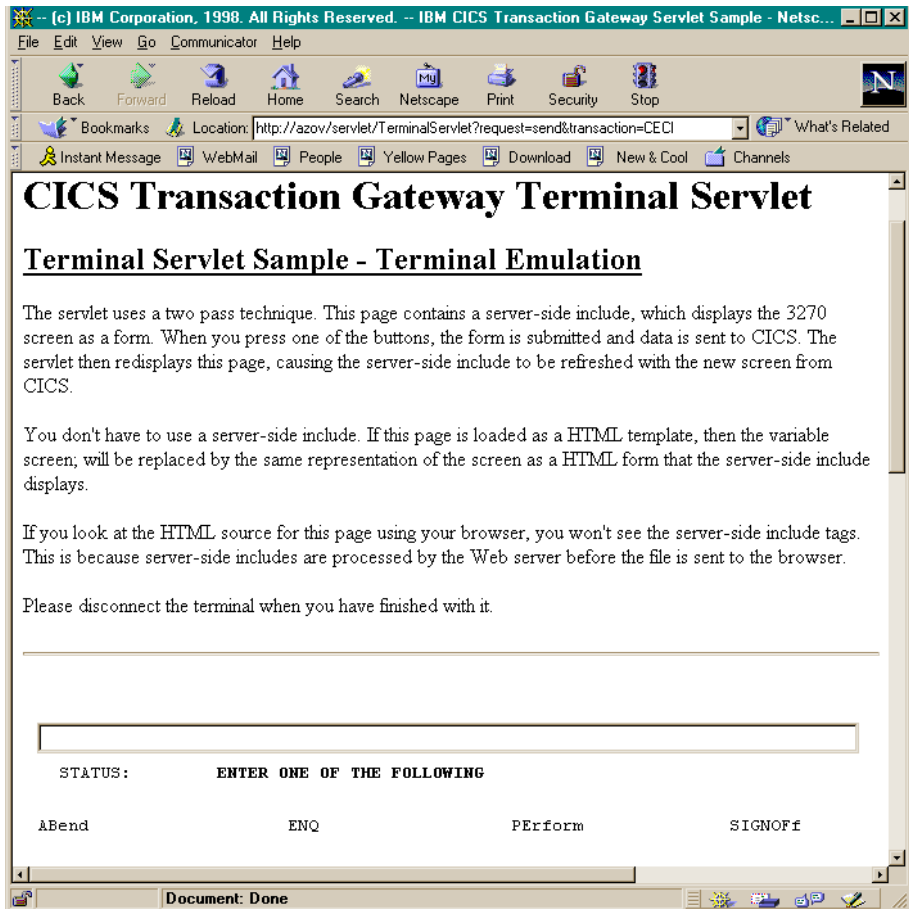


Figure 15. CICS Transaction Gateway Terminal Emulation page

Exit from the terminal emulation screen. Use the **disconnect** button to terminate the CICS session and return to the CICS Transaction Gateway Terminal Servlet Samples page.

Select the link **HTML Template Sample** on the CICS Transaction Gateway Terminal Servlet Samples page to start the EPIC transaction on the CICS server and display the EPIC Transaction page (see Figure 16 on page 24).

Testing using the supplied samples

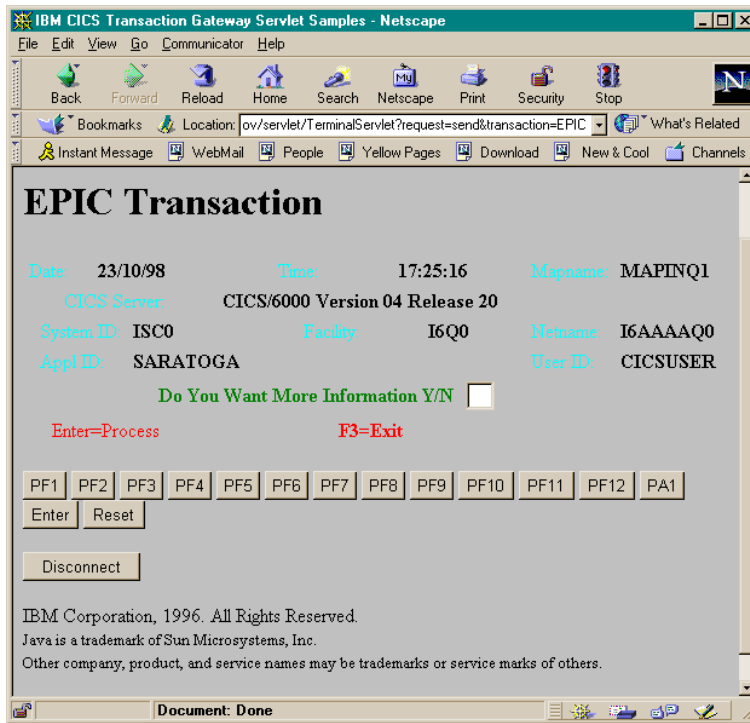


Figure 16. CICS EPIC Transaction page

Exit from the EPIC transaction. Use the **disconnect** button to terminate the session on the CICS server and return to the CICS Transaction Gateway Terminal Servlet Samples page.

Chapter 8. Troubleshooting

In this section we describe some of the facilities available for troubleshooting your configuration.

WebSphere Application Server log files

The following types of logs are generated in WebSphere Application Server Version 3.5:

- stderr.log
- stdout.log
- out-of-process logs for the application server
- out-of-process logs for the admin server
- wasdb2.log
- wssetup.log

These files are located in:

`\WebSphere\AppStudio\logs`

if you selected the default installation directory for WebSphere Application Server.

You should examine these files to find the causes of errors and configuration problems.

For more information about troubleshooting WebSphere Application Server, refer to the URL:

<http://www-4.ibm.com/software/webservers/appserv/doc/v35/ae/infocenter/>

TerminalServlet logging

The TerminalServlet is usually configured for minimal logging and maximum performance. You can increase the amount of TerminalServlet logging written to the log file by modifying the TerminalServlet parameter, **servlet@extendedLogging**.

See “Configuring the parameters” on page 18 for more information on setting the **servlet@extendedLogging parameter**. Set the parameter to true to increase the level of logging. Reset the parameter to false when logging is no longer required.

Troubleshooting

CICS Universal Client troubleshooting

If you experience problems in connecting to the CICS server, you can try the following troubleshooting procedure:

1. If necessary shut down the CICS Transaction Gateway, then shut down the CICS Universal Client with the command `cicscli -x`.
2. Restart the CICS Transaction Gateway and CICS Universal Client.
3. Try running a terminal servlet request that involves connecting to the CICS server.
4. Now enter the `cicscli -l` command. If the message:

```
CCL8043I Server 'NNNNNNNN' (using 'XXXXX' to 'NNNNNN') is unavailable
```

is displayed, then there is a CICS Universal Client/CICS server configuration problem.

If the message:

```
CCL8040I No servers are being used by the CICS Client
```

is displayed, this indicates that requests are not reaching the CICS Universal Client. The problem is therefore a WebSphere/servlet/CICS Transaction Gateway configuration issue.

As an alternative, you can try `cicsterm` the terminal emulator of CICS Universal Client. If this works correctly, then you know that CICS Universal Client is correctly configured.

CICS Universal Client log

The CICS Universal Client writes error messages to the file `CICSCLI.LOG` in the `\bin` directory.

These messages are explained in the *CICS Universal Client Messages* book.

CICS Transaction Gateway tracing

You can display trace information to the CICS Transaction Gateway console by starting the product with a trace parameter. Use the following start command to initiate tracing:

```
ctgstart -trace
```

Using `ctgstart -trace` is only relevant if you are using a network protocol between the terminal servlet and the CICS Transaction Gateway.

The terminal servlet uses the CICS Transaction Gateway API, therefore adding the following Java directive to the JVM that runs the servlet will enable servlet tracing:

```
java -Dgateway.T.trace=on
```

This causes tracing to be written to stderr.

For more information about this kind of tracing, refer to the *CICS Transaction Gateway Programming* book.

CICS Universal Client tracing

You can generate trace information for CICS Universal Client by entering the command:

```
cicscli -d
```

This produces the binary trace file CICSCLI.BIN, from which you can produce a text trace file using the cicsftrc command. For more information, refer to the *CICS Universal Client Administration* book.

Appendix. Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

Anynet	CICS
IBM	OS/390
VTAM	

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, or other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.