iWay

iWay Emulation Adapter (3270/5250) for BEA WebLogic User's Guide
Version 5 Release 5

DN3501487.1204

Preface

This document describes how to use the iWay Emulation Adapter (3270/5250) for BEA WebLogic to invoke mainframe applications from non-mainframe environments, such as UNIX and Windows.

How This Manual Is Organized

This manual includes the following chapters:

Chapter		Contents
1	*Introducing the iWay Emulation Adapter (3270/5250) for BEA WebLogic*	Provides an overview of the iWay Adapter for Telnet and describes how to install it.
2	*Using Remote Procedure Call Mode*	Describes how to use the iWay Adapter for Telnet in RPC mode.
3	*Using Web Mode*	Describes how to use the iWay Adapter for Telnet in Web mode.
4	*Developing Applications Using Emulation and Recorder Modes*	Describes how to use the iWay Adapter for Telnet in Emulation and Recorder modes when developing a Telnet application.
A	*Emulator Keyboard Mapping*	Describes how your keyboard maps to 3270 or 5250 keys when you use the emulator.

Documentation Conventions

The following conventions apply throughout this manual:

Convention	Description
`THIS TYPEFACE` or `this typeface`	Denotes syntax that you must enter exactly as shown.
`this typeface`	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
underscore	Indicates a default setting.
this typeface	Represents a placeholder (or variable) in a text paragraph, a cross-reference, or an important term.
this typeface	Highlights a file name or command in a text paragraph that must be lowercase.
this typeface	Indicates a button, menu item, or dialog box option you can click or select.
Key + Key	Indicates keys that you must press simultaneously.
{ }	Indicates two or three choices; type one of them, not the braces.
[ ]	Indicates a group of optional parameters. None are required, but you may select one of them. Type only the parameter in the brackets, not the brackets.
|	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
. . .	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (…).
. . .	Indicates that there are (or could be) intervening or additional commands.

Customer Support

Do you have questions about iWay Emulation Adapter (3270/5250) for BEA WebLogic?

Call Information Builders Customer Support Service (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 a.m. and 8:00 p.m. EST to address all your iWay Emulation Adapter (3270/5250) for BEA WebLogic questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code (*xxxx.xx*) when you call.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our World Wide Web site, http://www.informationbuilders.com. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of www.informationbuilders.com also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Help Us to Serve You Better

To help our consultants answer your questions effectively, please be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following tables list the specifications our consultants require.

Platform	
Operating System	
OS Version	
Product List	
Adapters	
Adapter Deployment	For example, JCA, Business Services Engine, iWay Adapter Manager
Container Version	

The following table lists components. Specify the version in the column provided.

Component	Version
iWay Adapter	
EIS (DBMS/APP)	
HOTFIX / Service Pack	

The following table lists the types of Application Explorer. Specify the version (and platform, if different than listed previously) in the columns provided.

Application Explorer Type	Version	Platform
Swing		
Servlet		
ASP		

In the following table, specify the JVM version and vendor in the columns provided.

Version	Vendor

The following table lists additional questions to help us serve you better.

Request/Question	Error/Problem Details or Information
Provide usage scenarios or summarize the application that produces the problem.	
Did this happen previously?	
Can you reproduce this problem consistently?	
Any **change in the application environment**: software configuration, EIS/ database configuration, application, and so forth?	

Request/Question	Error/Problem Details or Information
Under what circumstance does the problem *not* occur?	
Describe the **steps** to reproduce the problem.	
Describe the **problem**.	
Specify the **error** message(s).	

User Feedback

In an effort to produce effective documentation, the Documentation Services staff welcomes any opinion you can offer regarding this manual. Please use the Reader Comments form at the end of this manual to relay suggestions for improving the publication or to alert us to corrections. You can also use the Documentation Feedback form on our Web site, http://www.iwaysoftware.com.

Thank you, in advance, for your comments.

iWay Software Consulting and Training

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our World Wide Web site (http://www.iwaysoftware.com) or call (800) 969-INFO to speak to an Education Representative.

Contents

CHAPTER 1

Introducing the iWay Emulation Adapter (3270/5250) for BEA WebLogic

Topics:

- Integrating Mainframe Logic Using the iWay Emulation Adapter (3270/5250) for BEA WebLogic

The iWay Emulation Adapter (3270/5250) for BEA WebLogic enables you to invoke mainframe applications from non-mainframe environments, such as UNIX and Windows, without modifying anything on the mainframe. It enables you to make logic and data associated with legacy mainframe programs available to a new generation of applications.You also can replace green-screen displays with sophisticated user interfaces.

The adapter emulates IBM 3270 and 5250 terminals and enables you to integrate legacy mainframe applications, including those hosted by IBM z/OS and OS/400 systems.

Integrating Mainframe Logic Using the iWay Emulation Adapter (3270/5250) for BEA WebLogic

Telnet is a standard protocol for remote login over the Internet. (The name is an abbreviation of Telecommunications Network.) When a user establishes a Telnet connection to a remote mainframe computer, the user's local system emulates a terminal connected directly to the mainframe.

The iWay Emulation Adapter (3270/5250) for BEA WebLogic emulates IBM 3270 and 5250 terminals and adds "screen scraper" functionality. A screen scraper intercepts character-based data from a mainframe application to a terminal and passes the data to either:

- a user via a more sophisticated graphical user interface (GUI), such as a Web page.
- a client application, in order to automate interaction between the mainframe application and the client application.

For information about deploying the adapter and about the iWay Connector for JCA and the iBSE servlet, see the *iWay Installation and Configuration for BEA WebLogic* documentation

You define the type of emulation you want using the adapter's Telnet Designer.

You can use the adapter in several ways when developing and running applications that access a mainframe:

- **RPC mode** In a production environment, your application can communicate with a mainframe session using files. The application sends keystrokes to the mainframe in an XML request document, and the mainframe returns screen data in an XML response document. You can use these documents to integrate a mainframe session into your application. This is called RPC (remote procedure call) mode and is supported when you deploy the adapter through the iWay Connector for JCA or the iWay Business Services Engine servlet. RPC mode is described in Chapter 2, *Using Remote Procedure Call Mode*.
- **Web mode** In a production environment, you can provide an application user at a Web browser with live access to the mainframe. You can modify the appearance of the mainframe screens in the browser to create a common look-and-feel and to expand the functionality of your online application. This is called Web mode and is described in Chapter 3, *Using Web Mode*.

  You can choose to convert terminal display characteristics to the Web using a default mapping, using your own customized template, or using JavaServer Pages (JSP). JSP also enables you to modify the flow and logic of the mainframe screens.

- **Emulation mode** During development, you can access a mainframe session directly from your workstation using the Telnet Designer, which displays standard 3270 and 5250 screens. This is called Emulation mode and is described in *Emulation Mode* in Chapter 4, *Developing Applications Using Emulation and Recorder Modes*.
- **Recorder mode** During development, you can record interaction with a mainframe and later play it back to simulate a live mainframe session. Simulating a session enables you to develop an application offline. You can simulate a Web mode connection or an RPC mode connection. This simulation is called Recorder mode and is described in *Recorder Mode* in Chapter 4, *Developing Applications Using Emulation and Recorder Modes*.

CHAPTER 2

Using Remote Procedure Call Mode

Topics:

- Using Remote Procedure Call Mode
- Creating a Transaction Module and Submitting a Request

The Remote Procedure Call (RPC) mode enables your application to communicate with a mainframe session using files. The application sends keystrokes to the mainframe in an XML request document, and the mainframe returns screen data in an XML response document.

Using Remote Procedure Call Mode

In a production environment, the iWay Adapter for Telnet Remote Procedure Call (RPC) mode enables your client application to communicate with a mainframe application using XML files.

You can access the adapter in RPC mode using:

- **J2EE™ Connector Architecture (JCA)**. You must deploy the adapter to a JCA-compliant application server to which you also deployed the iWay Connector for JCA.
- A **Web service.** You must deploy the adapter to an application server to which you also deployed the iWay Business Services Engine (iBSE) servlet.

  You can combine JCA and Web service environments by deploying the iBSE servlet to a JCA-compliant application server.

For more information about deployment, see *iWay 5.5. Installation and Configuration for BEA WebLogic*.

At design time, you can create a transaction module for your mainframe application after you deploy and test the adapter.

At run time, your client application can:

- **Send information *to* a mainframe session.** The client application sends information required for navigating through the mainframe application to the application in an XML request document.
- **Receive information *from* a mainframe session.** The mainframe application returns screen data to the client in an XML response document.

Creating a Transaction Module and Submitting a Request

A transaction module is an XML request document that defines a transaction. It contains all of the information required to execute an online mainframe application: screen references, entered data, menu selections, and other keystrokes.

After the transaction module is saved and generated as a request document, you can deploy it as a business service (also known as a Web service).

Creating a Transaction Module

You can create a transaction module at design time. After you save the module, it automatically is generated in the form of a request document. Then, you can deploy the transaction module as a business service (also known as a Web service).

Procedure How to Create a Transaction Module

To create a transaction module at design time:

1. Connect to the mainframe application using Telnet Designer.
2. Navigate through the application's screens.

   For more information, see *How to Create a Transaction for a Mainframe Application* on page 2-4.

   For each screen:

   a. Identify the screen so that the adapter recognizes it at run time.

      For more informatiion, see *How to Identify a Screen to the Transaction Module* on page 2-7.

   b. Define each input field that requires a value at run time as an input parameter and specify the value.

      At run time, the request document provides the value to the field on the screen.

   c. Define each output field required by the client application as an output parameter.

      At run time, output parameter values are returned to the client application through the response document.

      For information on input and output parameters, see *How to Define an Input or an Output Screen Parameter* on page 2-10.
3. Save your work.

   Saving automatically generates the transaction module in the form of a request document.
4. To deploy the transaction module as a business service (also known as a Web service):

   a. Open iWay Servlet Application Explorer and connect to a Telnet target.

   b. Import the transaction module.

   c. Generate the Web service.

Reference Submitting a Telnet Request at Run Time

To execute a transaction at run time, you can submit a Telnet request to:

- **The iWay Connector for JCA**

  The client application sends the request document (including the transaction information) to the connector, which forwards it to the iWay Adapter for Telnet. The adapter then executes the transaction.

- **iWay Business Services Engine (iBSE) servlet**

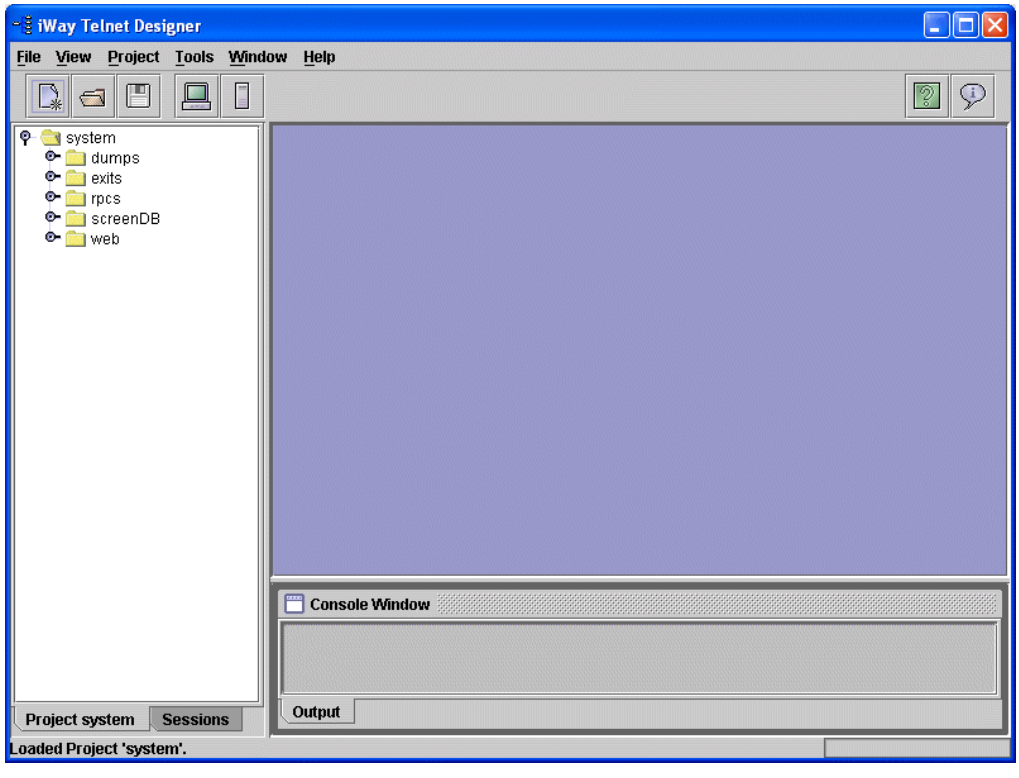
  The client application sends the request document (including the transaction information) to the iBSE servlet, which forwards it to the iWay Adapter for Telnet. The adapter then executes the transaction.

Procedure How to Create a Transaction for a Mainframe Application

To create a transaction for a mainframe application:

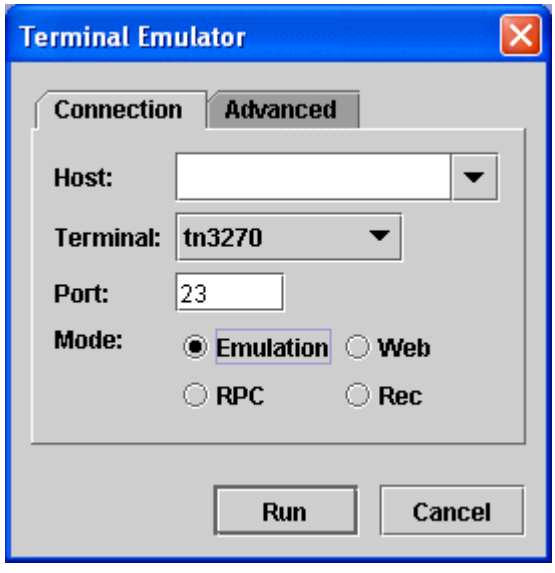
1. To open Telnet Designer, click the Start menu and select *Programs*, *iWay 5.5*, and then *iWay Telnet Designer.*

   Telnet Designer opens.



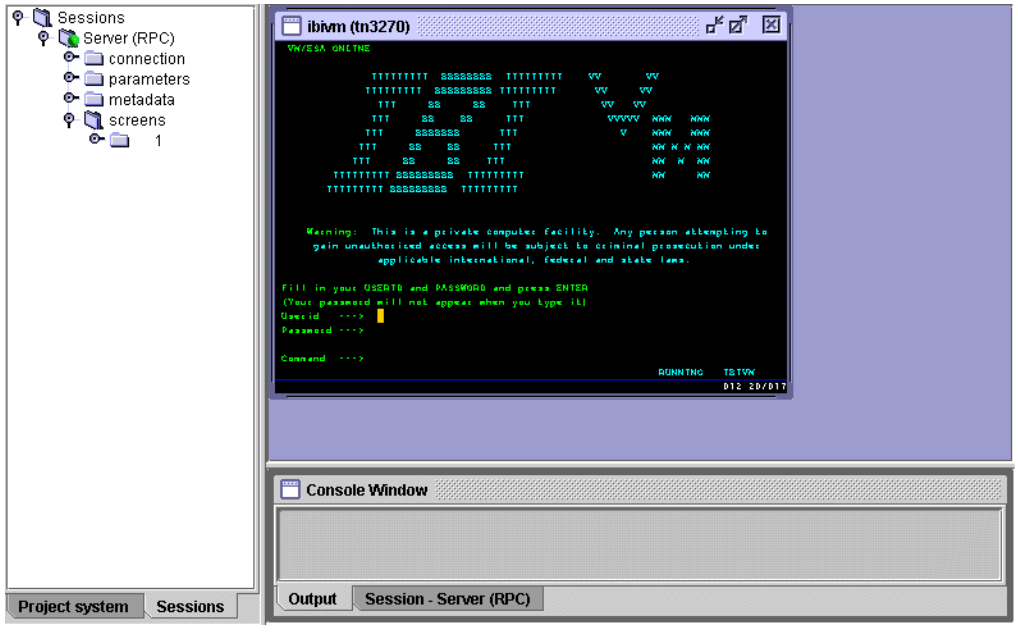
2. From the Tools menu, select *Emulator.*

The Terminal Emulator properties dialog box opens.



3. Specify the emulation properties:
   a. Type a new host name, or select an existing host name, for the computer on which the mainframe application resides.
   b. From the Terminal drop-down list, select the type of emulation.
   c. Type the number of the terminal port.
   d. For Mode, select the *RPC* option.
4. Click the *Advanced* tab.
   a. To specify a different national language character set, select *Extended Attributes*.
   b. Select the desired language.
5. To connect to the mainframe application and start the emulation session, click *Run*.

The following pane opens, displaying a VTAM session:



You can enlarge the emulator pane using the size controls in the upper right.

6. To ensure that the adapter recognizes the screen when it encounters it at run time and executes a request, identify each screen.

   a. Record all keystrokes to be played back at run time.

   b. For more information, see *How to Identify a Screen to the Transaction Module*.

7. For each screen, define the following:

   - **Input fields** that require values as input parameters at run time.

     These are to be provided by the request document.

   - **Output fields** that are returned by the response document as output parameters.

   For more information, see *How to Define an Input or an Output Screen Parameter* on page 2-10.

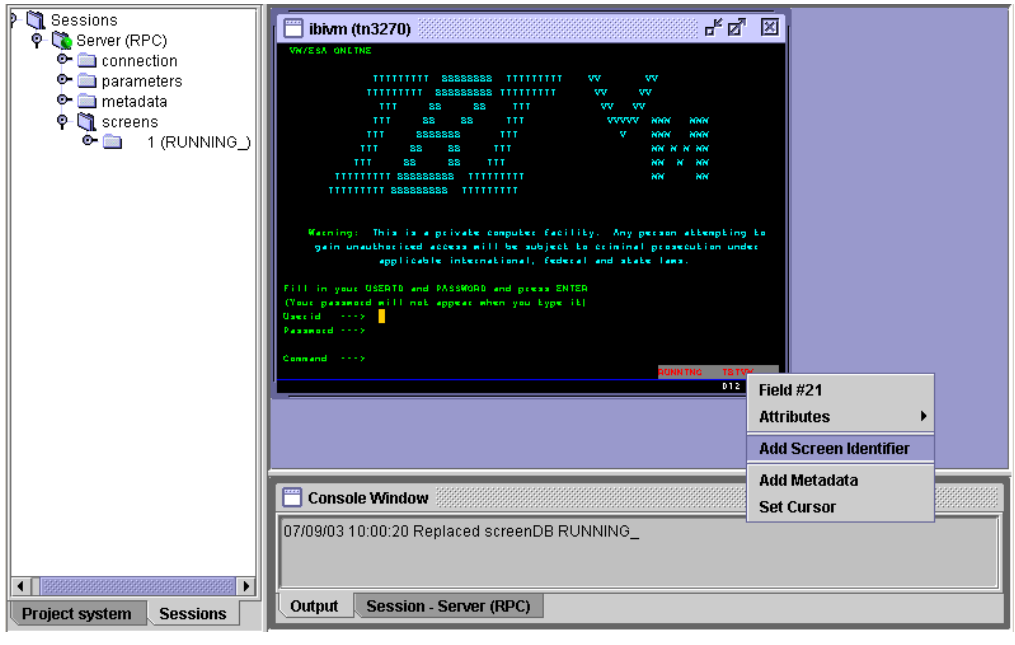
Identifying a Screen to the Transaction Module

You must identify each screen to the transaction module. You identify each screen only once. After you identify the screen, identify the values that may change each time the transaction is invoked.

Procedure How to Identify a Screen to the Transaction Module

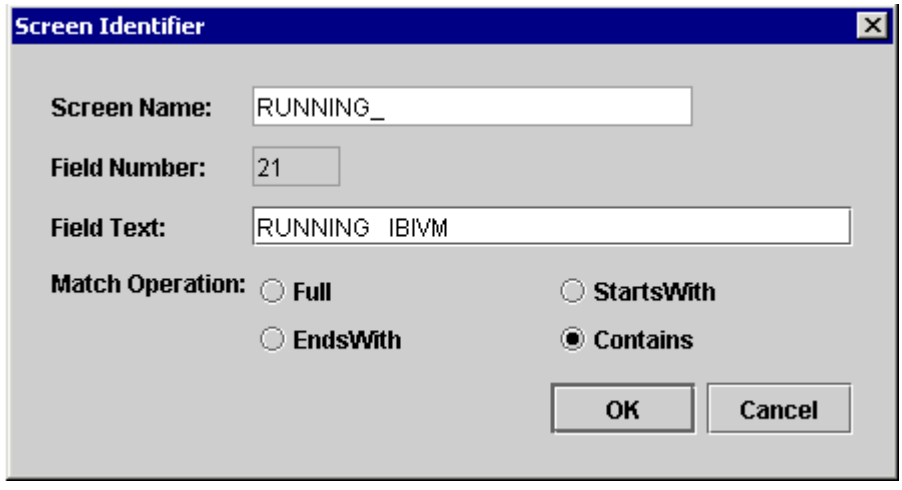
To identify a screen to the transaction module:

1. Determine the field(s) on the 3270 screen that distinguish this screen from the application's other screens.
2. If there are several fields, choose one to identify the screen.



3. Right-click the identifying field and select *Add Screen Identifier*.

The Screen Identifier dialog box opens.



a. In the Screen Name field, type a descriptive name for the screen you want to identify.

b. Type a value for Field Number.

c. In the Field Text box, type the text in the identifying field that uniquely identifies this screen.

d. For Match Operation, select an option that describes the relationship of the text you entered to the field's complete text.

For example, if the field's value is "OPERATOR INSTRUCTIONS," and you enter "OPERATOR INSTRUCTIONS," the field text you entered is the full text of the field on the screen and you would select the Match Operation option, Full.

If you enter "Instructions" for the same field, you would select the Match Operation option, EndsWith or Contains.

When a match is made, the adapter proceeds with the key pressed, such as Enter or a PF key. At run time, the screen is processed accordingly.

4. Click *OK*.

You have identified the screen to the transaction module. You must identify each screen in this way. You identify each screen only once.

After you identify a screen, identify the values that may change each time the transaction is invoked.

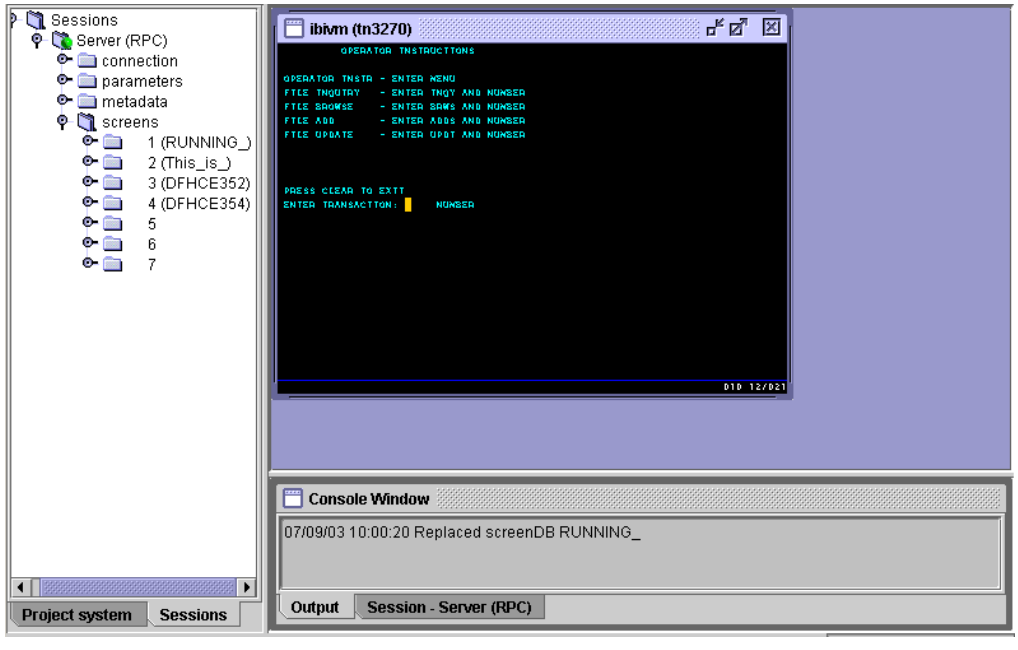
For more information, see *How to Define an Input or an Output Screen Parameter*.

5. Continue navigating through the application and ensure that you identify all of the screens.

   You are not required to identify a blank screen, such as the one you encounter after you navigate to CICS.

6. Enter the CICS transaction MENU and press *Enter*.

7. When the following menu screen opens, add a screen identifier to identify the screen.



Defining an Input or an Output Screen Parameter

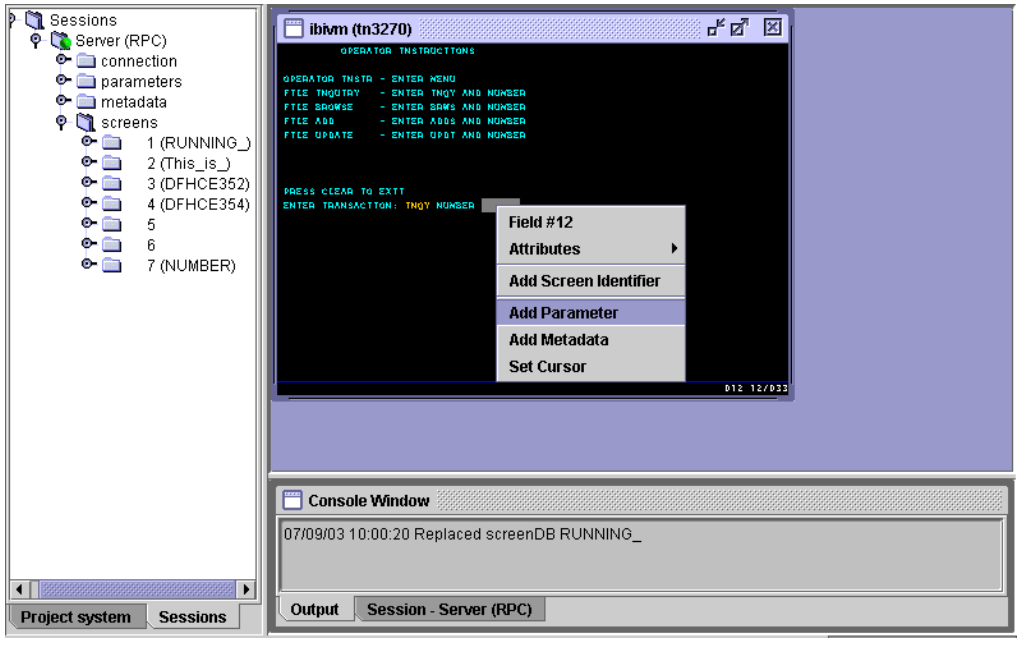
The following procedure describes how to define input and output screen parameters. Output fields are returned as part of the answer set by the transaction module you create.

Procedure How to Define an Input or an Output Screen Parameter

In the following procedure, INQY is a sample transaction, and NUMBER is a sample input parameter.

To define input and output parameters for a screen:

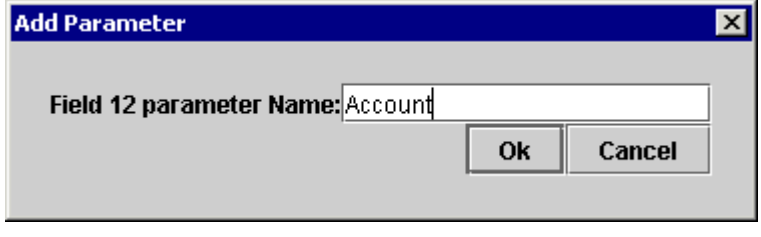
1. In the ENTER TRANSACTION field, type *INQY*.



a. Move the cursor to the *NUMBER* field.

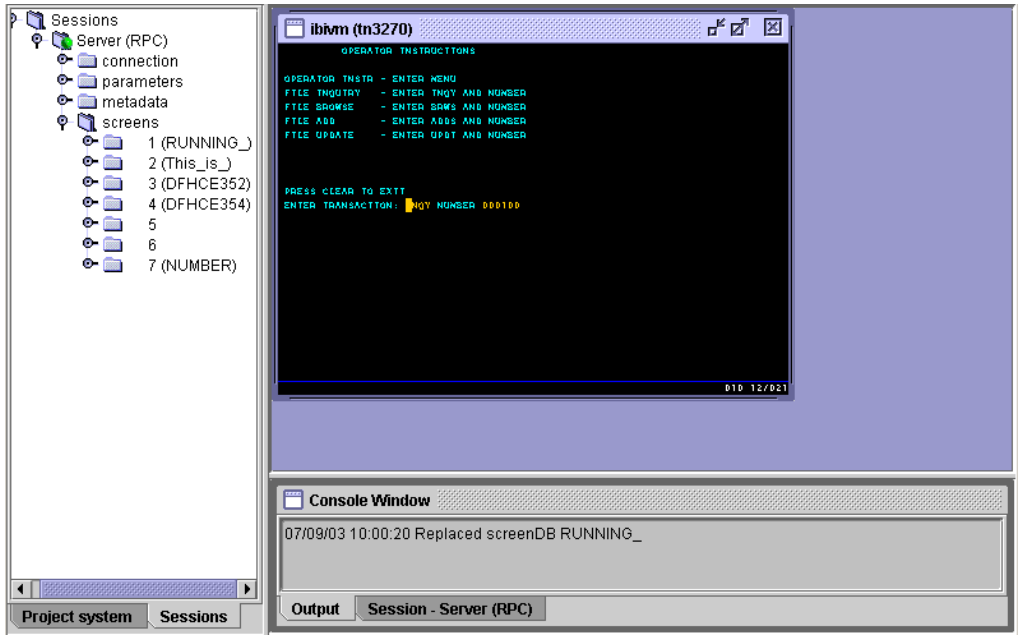
b. Right-click and select *Add Parameter*.

The Add Parameter dialog box opens.



2. Type a name, for example, Account, and click *OK*.

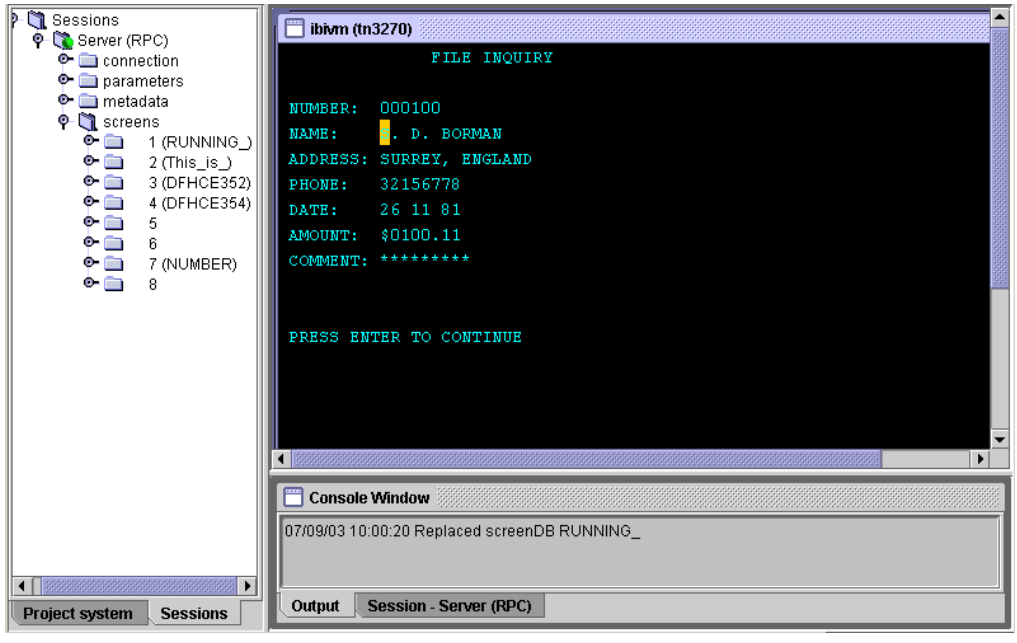
You return to the main screen.



a. In the Number field, type an existing account number, for example, 000100.

b. To view the detail pane, press the *Enter* key.

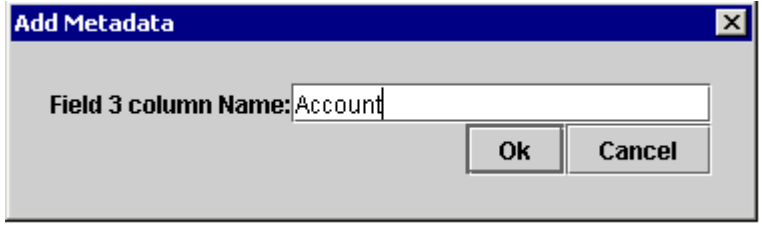
The detail pane opens.



The detail pane contains information that you must describe as output. The output fields are returned as part of the answer set by the transaction module you create.

3. Right-click one of the output fields, for example, NUMBER, and select *Add Metadata*.

The Add Metadata dialog box opens.



4. Type a name, for example, Account and click *OK*.

5. Repeat the steps to identify input and output parameters for the remaining fields that you want returned as output (two, in this example).

6. To end the emulation session, navigate through the screens.

a. Log off the CICS region.

b. To close the emulator, click the *X* in its upper-right corner.

7. Right-click the *Server (RPC)* icon and select *Save As*.

   The Save As dialog box opens.

   a. Type a name for the transaction module.

   b. Type an optional comment.

      The comment is stored in the remarks element of the module.

8. Click *Save*.

   The transaction module (also known as the request document) is saved as an XML document to

   DesignerDir\tools\projects\system\rpcs

   where:

   DesignerDir

   Is the directory where Telnet Designer is installed.

   A folder representing the new transaction module appears in the left pane under the Sessions folder.

9. To close Telnet Designer, click the *X* in the upper-right corner.

Example Modifying an Input Parameter in an XML Request Document

You can vary a request by modifying the input parameters in the XML request document. Input parameters are enclosed in parameter tags, for example:

```
<parameters>
   <parameter NAME="Account" LENGTH='6'>000100</parameter>
</parameters>
```

To alter the request, you change parameter values. For example, you may change the value of Account from 000100 to another account number.

CHAPTER 3

Using Web Mode

Topics:

- Overview
- Default HTML Translation
- Using Telnet Designer to Create JavaServer Pages
- Using HTML Translation Services
- Creating an HTML Page to Start a Telnet Session

For the iWay Emulation Adapter (3270/5250) for BEA WebLogic, Web deployment refers to creating a Web application using Telnet Designer. You can create Web applications using one of the following techniques:

- Default HTML translation
- Default HTML translation using a generic template
- Redesign using JavaServer Pages (JSP)

The following topics describe how to create a Web-based application to run a mainframe CICS application on the Web.

Overview

In Web mode, Telnet Designer enables you to create a Telnet application using the following techniques:

- **Default HTML translation.** You can run your 3270 or 5250 application instantly on the Web using default emulation. A screen that is "not recognized" appears using default HTML translation.
- **Default HTML translation templates.** You can create a generic template that provides a customized look for all default translated screens (unrecognized screens). You can create a common template that surrounds your HTML translated mainframe screen.
- **JavaServer Pages (JSP) for redesign and added functionality.** You can redesign the look and feel of a mainframe screen and expand the functionality of your online application by combining multiple screens or extracting screen data to one or more newly designed JSP pages.

A thorough understanding of each topic is recommended before moving to the next topic. In addition, before continuing to the following topic, see *Using HTML Translation Services* on page 3-37 for information about HTML Translation Services.

Note: When you create your Web application using Telnet Designer, you can immediately deploy your application using the supplied Web application server.

Default HTML Translation

Default HTML translation automatically translates your terminal screens into HTML pages. The HTML translation mode manages the various Telnet sessions (connections) to the mainframe.

You can create a generic template that surrounds the translated mainframe screens. You use the generic template for default HTML translation only. It provides a customized look for default translation and provides cosmetic changes only. To move fields or completely redesign the screen, you must use JavaServer Pages. For more information, see *Using Telnet Designer to Create JavaServer Pages* on page 3-26.

There are two main servlets for default HTML translation:

- **TelnetServlet** is the starting point for all of the features in the HTML translation. The servlet creates and manages Telnet sessions. It uses the cmd parameter and other parameters, depending on the requested command.
- **TelnetHtmlServlet** processes screens for an active session.

Creating a Web Application Using Default Translation

The following sample procedure describes how to create a Web application using default translation and how to execute this application from a Web browser. The steps required for your site may differ, but the concept is the same.

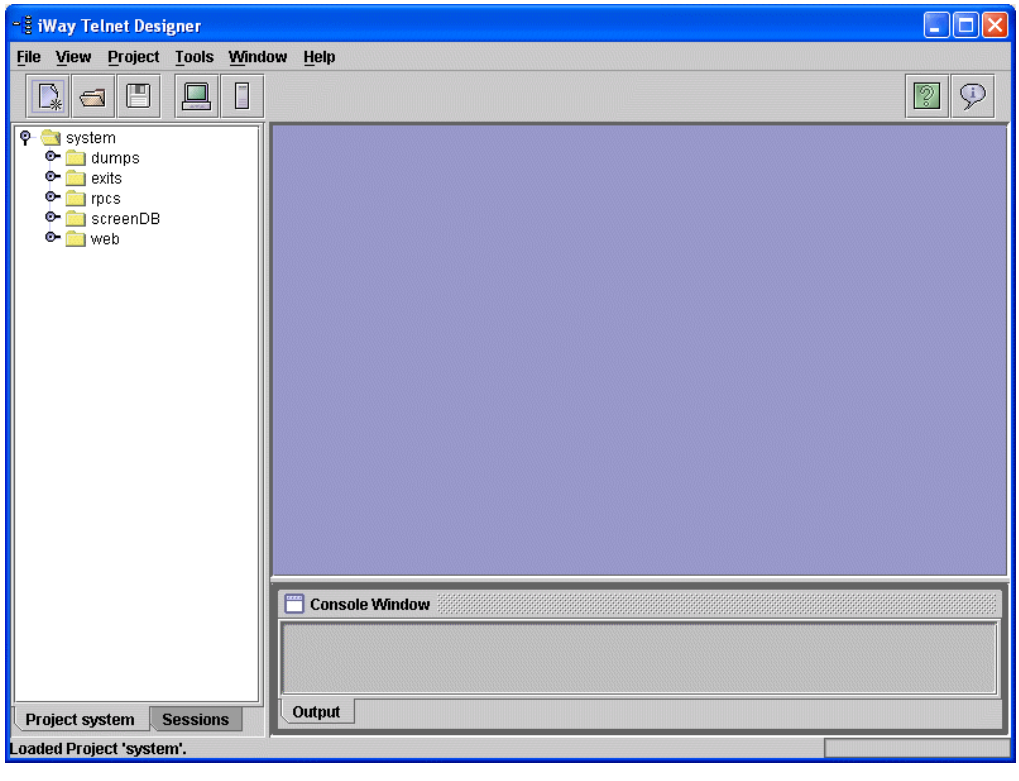
The screens that illustrate the sample procedure represent a portion of a mainframe CICS system. When you run the application on the Web, these screens and other screens (not "seen" by Telnet Designer) are automatically translated.

Procedure How to Create a Web Application Using Default Translation

To create a Web application using default translation:

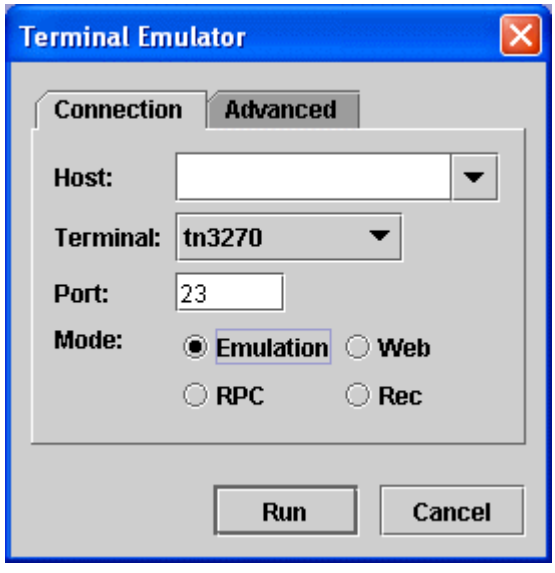
1. To open Telnet Designer, click the Start menu and select *Programs*, *iWay 5.5*, and *iWay Telnet Designer*.

   The Telnet Designer opens.



2. Select *Emulator* from the Tools menu.

The Terminal Emulator properties dialog box opens.



a. In the Host field, type a new host name or select an existing host name for the computer on which the mainframe application resides.

b. From the Terminal drop-down list, select the type of emulation.

c. Type the number of the terminal port.

d. For Mode, select the *Web* option.

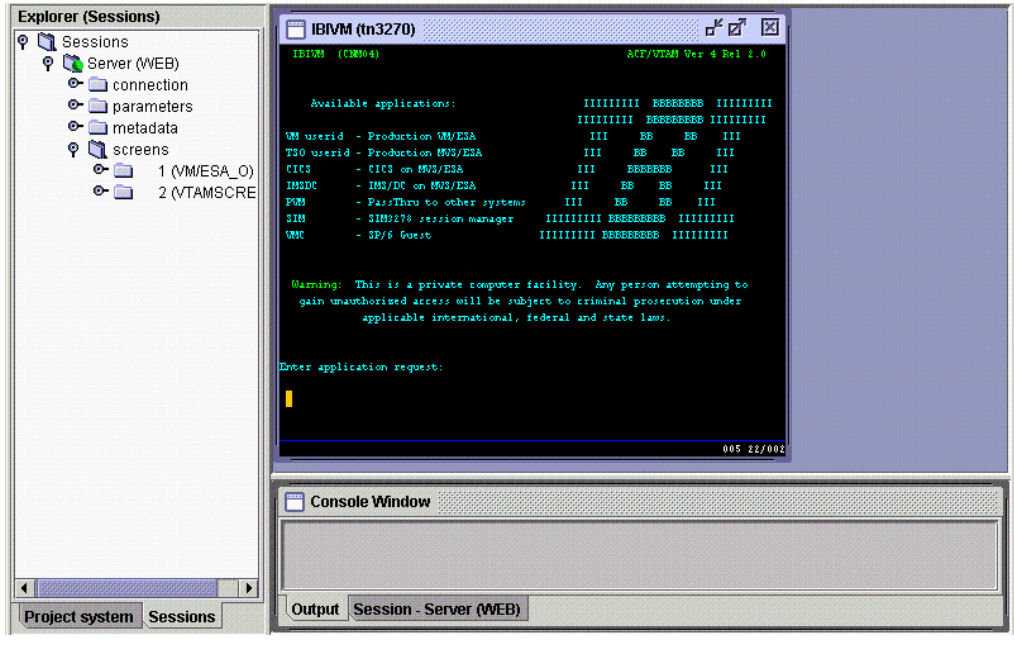
3. Click the *Advanced* tab.

a. To specify a different national language character set, select *Extended Attributes*.

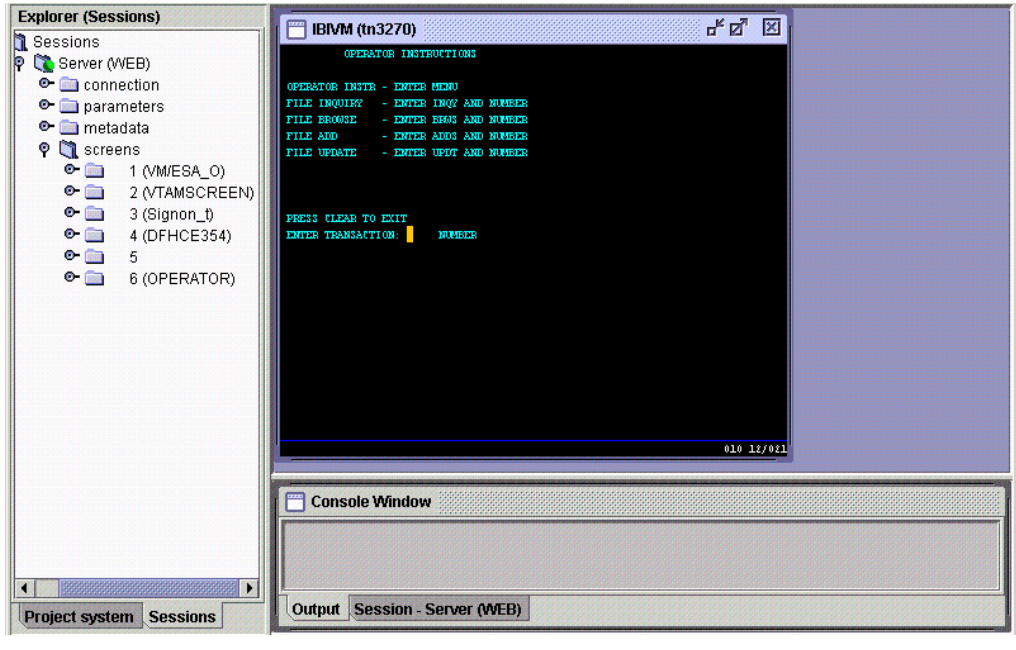
b. Select the desired language.

4. To connect to the mainframe application and start the emulation session, click *Run*.

The following pane opens, displaying a VTAM session.



5. To return to a currently running CICS region, type *LOGON APPLID(IWAYDEMO)*.



a. Enter a valid user ID and password for the mainframe.

b. If the CICS region is running unsecured, press the clear key to navigate to a blank CICS screen.

6. In the blank screen, enter *MENU* to invoke the IBM CICS MENU APPLICATION.

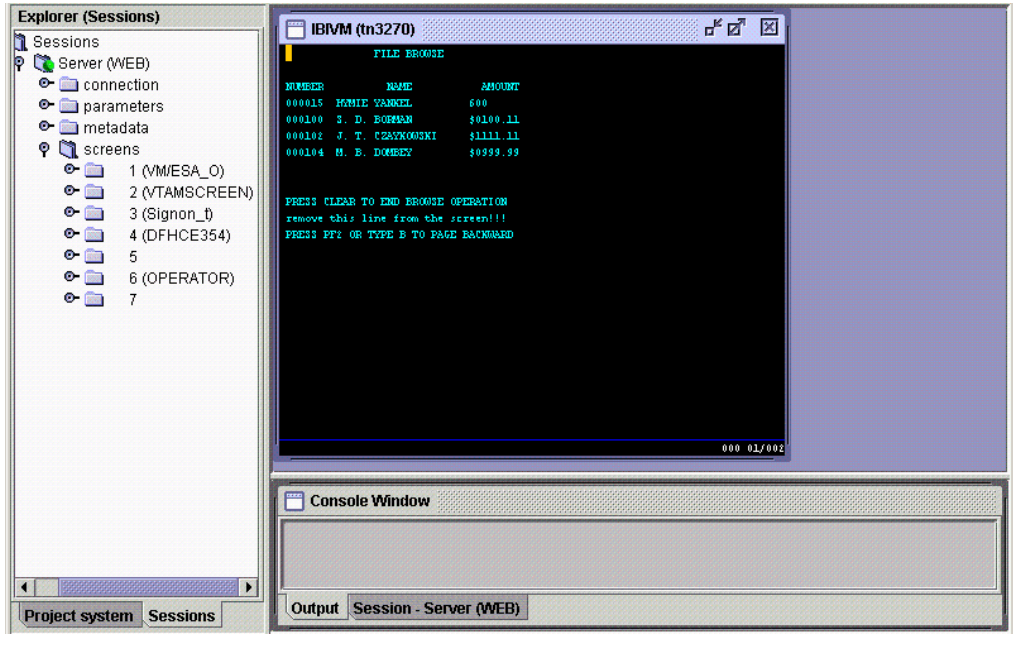
This application, specifically the BROWSE transaction, is transformed into a Web application using default translation.

You can select from several options, for example:

- **INQY** queries a particular VSAM record.
- **BRWS** browses the file.

7. Select *BRWS*.

The following pane opens.



Four records appear.

You can now create the iWay Adapter for Telnet application.

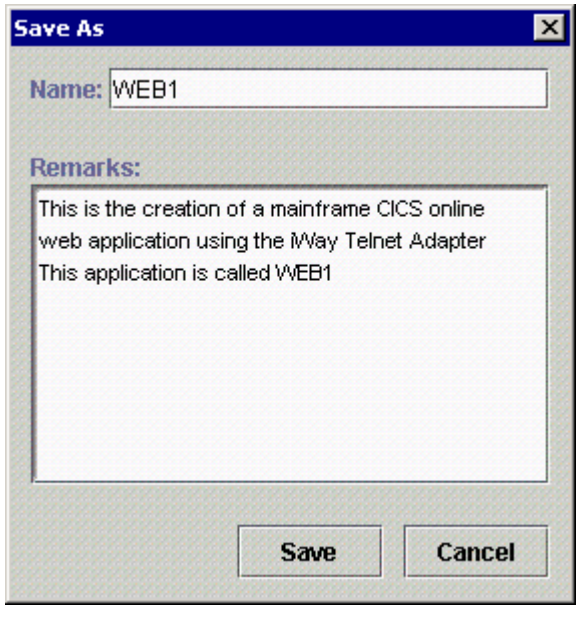
Procedure How to Create the iWay Adapter for Telnet Application

To create the iWay Adapter for Telnet application:

1. To log off CICS, enter the command *CESF LOGOFF*.

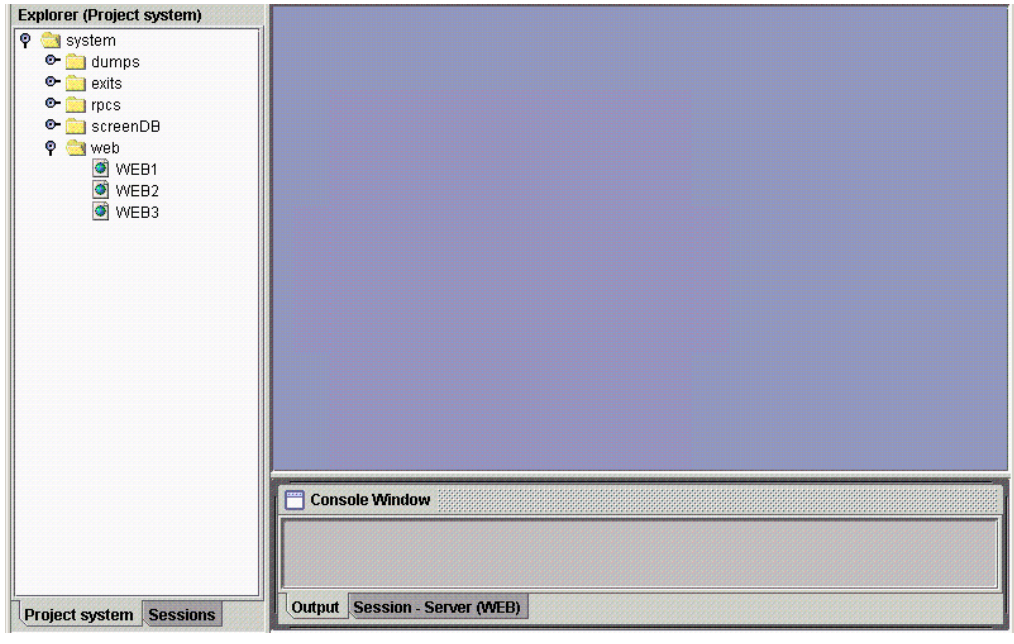
2. To close the screen, click the *X* on the upper-right corner of Telnet Designer.
3. In the left pane, right-click the *Server (WEB)* folder under Sessions and select *Save As*.

   The Save As dialog box opens.



4. Type a name, for example, WEB1.
5. To save the adapter application called WEB1, click *Save*.

You return to Telnet Designer.



6. In the lower left corner, click the *Project system* tab.

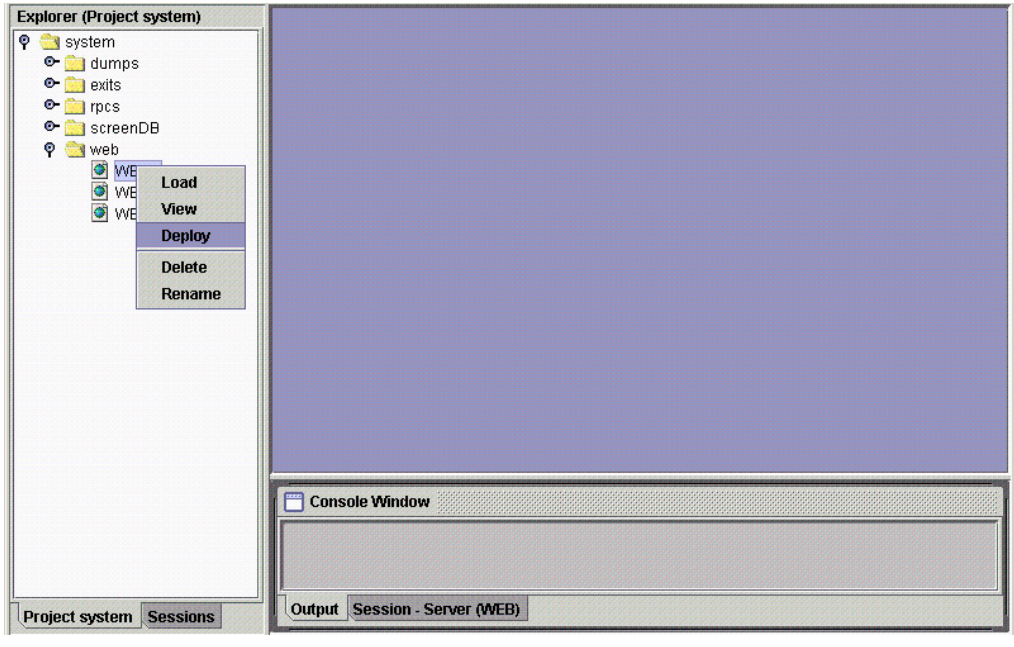
   The Web application called WEB1 appears in the left pane.

Testing the Web Application

You can test an adapter application by deploying the Web application to the supplied Web server.

Procedure How to Test the iWay Adapter for Telnet Web Application

To test the Web application, WEB1, that you created previously:



1. Right-click *WEB1*.
2. Select *Deploy*.

   The Deploy option automatically transfers the Web components created by Telnet Designer to the supplied Web application server.

   An internal XML rules file used by the Designer, called WEB1.XML, is created and placed in the following directory:

   iWay55InstallDir/tools/Telnet/projects/system/web

   A directory called ibitelnet/WEB1 is created within the webapps directory of the Web server.

   The WEB1 folder is empty as you are using default HTML translation.

   If you created JavaServer Pages (JSP), they are placed in the WEB1 folder when you deploy the application. For more information about JavaServer Pages, see *Using Telnet Designer to Create JavaServer Pages* on page 3-26.

Running the Web Application

The iWay Adapter for Telnet must locate the JDK or JRE file. Otherwise, a NoJavaLangClass error appears in the browser. If this error occurs, set the java_home variable in the iwaytelnet.bat file to the location of your JDK or JRE file, for example,

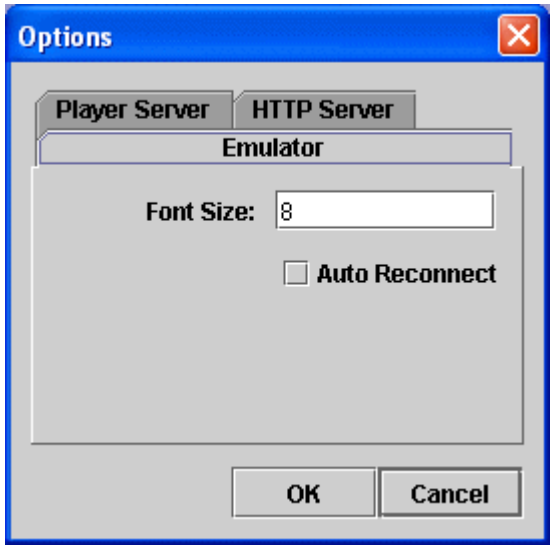
```
SET JAVA_HOME= drive:\jdk1.3
```

Procedure How to Start the Supplied Web Server to Run the Web Application

To start the Web server:

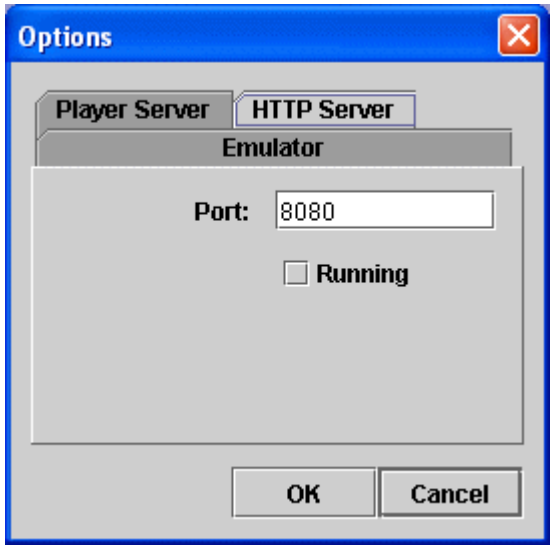
1. In Telnet Designer, click *Tools* and then select *Options*.

   The Options dialog box opens.



2. Click the *HTTP Server* tab.

The HTTP Server tab appears.



a. Type the port number on which the Web server is listening.

b. Select the *Running* check box.

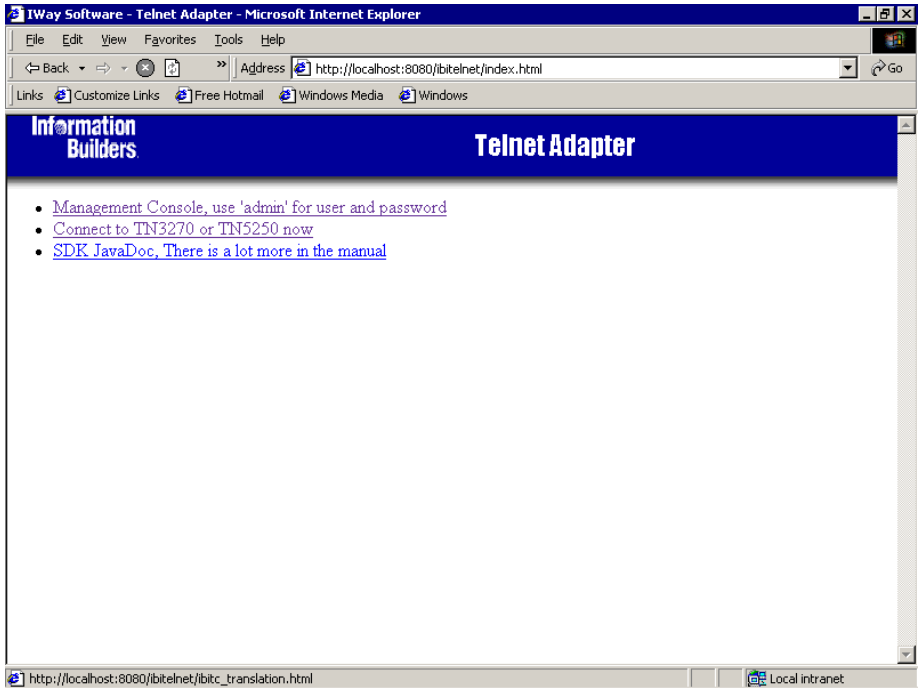
3. To start the Web server, click *OK*.

Procedure How to View the iWay Adapter for Telnet Web Application

1. In a Web browser that has access to the machine where the iWay Adapter for Telnet is running, enter the following URL:

`http://domain:port/ibitelnet/index.html`

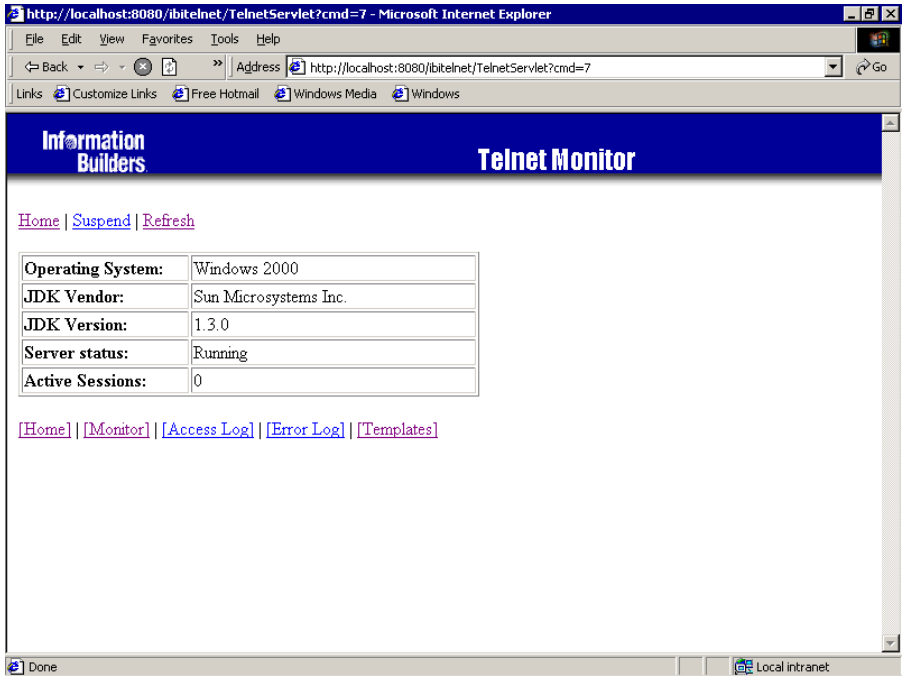
The iWay Adapter for Telnet default HTML window opens.



This window contains sample Web applications that connect to CICS and VM systems in New York.

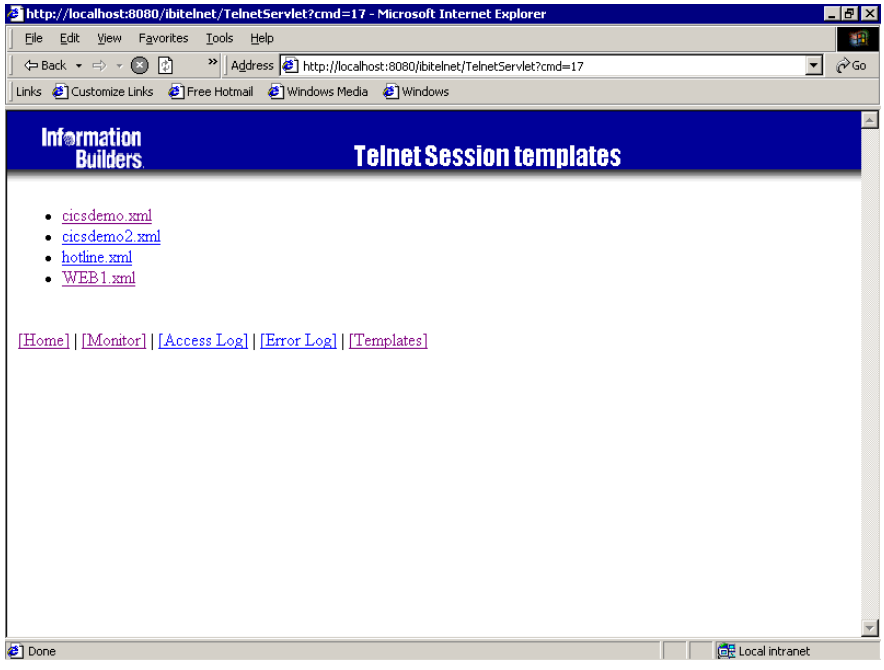
2. Click the hyperlink called *Management Console*; *use 'admin' for user and password*.

The iWay Adapter for Telnet console opens.



3. To view the iWay Adapter for Telnet applications that you created previously, including WEB1, click *Templates*.

The list of templates appears in the Telnet Session templates window.

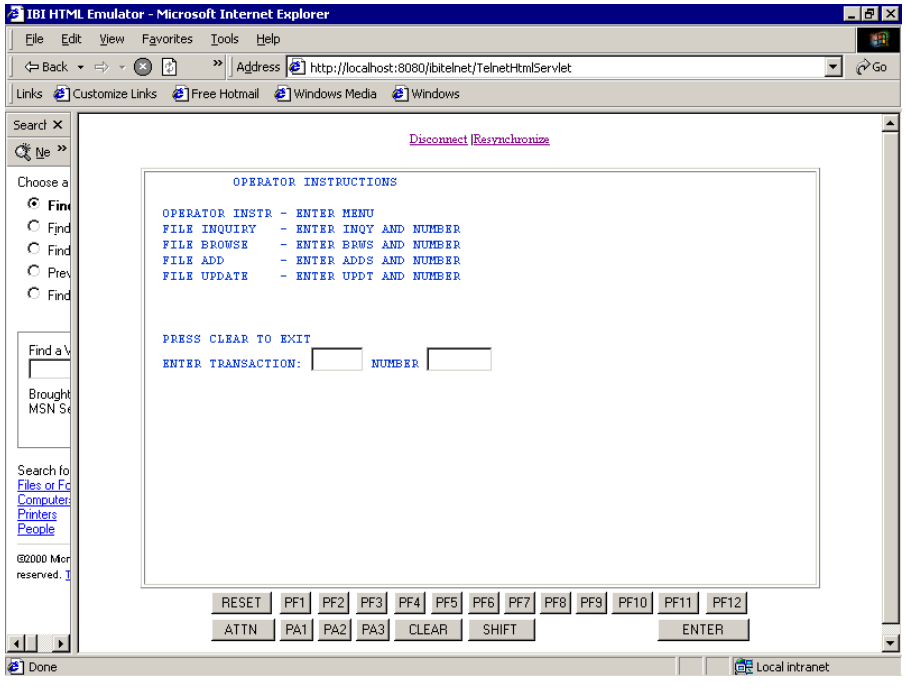


4. Click *WEB1.xml* to navigate through the screens in default emulation mode.

Example Using HTML Translation for Mainframe Screens

The following screen illustrates a default HTML translation of the mainframe application. The online application is running in the back-end. Function keys and other buttons appear at the bottom of the screen

Templates and Java Server Pages were not used. This screen illustrates default translation and the ease of development, Web deployment, and execution.



Running Telnet Designer in Web Mode: Default Translation

In the procedure of following topic, *Using the Add Screen Identifier Feature*, you create a second iWay Adapter for Telnet Web application (WEB2). In this procedure, all the screens are hidden except for the final Browse screen. This technique is called *scripting* or *autologon*. The Web application may seem to skip screens, but in reality, the screens (and the application programs behind the screens) are running on the mainframe but are invisible to you.

When you click the application, the adapter runs through all of the screens except for the final Browse screen. As a result, an application is created that runs through all the previous screens, streamlining the mainframe online session. The same procedure as just described is used for each screen that is to be hidden.

You must identify each screen so that the application "records" your keystrokes and data. To identify the screens, you must choose a portion of the screen that is unique to each mainframe screen. The unique portion enables the iWay Adapter for Telnet to determine what action to take when the screen becomes active as part of a 3270 or 5250 data stream. You can select options for how the iWay Adapter for Telnet identifies the screen when you identify the screen using a field or literal on the screen.

Using the Add Screen Identifier Feature

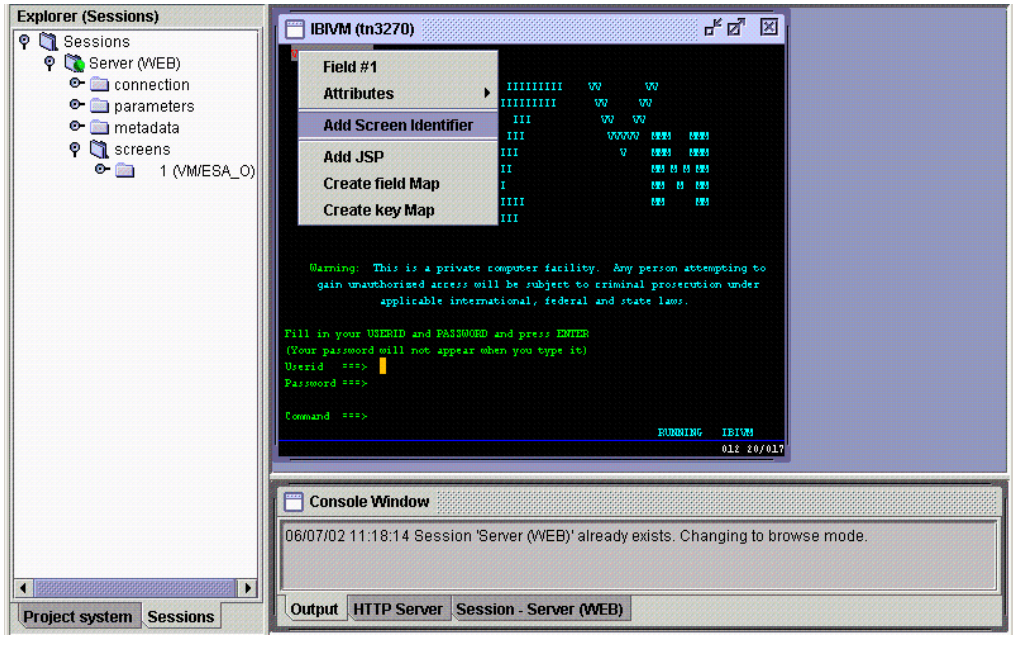
The Add Screen Identifier feature enables you to add patterns (filters) based on screen fields. You can click the region of the screen that uniquely identifies the current screen. Telnet Designer highlights the field, and a dialog box appears with additional string matching capabilities. You can enter a screen name and after you click OK, the screen database updates.

By adding a screen identifier, the iWay Adapter for Telnet records the screen and the key data entered in a local database. After the screen is identified, it is recognized by the iWay Adapter for Telnet.

Procedure How to Add a Screen Identifier

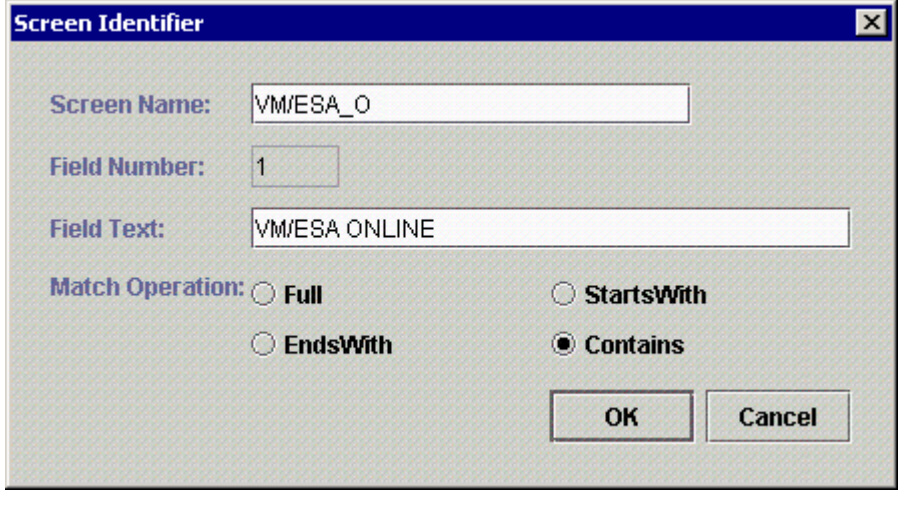
The first screen that appears when connecting to the mainframe is the VTAM screen. This screen always appears, and the first literal line on top of the screen is (CM04) ACF VTAM VER 4. This is an example of a screen identifier.

To add a screen identifier:



1. When the VTAM screen appears in Telnet Designer, right-click the top literal field and select *Add Screen Identifier*.

The Screen Identifier menu appears.



a. In the Screen Name field, type a name for the screen.

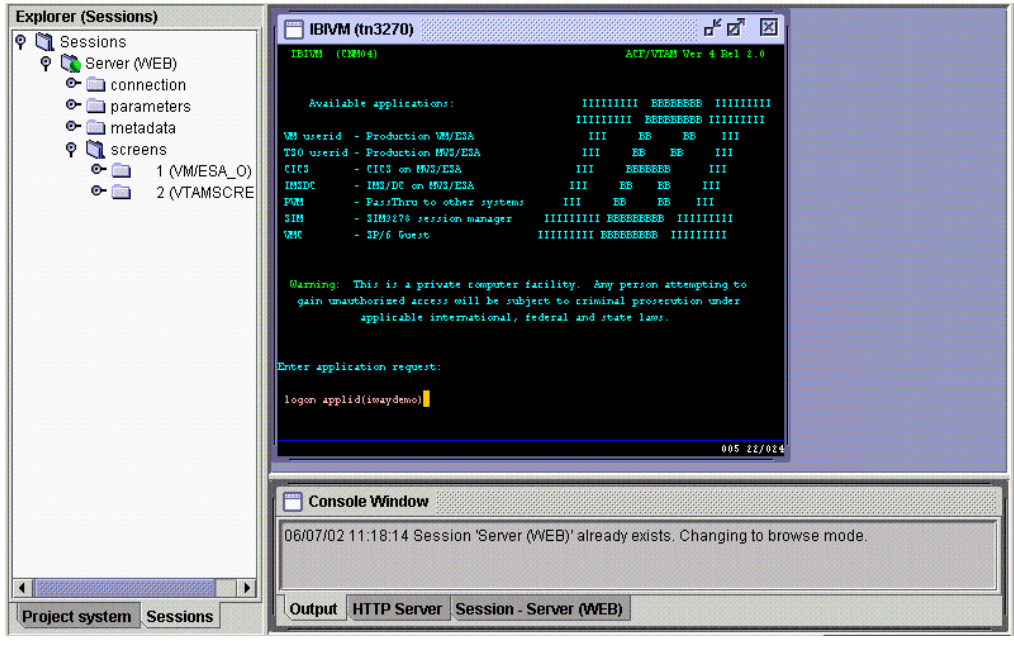
b. For match operations, select the appropriate option.

When a match is made, the iWay Adapter for Telnet proceeds with the keystroke that is pressed, for example, the Enter key or the PFKEY. As a result, the screen is not revealed to the user during run time.

2. Click *OK*.

The screen is now recognized by the iWay Adapter for Telnet.

3. Continue as you did during default emulation (*Creating a Web Application Using Default Translation* on page 3-3) and enter the *LOGON APPLID(IWAYDEMO)* command.



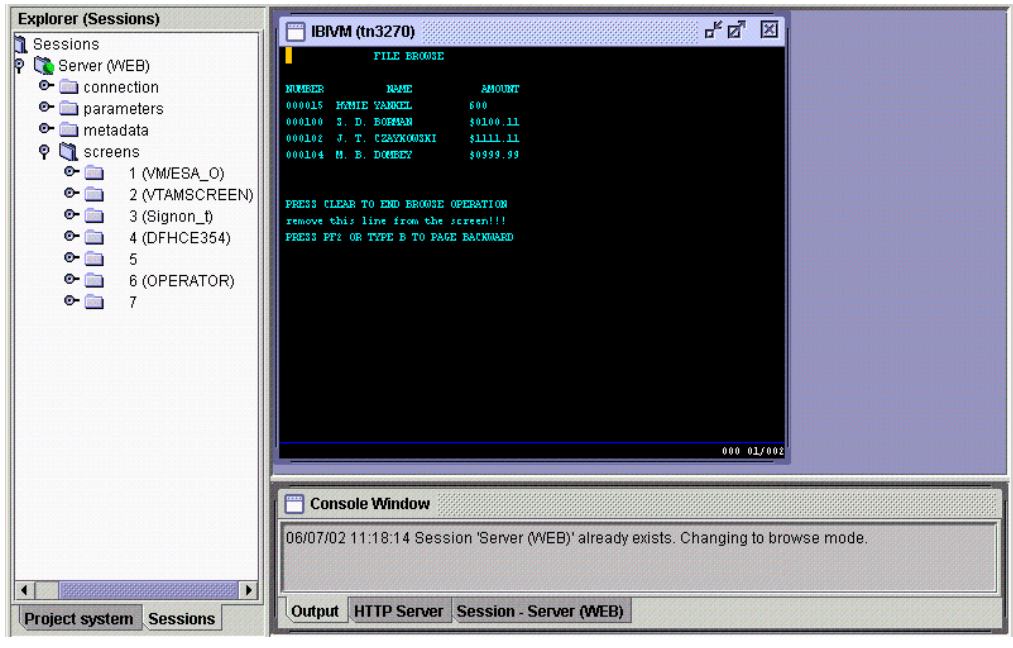
a. Add screen identifiers to the screens that you want to hide.

b. Continue the Add Screen Identifier option for other screens.

For a blank screen, select *Add Blank Screen Identifier.*

Important: Ensure you do *not* identify the Browse screen as you want it to appear in the browser in default translated HTML.

The following pane shows that screens that are recognized are added to the database.



a. To log off CICS, enter the command, *CESF LOGOFF*.

b. Click the *X* in the upper-right corner of the Telnet Designer emulator screen.

4. Right-click the *Server (WEB)* folder under Sessions and select *Save As*.

The Save As dialog box opens.

a. Type a name, for example, WEB2.

b. To save the Telnet application called WEB2, click *Save*.

5. To view the Web application called WEB2, click the *Project system* tab in the lower left.

You can now access the application, WEB2,on the Web.

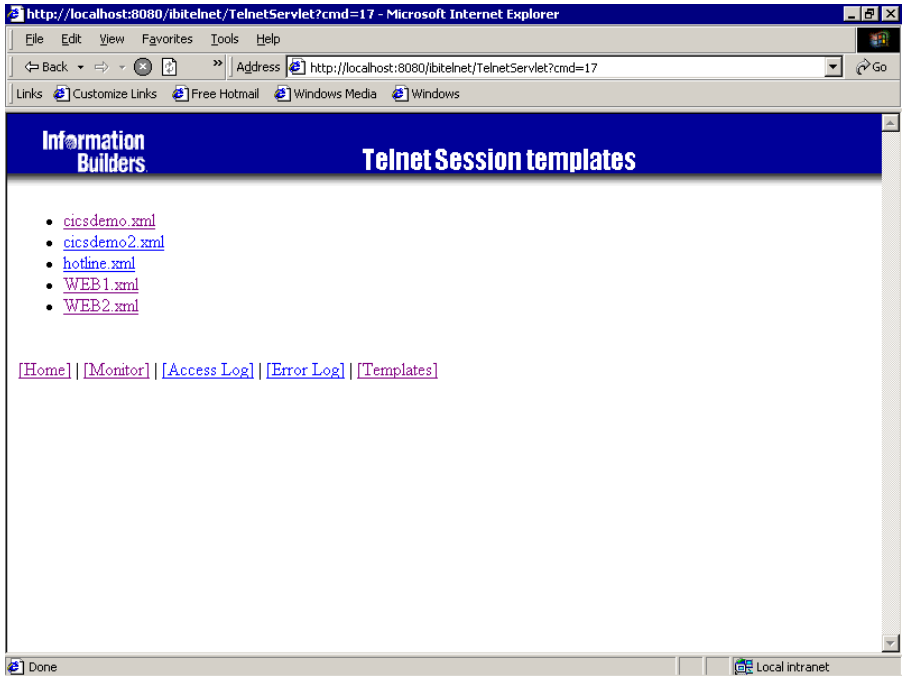
For more information on how to deploy the WEB2 Telnet application, see *How to Test the iWay Adapter for Telnet Web Application on page 3-9*.

For more information on how to start the supplied Web application server to view the WEB2 Telnet application, see *How to Start the Supplied Web Server to Run the Web Application on page 3-10*.

6. In a Web browser that has access to the machine where the iWay Adapter for Telnet is running, enter the following URL:

`http://domain:port/ibitelnet/index.html`

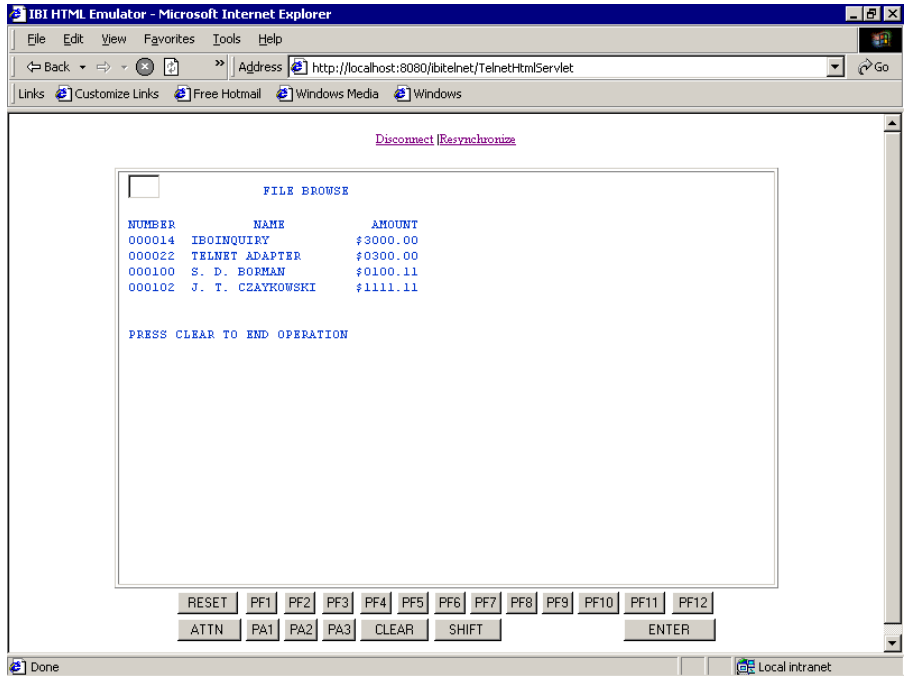
The default HTML window for the iWay Adapter for Telnet opens.



The window contains sample Web applications that connect to CICS and VM systems.

7. Click *WEB2.xml*.

The following window opens.



The VTAM screen, the CICS logon screen, and other screens were eliminated from view.

The screen that appears is default HTML translation. To change its look and feel, you can create:

- A generic template that surrounds the mainframe screen and the other default translated screens. This method requires a manual edit to a file. For more information, see *Using Default HTML Translation With a Generic Template*.
- JSP for the page to totally change its appearance and add functionality. You accomplish this in Telnet Designer. For more information, see *Using Telnet Designer to Create JavaServer Pages* on page 3-26.

Using Default HTML Translation With a Generic Template

With default HTML translation, you can create a generic template that surrounds the translated screen to provide a more customized look and feel. You can make only cosmetic changes using HTML. You cannot completely redesign the screen or combine multiple mainframe screens into one HTML screen.

With a generic template using default translation, you can:

- Change the background color.
- Move the Resynchronize and Disconnect links to the top or bottom of the Web page.
- Change fonts, intensities, and other characteristics of fields.
- Move the translated keyboard to the top or bottom of the Web page.
- Add images.
- Make cosmetic HTML modifications.

Using a Generic Template

In the following procedure, you create a template for the WEB2 application. The Browse screen appears surrounded by an HTML template. You change the background color and also add a new title to the screen. The background color and new title apply to all of the screens that appear, not only for the Browse screen.

Procedure How to Use a Generic Template

To use a generic template:

1. Create a JSP file in the directory, webapps\ibitelnet.

   You can copy the JSP file distributed with the adapter (ibitc_default.jsp) located in the iway\telnet\projects\tomcat\webapps\ibitelnet directory as a sample.

2. Save the *JSP* file as

   ```
   web2template.jsp
   ```

3. Edit the *web2template.jsp* file manually to change the background color and to add a title.

   ```
   <!-- The fieldTab javascript takes care of cursor positioning -->
   <!-- 1) Here you can change the background color -->
   <BODY onLoad='fieldTab()' bgcolor='#FFFF00'>
   ```

   where:

   #FFFF00

   Is the hexadecimal code for the color yellow.

   You can make other HTML modifications to the JSP file, which are reflected in the translated mainframe screen, in this case, the CICS Browse screen.

The following is a new title that is seen for all translated screens:

```
<!-- ------------------------ -->
<!-- THIS MUST ALWAYS BE HERE -->
<!-- ------------------------ -->
```

Snippets from the file reveal notes that can help you make other changes, such as:

```
<!-- 2) Here you can change the translation commands properties -->
<!--    or move it to the bottom of your page                   -->
<font style='font-size:8pt' color='#0033CC'>
<!-- 3) Here you can change the font properties -->
<% telnetConnection.setIntensifiedColor("#006600"); %>
<font style='font-size:8pt' color='#0033CC'>
<!-- 4) Here you can move the keyboard to the top of the page -->
<%= TelnetHtmlHelper.buildKeyboard(emulation) %>
```

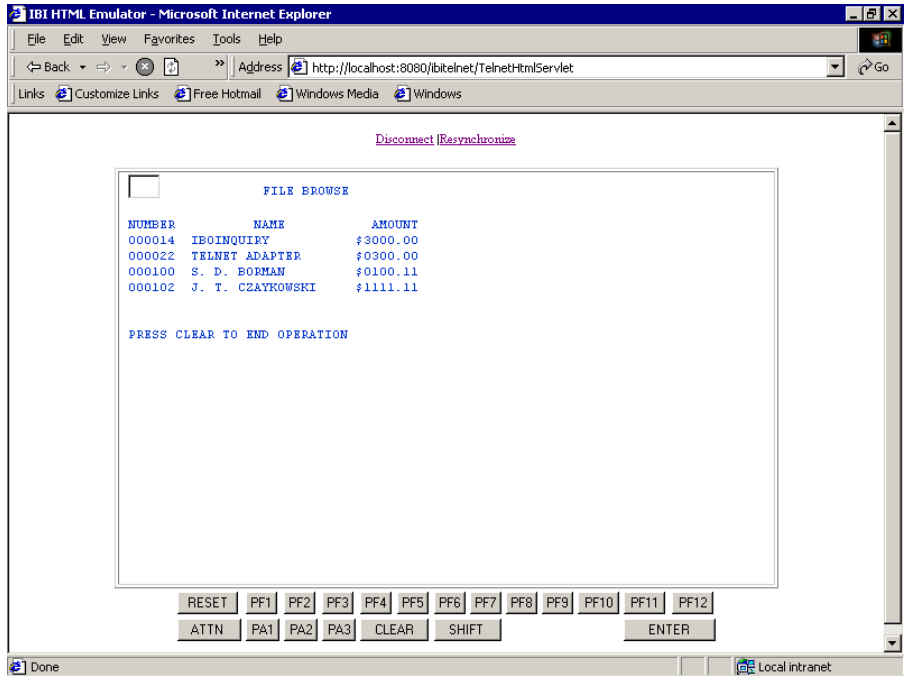
4. Edit the *WEB2.XML* rules file located in the ibitelnet/sessions directory within the webapps directory of the Web server.

   The WEB2.XML file was created when you deployed the application using Telnet Designer.

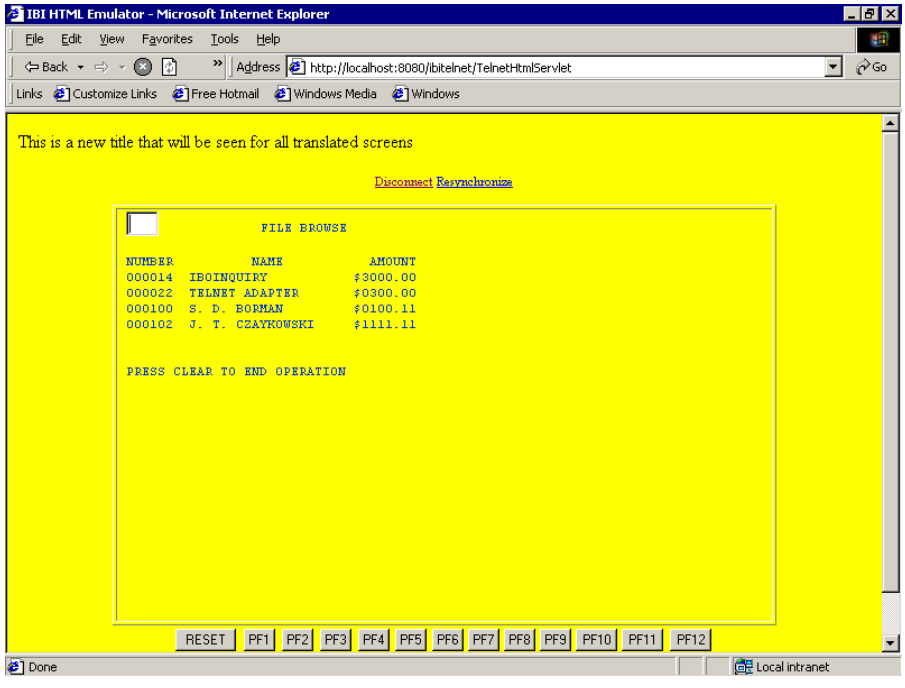
5. Make the following change that identifies the web2template.jsp:

```
<sequence NAME='WEB2' MODE='WEB' VERSION='2.0' >
  <remarks>
  5.Hide the set of screens until the BROWSE Transaction.
  </remarks>
 <connection HOST='IBIVM' PORT='23' EMULATION='tn3270' EXTENDED='OFF'
LANGUAGE='Cp037' TIMERTIMEOUT='15' URL='web2template.jsp' />
```

The following screen illustrates the translated CICS Browse screen *before* adding a template.

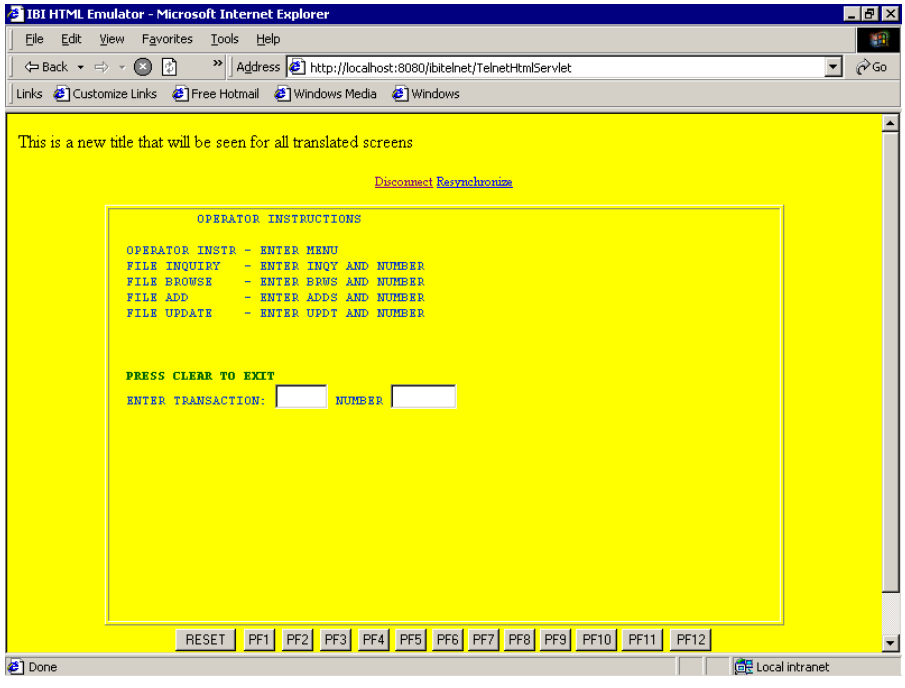


The following screen illustrates the CICS Browse screen *after* adding the template, web2template.jsp, to the WEB2.XML rules file.



Notice the new title and the yellow background color.

When you click the CLEAR key, the menu screen appears with a yellow background and the new title.



Using Telnet Designer to Create JavaServer Pages

Telnet Designer, in Web mode, enables you to make more significant changes to the appearance of the mainframe application than you can make using templates. You also can enhance the functionality of your mainframe online application. You can move fields, extract screen data, and combine one or more screens into one HTML screen.

JavaServer Pages (JSP) enable you to convert mainframe legacy screens using Web elements. The result is an XML rules file as discussed previously as well a JSP file for each screen to which you want to apply a custom HTML page. The XML file contains all the identification rules and the name of the JSP file associated with the Telnet screen.

Creating JavaServer Pages

The iWay Adapter for Telnet and its capability of employing user-defined JavaServer Pages (JSP) is an extension of the HTML translation capability previously described. With default HTML translation, you cannot control the appearance of the HTML page. However, with JSP, you can customize an HTML page so a screen on your mainframe system appears as a Web application. For example, you can add images, combo boxes, and check boxes to customize your application.

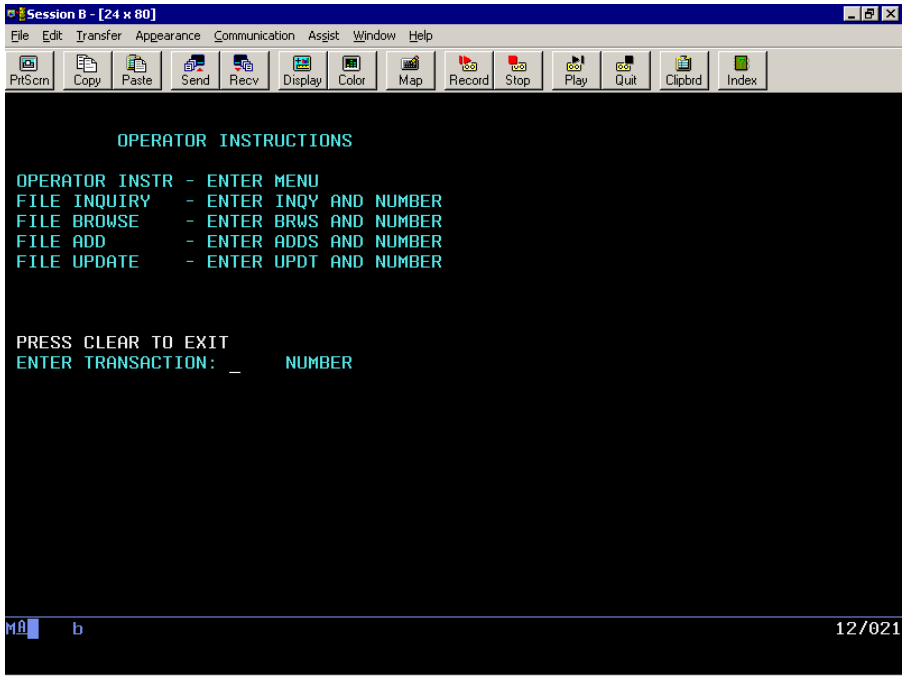
The iWay Adapter for Telnet enables you to control the mainframe Telnet session by associating one JavaServer Page to many mainframe screens.

Note: You can also integrate other applications and data sources with your existing mainframe system to add limitless functionality.

The following procedure describes the steps required to create a Telnet application called WEB3. The procedure describes a system identical to the CICS system that was described for creating the WEB1 and WEB2 applications. In this case, a new application called WEB3 displays the CICS MENU screen in a new way.

When the application finishes using Telnet Designer, you can click WEB3 to view the CICS menu screen JavaServer Pages. All previous mainframe screens are hidden.

The following is the original CICS menu screen to be converted by a JSP called WEB3.JSP.



Procedure How to Run Telnet Designer in Web Mode: JavaServer Pages

To run Telnet Designer in Web mode:

1. To open Telnet Designer, click the Start menu and select *Programs*, *iWay 5.5*, and *iWay Telnet Designer*.

   The Telnet Designer opens.

2. Select *Emulator* from the Tools menu.

   The Terminal Emulator properties dialog box opens.

   a. In the Host field, type a new host name or select an existing host name, for the computer on which the mainframe application resides.

   b. From the Terminal drop-down list, select the type of emulation.

   c. Type the number of the terminal port.

   d. For Mode, select the *Web* option.

3. Click the *Advanced* tab.

   a. To specify a different national language character set, select *Extended Attributes*.

   b. Select the desired language.

4. To connect to the mainframe application and start the emulation session, click *Run*.

   The Telnet Designer displays a VTAM session.

5. Identify all the screens as previously described, remembering that for every blank screen you must select *Add Blank Screen Identifier*.

6. When you come to a screen for which you want to create a JSP (for example, the MENU screen), you must right-click anywhere in the screen and select *Add JSP*.

Adding a JavaServer Page for a Screen

You must add a JSP for each screen that you customize and display during a browser session. The Add JSP option is used to associate a JSP with a mainframe screen or set of mainframe screens.

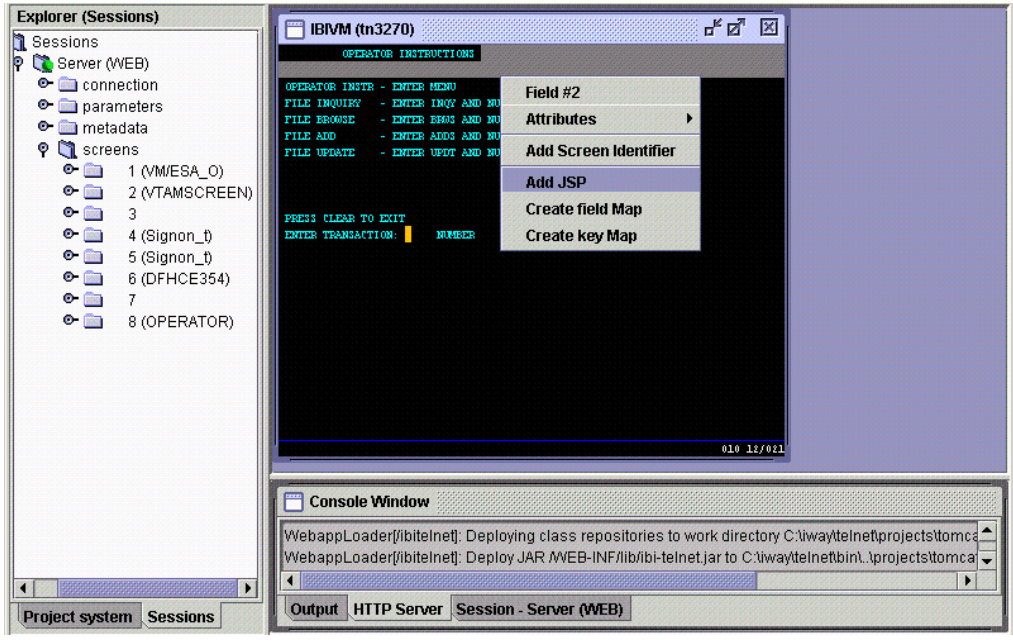
As described previously, all screens for which you add a screen identifier are hidden until the application displays a screen with an associated JSP. After you add the JSP using Telnet Designer, the JSP can be manually edited or used as input to any HTML editor.

Note: You must use caution as the JSP is not a pure HTML file. It contains JSP-specific properties as well as iWay Adapter for Telnet commands that must not be removed or modified.

Procedure How to Add a JavaServer Page for a Screen

In this procedure, you want to add a JavaServer Page (JSP) only for the CICS menu screen.

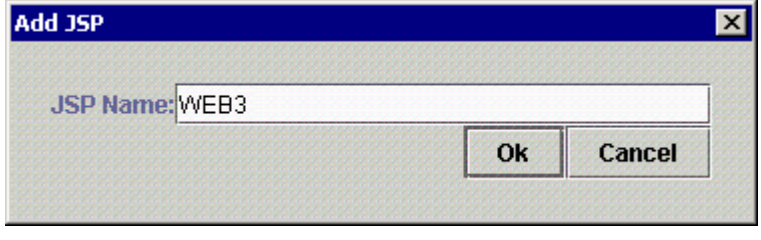
To add a JavaServer Page:



1. Right-click anywhere in the screen and select *Add JSP*.

The Add JSP option is only available if you have identified the screen or if it has been previously identified as indicated in the left pane of Telnet Designer. This indicates that the adapter database was updated.

The Add JSP dialog box appears.



2. Type a name, for example, WEB3 and click *OK*.

This creates the WEB3.JSP file.

When the application (also called WEB3) is deployed, the JSP is placed into a directory called ibitelnet/WEB3 within the webapps directory of the Web server. All JSP that you create for the WEB3 application are placed in the WEB3 folder when you deploy.

To ease development, other options are available to identify the fields on a screen, including the option to rename the PF/PA and AID keys associated with the mainframe screen.

Creating a Field Map and a Key Map

Telnet Designer enumerates the fields on a screen. To work with a particular field in a mainframe screen, you must know the field number. You can view this information in the pop-up menu in any development mode. To facilitate the development of custom HTML pages (as JavaServer Pages), this option enables you to associate a name to the field number.

The generated JavaServer Page has the default keyboard. After you add a JSP, the Create Key Map option also is available. You can redefine any key on the keyboard. The default keyboard is suppressed after you create the first key map.

Procedure How to Create a Field Map

To create a field map:

1. Right-click anywhere on the screen and select *Create field Map*.

   The Field Map dialog box appears.

2. Type the field number and click *Ok*.

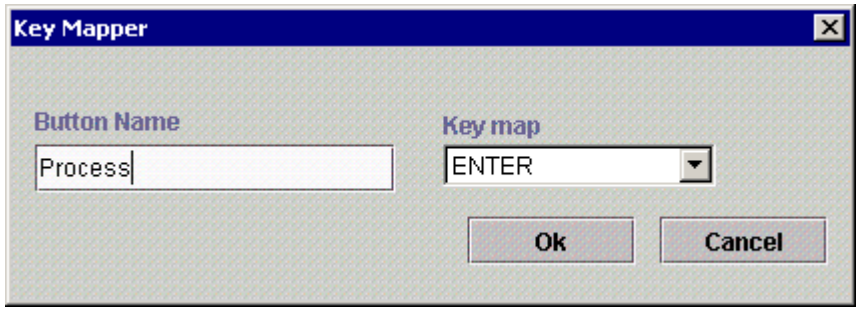
   This information is placed in the generated JavaServer Pages for further reference.

Procedure How to Create a Key Map

To create a key map:

1. Right-click anywhere on the screen and select *Create key Map*.

The Key Mapper dialog box appears.

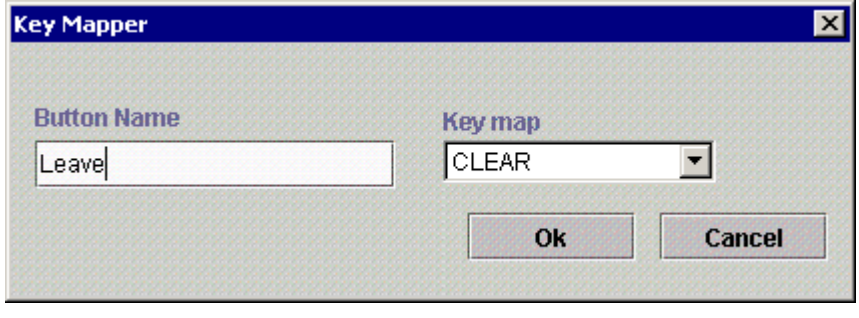


a. In the Button Name field, type a name for the button, for example, Process.

b. From the Key map drop-down list, select a key to associate with the button, for example, Enter.

A button with the title, Process, executes the Enter key when you click *Process*.

Similarly, a button with the title, Leave, executes the CLEAR key when you click the button.



You can continue with the development of the WEB3 application.

2. To log off of CICS after you add the JSP for the Menu Screen, enter the command *CESF LOGOFF*.

3. Click the *X* in the upper right of the Telnet Designer emulator screen.

You return to Telnet Designer.

4. Right-click the *Server (WEB)* folder under Sessions and select *Save As*.

The Save As dialog box appears.

a. Type a name, for example, WEB3.

b. To save the Telnet application, WEB3, click *Save*.

You return to Telnet Designer.

5. Click the *Project system* tab in the lower left.

   The Web application, WEB3, appears in the left pane. You can now access the application, WEB3 from the Web.

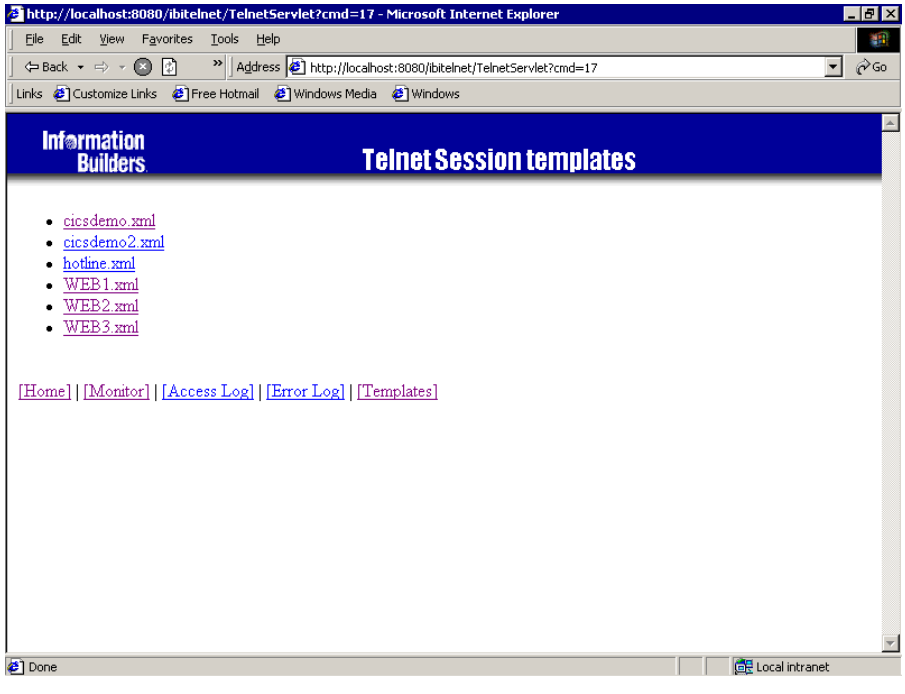
   For more information on how to deploy the WEB3 Telnet application, see *How to Test the iWay Adapter for Telnet Web Application on page 3-9*.

   For more information on how to view the WEB3 Telnet application, see *How to Start the Supplied Web Server to Run the Web Application on page 3-10*.

6. In a Web browser that has access to the iWay Adapter for Telnet, enter the following URL:

   `http://domain:port/ibitelnet/index.html`

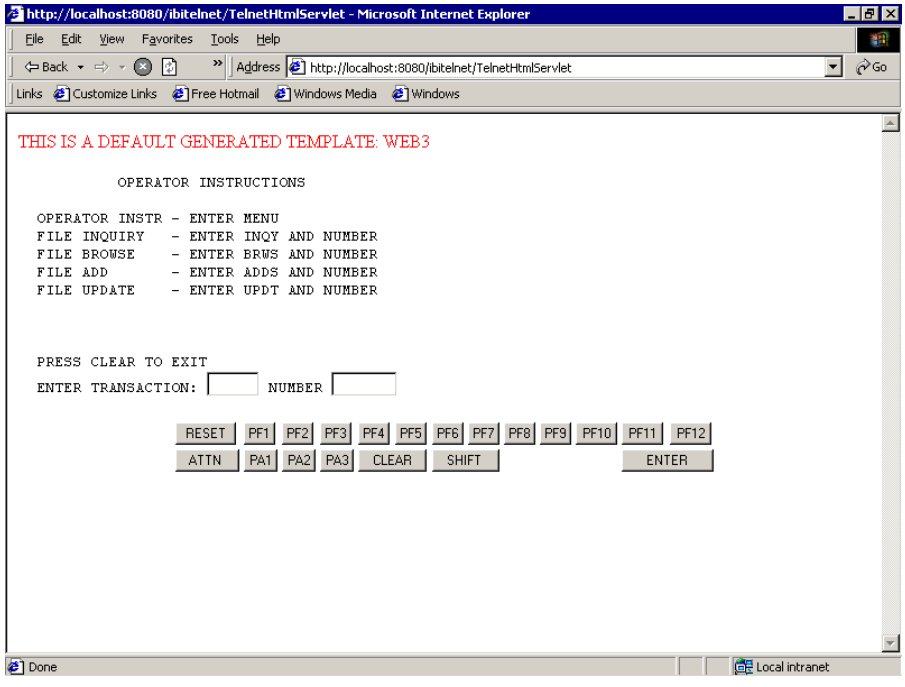
   The iWay Adapter for Telnet default HTML page opens.



   The page contains sample Web applications that connect to CICS and VM systems.

7. Click *WEB3.xml*.

The following screen opens.



The VTAM screen, CICS logon screen, and other screens were eliminated from view.

The following default message appears on the screen:

```
THIS IS A DEFAULT GENERATED TEMPLATE: WEB3
```

This means that the screen uses the JSP called WEB3.JSP.

Now you can edit the WEB3.JSP to completely redesign the look and feel of this CICS menu screen.

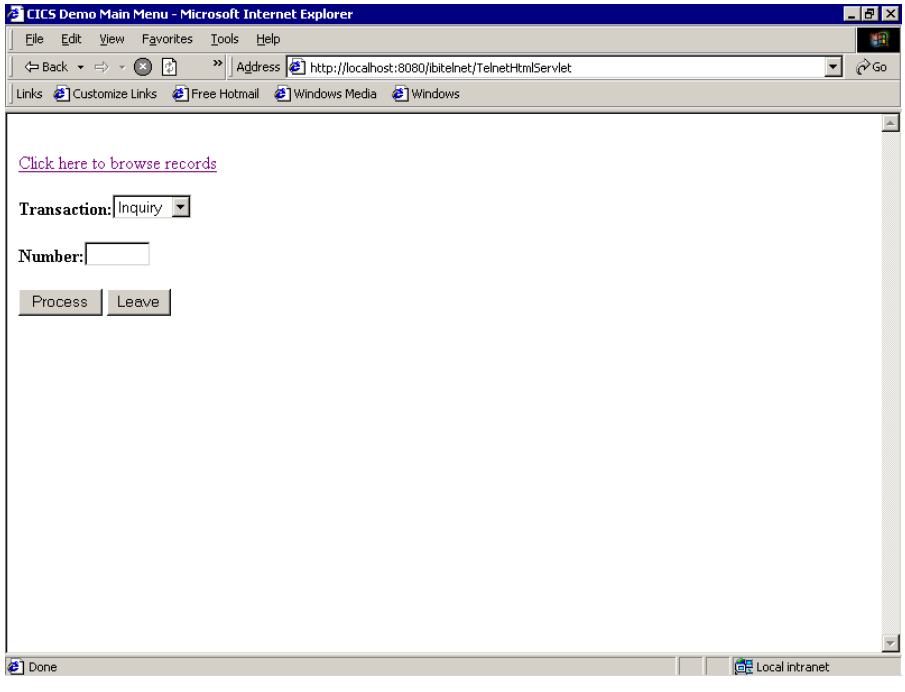
Reference The Default WEB3.JSP File

The following is the default WEB3.JSP file. Upon deployment using Telnet Designer, the file is located in the directory called ibitelnet/WEB3 within the webapps directory of the Web server.

```
<HTML>
<BODY>
<font color="#FF0000">THIS IS A DEFAULT GENERATED TEMPLATE: WEB3</font>
<!-- THIS MUST ALWAYS BE HERE -->
<%@ include file="../templates/beginTemplate.jsp" %>
<!-- Default Translation -->
<pre>
<%= screen.paintHtml() %>
</pre>
<!-- Default keyboard -->

<%= screen.paintKeyboard(emulation) %>
<!-- THIS MUST ALWAYS BE HERE -->
<%@ include file="../templates/endTemplate.jsp" %>
</BODY>
</HTML>
```

You can make modifications to the JSP that can change the look and functionality of the original CICS menu screen as illustrated by the following example:



- The screen uses the JSP called WEB3.JSP.
- It includes a link to execute the BROWSE transaction.
- It includes a drop-down list with options to Inquire, Add, and Update.
- The Process button denotes the Enter key (created a key map in Telnet Designer).
- The Leave button denotes the CLEAR key (created a key map in Telnet Designer).

Example The Enriched WEB3.JSP File

The following sample code is for the enriched WEB3.JSP file that illustrates the previous enhancements. This supplied JSP is called cicsMenu.jsp and is located in the directory called ibitelnet/cicsdemo within the webapps directory of the Web server.

```
<HTML>
<HEAD>
<TITLE>CICS Demo Main Menu</TITLE>
</HEAD>
<BODY>
<!-- ------------------------ -->
<!-- THIS MUST ALWAYS BE HERE -->
<!-- ------------------------ -->
<%@ include file="../templates/beginTemplate.jsp" %>
<!-- BROWSE TRANSACTION
     creating hyperlink that will be equivalent to
     - Entering "brws" in the field 10
     - Issuing the attention key "ENTER"
-->
<br><%= screen.paintHyperlink("Click here to browse records", "ENTER",
10, "brws") %>
<!-- OTHER TRANSACTIONS
     Using the combo box to enter the transaction name to field 10
-->
<p><b>Transaction:</b><select name="10">
                      <option value="inqy">Inquiry
                      <option value="adds">Add
                      <option value="updt">Update
                      </select>

<!-- Painting field 12, which is an unprotected field for entering number
-->
<p><b>Number:</b><%= screen.paintField(12) %>
<!-- SCREEN MESSAGE -->
<%
    String message = screen.getField(8).getValue().trim();
    if (message.startsWith("PRESS CLEAR TO EXIT"))
     message = "";
%>
```

```
<p>
<center>
<font color="#FF0000">
<b><%= message %></b>
</font>
</center>
<p>
<!-- CREATING BUTTONS FOR KEYBOARD STROKES -->
<%= screen.paintButton("Process", "ENTER") %>
<%= screen.paintButton("Leave", "CLEAR") %>
<!-- ------------------------ -->
<!-- THIS MUST ALWAYS BE HERE -->
<!-- ------------------------ -->
<%@ include file="../templates/endTemplate.jsp" %>
</BODY>
</HTML>
```

Deploying Your Application on a Web Application Server

The previous examples illustrated the deployment of your Telnet application on the supplied Web application server. All of the files required to run the application on the Web are automatically deployed on the Web server when you select to deploy in Telnet Designer. In addition, the adapter software is already installed within the supplied Web application server.

To run an iWay Adapter for Telnet Web application on a Web application server other than the supplied one (the case in a production environment), you must perform the following configuration.

- The iWay Adapter for Telnet software is required for deployment.

  Most Web application servers support WAR files. The ibitelnet.war file must be deployed on the particular Web application server.

  In some cases, the files contained within the ibitelnet.war must be appended to the class path.

- ibitelnet directory is required.

  The directory contains the created XML rules file and JavaServer Pages (JSP) that were created as a result of running Telnet Designer in Web deployment mode.

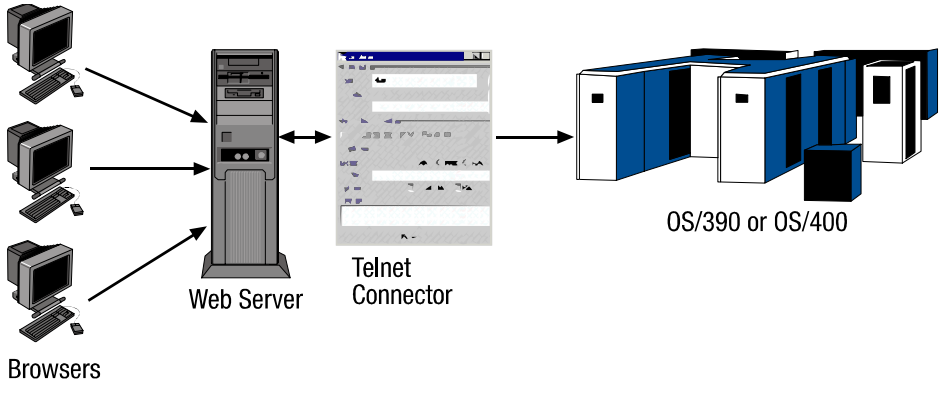
  These folders and files must be made available to your Web application server.

Using HTML Translation Services

In Web mode, the iWay Adapter for Telnet Designer creates an XML rules file that controls your online session with appropriate JavaServer Pages (JSP) for each screen you want to run with a template. These templates change the look and feel of the screen or add functionality to the screen.

It is important to understand the effects of running a mainframe online application over the Web. These effects can occur when using default translation with or without a generic template or when using the template facility of Telnet Designer'.

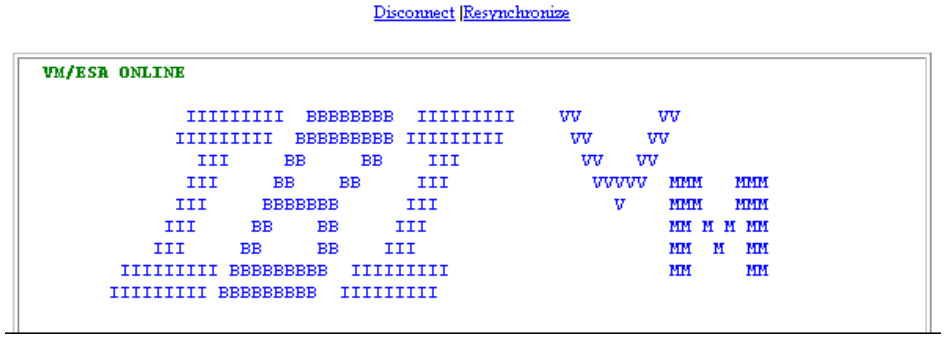
HTTP communication occurs in half-duplex mode, which means the communication occurs in both directions, but with only one entity communicating at a time. To bring Terminal Emulator capabilities to the Web, the HTML translation services determine when the communication from the back end completes.



The following commands are available:

- **Unlock Keyboard Command.** A field in the TN3270 and the TN5250 protocol determines the state of the terminal keyboard. The TelnetHtmlServlet stops reading screens from the back end when the keyboard is unlocked by this command.

- **Disconnect.** The Disconnect command terminates the connection to the back end (OS/390 or OS/400). It is recommended that you log off the back end before using the Disconnect feature. For example, failing to log off the mainframe session could leave your TSO ID or your CICS connection in an active state.



- **Resynchronize.** You can use the Resynchronize option to synchronize the HTML page with the current Telnet screen when an application releases the keyboard before it finishes transmitting.

  It is recommended that you resynchronize each screen for either deployment method, default translation or template modes.

- **Minimal Buffer Size.** Occasionally, an application sends small messages while keeping the keyboard unlocked. You can avoid displaying incomplete screens by using the TelnetServlet property, minBufferSize. It enables you to configure the minimal screen size you want to display. The TelnetHtmlServlet attempts to read more screens if the current screen is smaller than the minBufferSize. The TelnetServlet property, TelnetTimeout, defines the maximum time, in seconds, it waits for an answer.

Reference The HTML Configuration File

Telnet Designer enables you to control parameters that affect the Emulator console, screen navigation, and the Web session. The parameters have an effect on default translation and the template facility. The configuration file called web.xml is located in the telnet/wwwtelnet/web-inf directory.

The following parameters affect the Terminal Emulator console facility.

Parameter	Description
AccessLogOn	Logs an entry for each established session.
User	Authenticates the management commands for HTTP.
Password	Password associated with the user.

The following parameters affect screen navigation.

Parameter	Description
TelnetTimeout (seconds)	Number of seconds to retry a connection if the initial connection attempt fails.
MaxTimeoutLimit	Number of retries when a time-out occurs.
MinBufferSize	Minimum size acceptable for a Telnet screen. A smaller screen requires another read.

The following parameter affects the HTML Web session:

Parameter	Description
HTTP Timeout (hours)	The time, in hours, an inactive session is retained.

The following parameter affects the location of created XML rules files when using Telnet Designer in template mode (template facility).

Parameter	Description
Repository	Location of the files that contain the template rules. Used only in template mode.

TelnetHtmlServlet Parameters

Understanding the HTML service request helps you to create screen templates. The TelnetHtmlServlet controls Telnet sessions. Each HTML screen sends parameters for the TelnetHtmlServlet to handle the Telnet screen. The types of parameters are:

- **Field Handling**

  In HTML, unprotected fields are represented as text fields, and protected fields are regular text. The unprotected fields are sent back to the TelnetHtmlServlet for handling. In Default Translation, unprotected fields look similar to the following:

  ```
  <input onFocus='setCursor(this.name)' type='text' value='' size='8'
  maxlength='8' name='12'>
  ```

  Note: The names of the field must match the numbers represented in the Telnet screen.

- **Keyboard Handling**

  The keyboard is created using HTML buttons, as in the following example:

  ```
  <input type='button' value='PA1' style='font-size: 8pt'
  onclick='setAttention("PA1")'>
  ```

  The onClick event uses a Java Script function to set the current keyboard value in the keyboard hidden field. The Shift key is used to switch from PF1-PF12 to PF13-PF24.

  These two fields track which attention key is pressed:

  ```
  <input type='hidden' name='shift' value='0' size='2'>
  <input type='hidden' name='keyboard' value='ENTER' size='15'>
  ```

- **Cursor Positioning**

  Cursor positioning reflects the field position of the cursor for the current Telnet screen. Each field is called to a Java Script function, setCursor(), on the onClick event:

  ```
  <input type='hidden' name='cursor' size='3'>
  ```

Screen Objects

Screen templates enable you to redesign Telnet screens. The template screen is a JavaServer Page (JSP) on which Telnet HTML Services expose a screen object. The screen object obtains the dynamic value of the Telnet Session.

Example Using a Hyperlink Equivalent to a Keystroke

You can use the hyperlink equivalent to a keystroke as follows:

```
<!-- Issuing the attention key "CLEAR" -->
<a href='TelnetHtmlServlet?keyboard=CLEAR&10=brws'>Click here to browse
records</a>
```

Example Using a Hyperlink Equivalent to a Field and a Keystroke

You can use the hyperlink equivalent to a field and a keystroke as follows:

```
<!-- Entering "brws" in the field 10  and Issuing the attention key
"ENTER" -->

<br><a href='TelnetHtmlServlet?keyboard=ENTER&10=brws'>Click here to
broBSE records</a>
```

Example Using a Hyperlink Equivalent to a Cursor and a Keystroke

You can use a hyperlink equivalent to a cursor and a keystroke as follows:

```
<!-- Positioning the cursor at field 16 and Issuing the attention key
"F4"
-->
<a href='TelnetHtmlServlet?keyboard=F4&cursor=16'>Names </a>
```

Reference Form Fields and Form Buttons

Form fields include combo boxes for specific fields and protected and unprotected screen fields. Form buttons are used to create keyboard strokes.

Example Using a Combo Box for a Specific Field

You can use the combo box for a specific field as follows:

```
<!-- Using the combo box to enter the transaction name to field 10 -->
<b>Transaction:</b><select name="10">
                     <option value="inqy">Inquiry
                     <option value="adds">Add
                     <option value="updt">Update
                   </select>
```

Example Using an Unprotected Screen Field

You can use unprotected screen fields as follows:

```
<!-- Painting field 12, which is an unprotected field for entering number
-->
<b>Number:</b><input type='text' value='' size='6' maxlength='6'
name='12'>
```

Example Using a Protected Screen Field

You can use protected screen fields as follows:

```
<!-- Painting field 3, which is protected field  -->
<b>Number..:</b><%= screen.paintField(3) %>
```

Example Using a Form Button

You can use form buttons as follows:

```
<!-- CREATING BUTTONS FOR KEYBOARD STROKES -->
<input type='button' value='Process' onclick='setAttention("ENTER")'>
<input type='button' value='Leave' onclick='setAttention("CLEAR")'>
```

Creating an HTML Page to Start a Telnet Session

You can create your own HTML page to start a mainframe session over the Web. To establish a Telnet connection, you must use the Connect command, which requires the following parameters.

Parameter	Description
host	The host to which you want to connect. **Note:** Must be in lower case.
port	The port number on which the Telnet target listens. The default port is 23.
emulation	The terminal (term) type: Tn3270 or Tn5250. The default is Tn3270.

Example Establishing a Telnet Connection With a Hyperlink

You can establish a Telnet connection by adding the following code to an HTML page:

```
<a href=TelnetServlet?cmd=connect&host=ibivm&emulation=tn3270>IBI VM</a>
```

Procedure How to Establish a Telnet Connection With an HTML Form

To establish a Telnet connection with an HTML form:

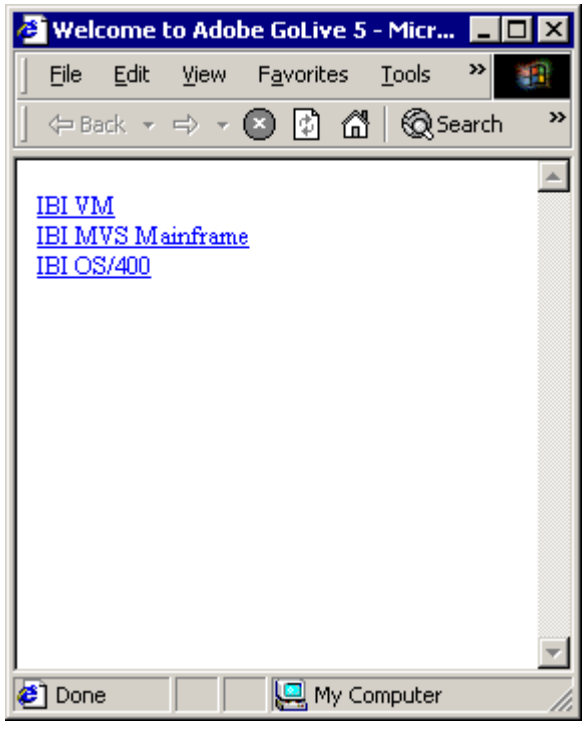
1. Enter the following in a text editor and save as *sample1.html* in your Telnet\wwwtelnet directory:

```
<html>
<body>
<a href=TelnetServlet?cmd=connect&host=vmhost&emulation=tn3270>MY VM
Machine</a>
<br>
<a href=TelnetServlet?cmd=connect&host=mvshost&emulation=tn3270>
MY MVS Machine
</a>
<br>
<a href=TelnetServlet?cmd=connect&host=as400host&emulation=tn5250>
MY As/400 Machine
</a>
</body>
</html>
```

2. Start your Web server.
3. Run your *sample1.html* file from the browser.

```
http://domain:port/telnet/TelnetHtmlServlet
```

The following window opens.



CHAPTER 4

Developing Applications Using Emulation and Recorder Modes

Topics:

- Emulation Mode
- Recorder Mode

The iWay Emulation Adapter (3270/5250) for BEA WebLogic provides the following two modes to help you develop Telnet applications:

- **Emulation mode**, in which you can access a mainframe session directly from your workstation using Telnet Designer. Designer displays standard 3270 and 5250 screens.
- **Recorder mode**, in which you can record interaction with a mainframe and later play back the recording to simulate a live mainframe session. Simulating a session enables you to develop an application offline. You can simulate a Web mode connection or an RPC mode connection.

Emulation Mode

The iWay Adapter for Telnet Emulation mode enables you to access a mainframe session directly from Telnet Designer. This simple emulation displays standard 3270 or 5250 screens and does not create any work files on your system.

Using Emulation Mode

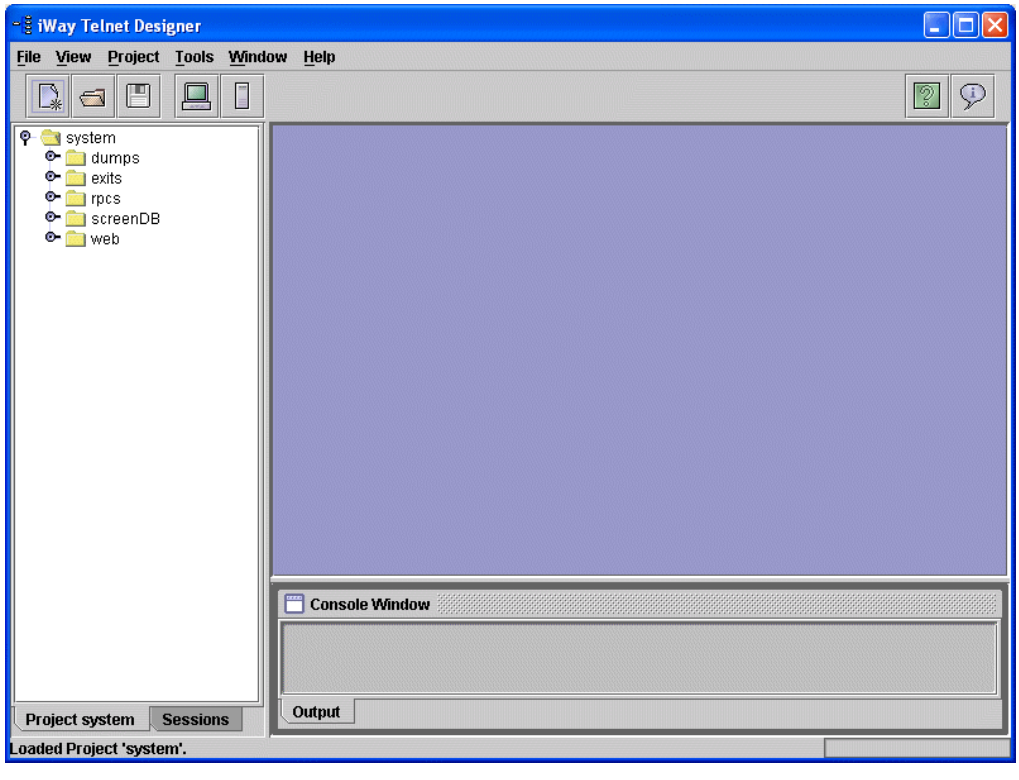
Emulation mode enables you to access the mainframe as you develop Telnet applications.

Procedure How to Run Telnet Designer in Emulation Mode

To run Telnet Designer in Emulation mode:

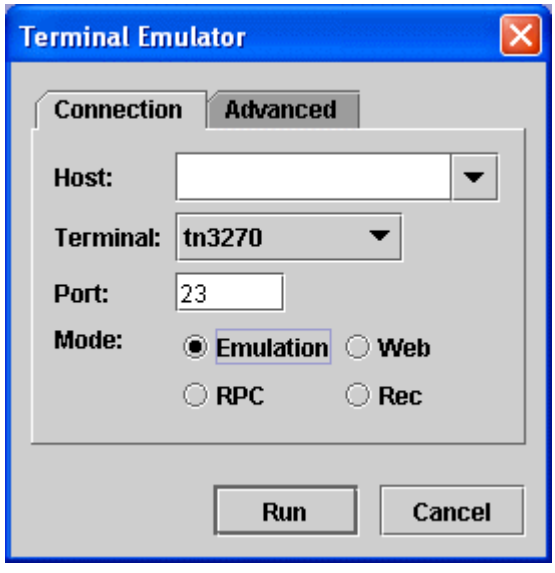
1. Click the Start menu and select *Programs*, *iWay 5.5*, and *iWay Telnet Designer*.

   Telnet Designer opens.



2. Select *Emulator* from the Tools menu.

The Terminal Emulator properties dialog box opens.



a. In the Host field, type a new host name or select an existing host name for the computer on which the mainframe application resides.

b. From the Terminal drop-down list, select the type of emulation.

c. Type the number of the terminal port.

d. For Mode, select the *Emulation* option.

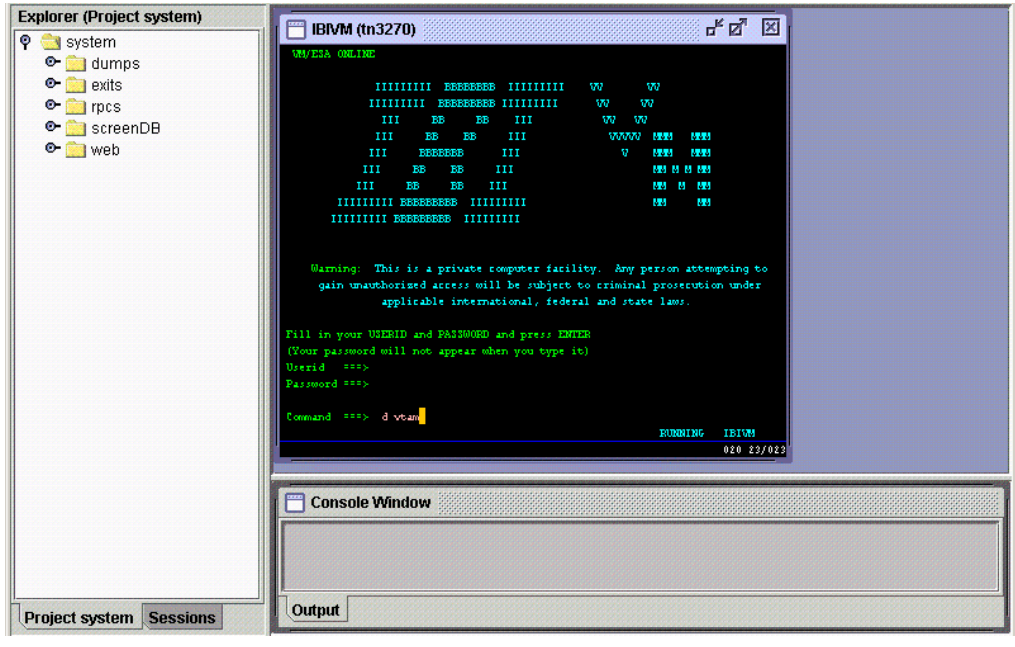
3. Click the *Advanced* tab.

a. To specify a different national language character set, select *Extended Attributes*.

b. Select the desired language.

4. To connect to the mainframe application and start the emulation session, click *Run*.

The following window opens, displaying a VTAM session.



You can move through each mainframe screen.

a. To navigate through the mainframe online system of your site, enter the appropriate information.

b. To terminate emulation at any time, click the *X* in the upper right of the emulation window.

Recorder Mode

The iWay Adapter for Telnet Recorder mode enables you to capture all of your mainframe application screens and end-user keystrokes using the adapter's Designer terminal emulation. You can store the screens and keystrokes in a file. The 3270 or 5250 communication information is transferred into a binary file that you can refer to when you start a Player Server.

Creating a Prerecorded File

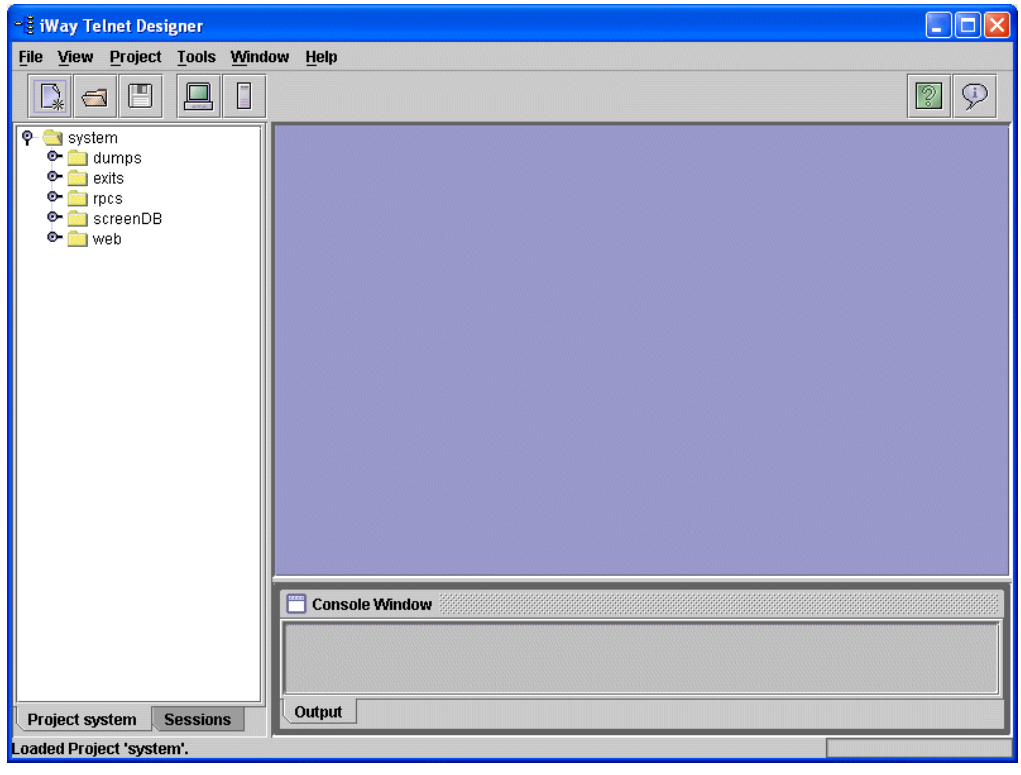
The Player Server uses the prerecorded file as a live connection to the mainframe. With the prerecorded file, you can continue your development off-line without a connection to the mainframe host. This is a useful feature for off-line development, testing, benchmarking, prototyping, debugging, and demonstration purposes.

Procedure How to Run the Telnet Designer in Recorder Mode

To run Telnet Designer in Recorder (REC) mode:

1. Click the Start menu and select *Programs*, *iWay 5.5*, and *iWay Telnet Designer*.

The Telnet Designer opens.



2. Select *Emulator* from the Tools menu.

   The Terminal Emulator properties dialog box opens.

   a. In the Host field, type a new host name or select an existing host name for the computer on which the mainframe application resides.

   b. From the Terminal drop-down list, select the type of emulation.

   c. Type the number of the terminal port.

   d. For Mode, select the *Rec* option.

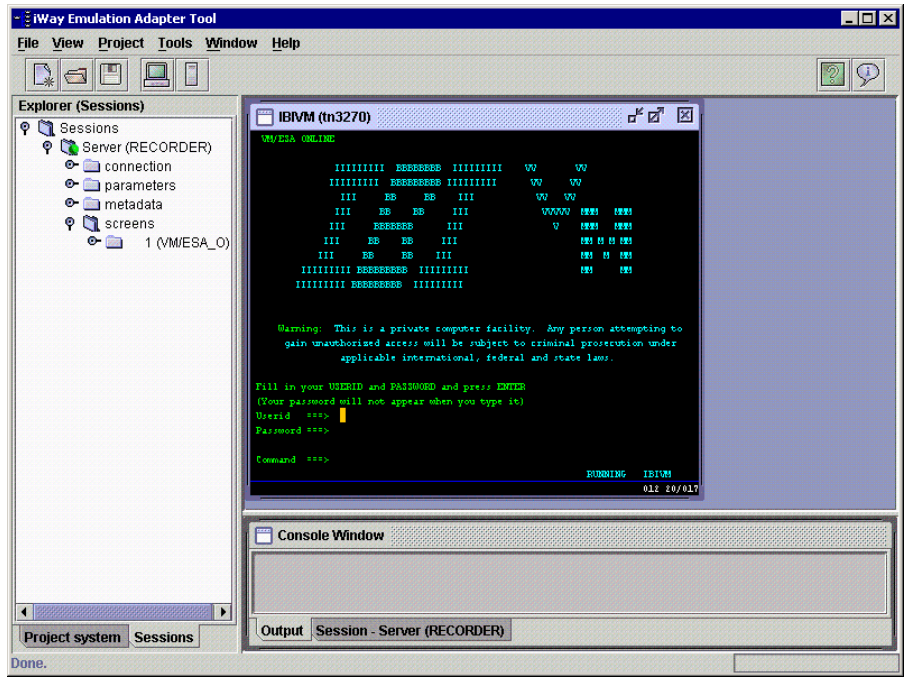
3. Click the *Advanced* tab.

   a. To specify a different national language character set, select *Extended Attributes*.

   b. Select the desired language.

4. To connect to the mainframe application and start the emulation session, click *Run*.

The following screen opens, displaying a VTAM session.



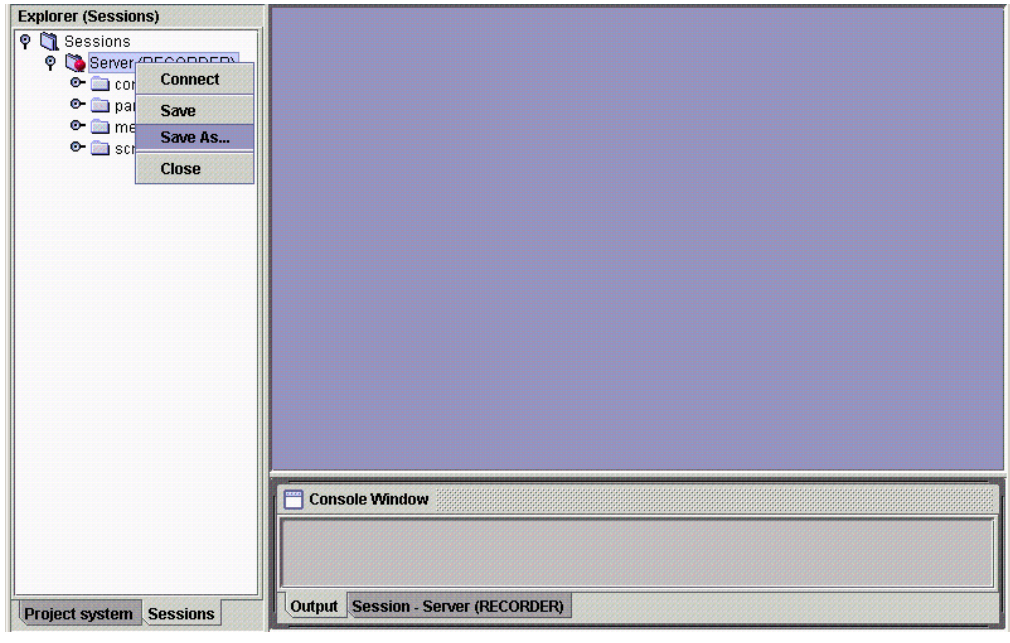
You can move through each mainframe screen.

a. To record each screen, enter the appropriate information to navigate through the mainframe online system of your site.

b. When you are finished, log off from the mainframe session, for example, for CICS, issue the command CESF LOGOFF.

5. To close the window, click the *X* in the upper right of the mainframe session window.

You can now save the prerecorded file.



6. Right-click the *Server (RECORDER)* icon and select *Save As*.

   The Save As dialog box opens.

7. Type a name, for example, REC1, and click *Save*.

   You return to Telnet Designer.

8. Click the *Project* system tab on the lower left corner.

   In the left pane, the REC1 file appears under the dumps folder. You are ready to develop against the prerecorded file called REC1.

Developing Against a Prerecorded File

Telnet Designer enables a developer to create the Telnet components offline without a connection to the mainframe by using the Recorder (REC) mode.

The Player Server executes against a prerecorded file. You can create several files where each one can simulate a specific online application, such as a browse transaction or an update transaction. You also can create one large file that encompasses many mainframe transactions. For example, two or more developers can work on separate transactions on a laptop that is disconnected from the actual mainframe.

You have all options available to create an iWay Adapter for Telnet application against a prerecorded file. As explained in the previous chapters, you can run Telnet Designer to create a Web-based solution or an RPC solution. When running against a prerecorded file, development is not running against the mainframe; there is no mainframe connection.

The following procedures describe how to:

- Start a player session against a prerecorded file.

  You can start a player session from the dumps node, by selecting from the Tools menu, or by clicking the Player Server icon.

- View or change the Player Server connection information.

  Telnet Designer enables you to control the port on which the player server is running. The default is 1010. The port number can change to any desired port number. The port number is used to connect to the Telnet session to create an adapter application

- Run Telnet Designer against a prerecorded file.

  While the Player session is running, you can run Telnet Designer against the session to create a Web-based or RPC application. You can use Telnet Designer as if you were connected to the mainframe.

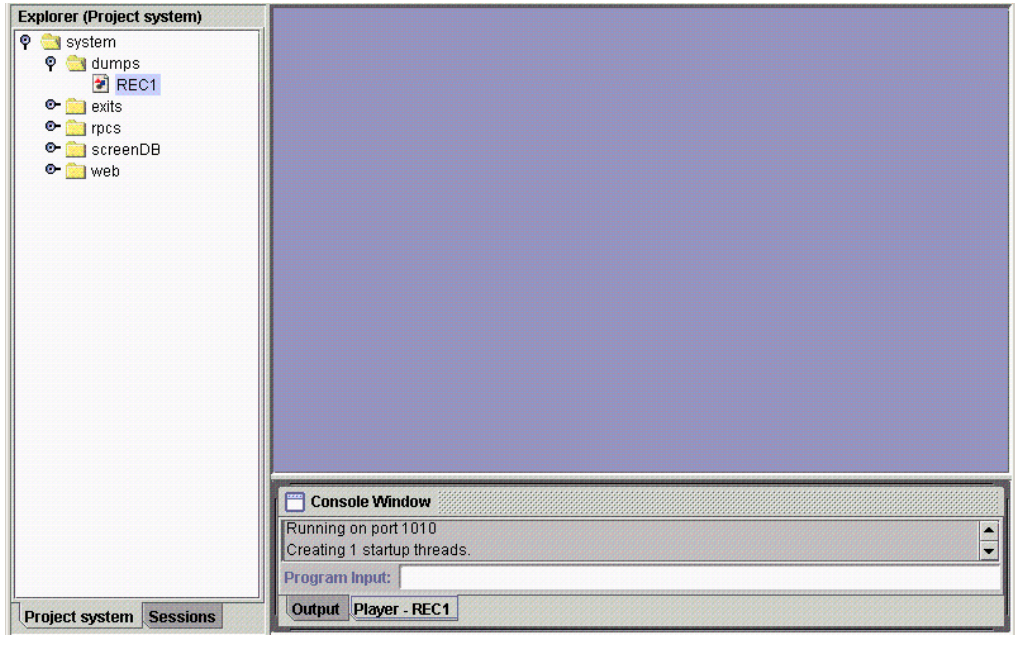
Procedure How to Start a Player Session Against a Prerecorded File

You can start a player session in one of the following ways:

- To start a player session by expanding the dumps node:

1. In the left pane of Telnet Designer, expand the dumps node to view all the prerecorded files.
2. Right-click *REC1*and click *Start*.

The Player - REC1 tab becomes available in the lower right pane.



3. Click the *Player - REC1* tab.

A console window opens and displays the port number on which the player is running. Port 1010 is the default port.

- To start a player session from the Tools menu or from an icon on the menu bar:

1. Click *Tools* and select *Player Server* or click the *Player Server* icon on the menu bar.

The Player Server dialog box, listing the available prerecorded files, opens.

2. Select *REC1* and click *Run* to start a player session.

   Before you connect, using the Emulator function, you must know on which port the player server is running. Port 1010 is the default port.

3. Specify the value for the port when you use Telnet Designer to create a Telnet application.

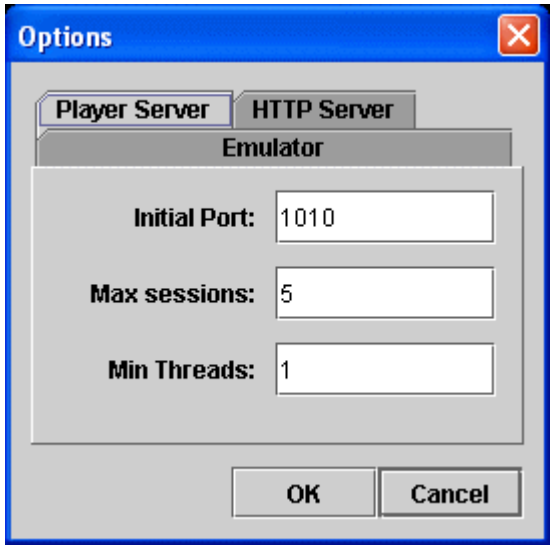
Procedure **How to View or Change the Player Server Connection Information**

To view or change the player server connection information:

1. In Telnet Designer, select *Options* from the Tools menu.

   The Options dialog box opens.

2. To view the options on the *Player Server* tab, click it.



The player session is running on port 1010 on the local machine. This port number is required for a developer on a machine other than the one where the Player Server is running. A developer with a network connection to the Player machine (in this case, localhost), can develop a Telnet application using Telnet Designer against the local prerecorded file, REC1.

Procedure How to Run the Telnet Designer Against a Prerecorded File

To develop against the prerecorded file:

1. Develop a Web-based adapter application.

   For more information, see Chapter 3, *Using Web Mode*.

2. From the Tools menu of Telnet Designer, select *Emulator*.

   The Terminal Emulator properties dialog box opens.

   a. In the Host field, type a new host name or select an existing host name for the computer on which the mainframe application resides.

   b. From the Terminal drop-down list, select the type of emulation.

   c. Type the number of the terminal port.

   d. For Mode, select the *Web* option.

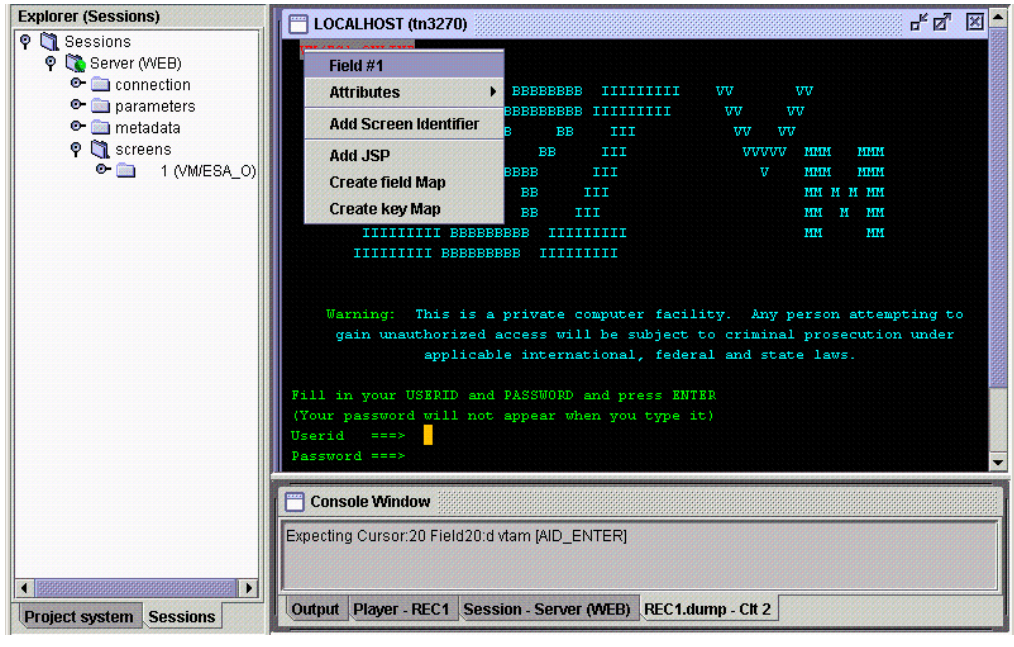
3. Click the *Advanced* tab.

a. To specify a different national language character set, select *Extended Attributes*.

b. Select the desired language.

4. To connect to the mainframe application and start the emulation session, click *Run*.

The following screen opens that matches the first screen in the prerecorded dump file.



You can navigate through the screens as if you were connected to the mainframe.

Note: When you are using Telnet Designer against the prerecorded file, you must enter the same commands and input data that were used when the prerecorded file was created. If you input an incorrect field or keystroke (for example, an unmatched Enter or PF2 key), the Player session ends. You must press Enter to clear the waiting session. To verify what must be entered, you can play the dump file. For more information see *How to Start a Player Session Against a Prerecorded File* on page 4-9.

APPENDIX A

Emulator Keyboard Mapping

Topic:

- 3270/5250 Emulator Keyboard Mapping

This appendix describes how your keyboard maps to 3270 or 5250 keys when you are using the iWay Emulation Adapter (3270/5250) for BEA WebLogic.

Reference **3270/5250 Emulator Keyboard Mapping**

The iWay Emulation Adapter (3270/5250) for BEA WebLogic emulator keyboard mapping is:

3270 / 5250 Key	Keyboard
PF1 - PF12	F1 - F12
PF13 - PF24	Shift F1 - Shift F12
Clear	Ctrl-C
PA0	Alt F1
PA1	Alt F2
PA3	Alt F3
Page Up	Page Up
Page Down	Page Down
Forward Tab	Tab or Ctrl - F
Backward Tab	Shift Tab or Ctrl- B
Rubout	Backspace
Delete	Delete or Ctrl-E
Home	Home
New Line	Ctr -N

Reader Comments

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

Mail: Documentation Services - Customer Support
Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898

Fax: (212) 967-0460

E-mail: books_info@ibi.com

Web form: http://www.informationbuilders.com/bookstore/derf.html

Name:_____

Company:_____

Address:_____

Telephone:_____Date:_____

E-mail:_____

Comments:

Reader Comments

Information Builders, Two Penn Plaza, New York, NY 10121-2898 **(212) 736-4433**