OC://WebConnect

User's Guide and Reference Version 3.2

OPENCONNECT® SYSTEMS

2711 LBJ Freeway, Suite 800
Dallas, TX 75234
(972) 454-5200
Fax: (972) 888-0688

Trademarks

OC://WebConnect–Trademark of OpenConnect Systems Incorporated.

Other trademarks or registered trademarks are the property of their respective owners.

Contents

Chapter 1: Introduction and Overview

Introduction to OC://WebConnect

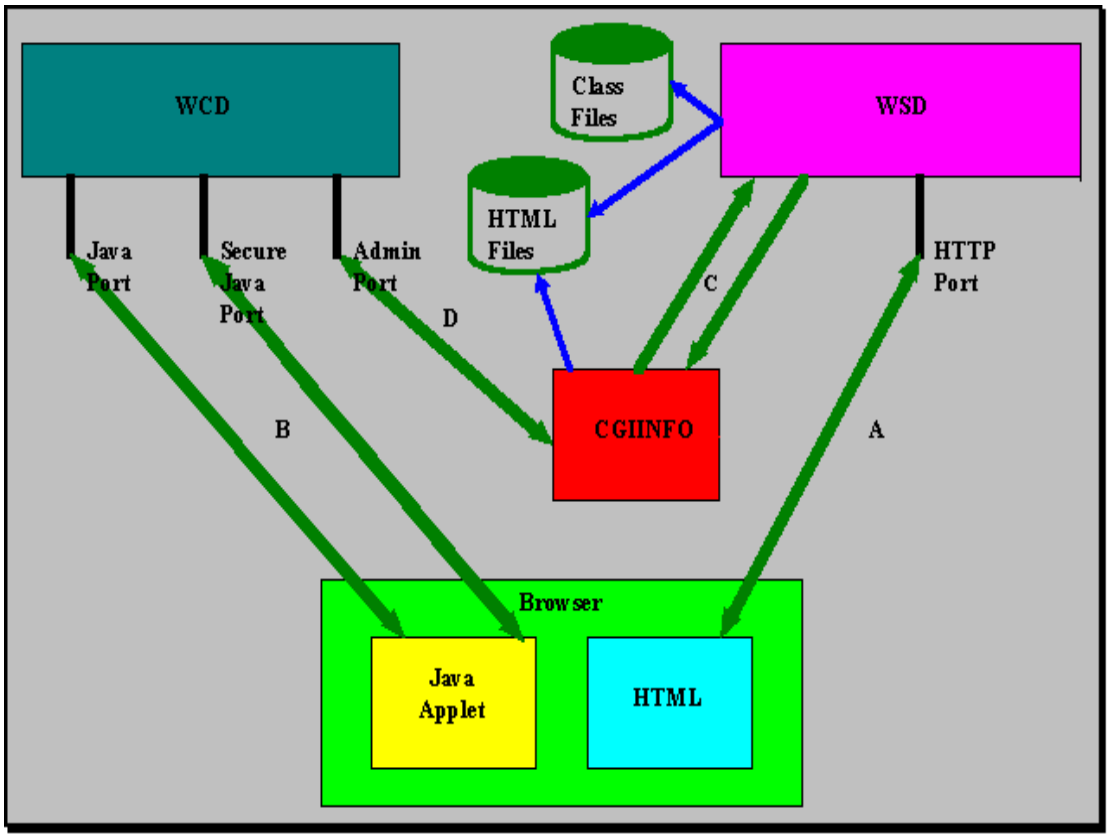
OC://WebConnect provides reliable, secure and scaleable host access using a Java enabled Web browser. The OC://WebConnect product takes advantage of the strengths of traditional client server technology while using the networking access and ease of use of popular browsers and Java technology to deliver the traditional host access emulation features over an intranet or the Internet.

Secure host access over an intranet or Internet is provided by combining a OC://WebConnect Java client on a Java enabled browser and the OC://WebConnect Server on UNIX or NT all secured by Secure Socket Layer(SSL) authentication and encryption or RSA Data SecurityTM encryption.

OC://WebConnect Architectural Overview

OC://WebConnect uses a three tier architecture to provide 3270, 3287, 5250 and VT emulation sessions via Web access. This three tier architecture consists of OC://WebConnect JAVA client applets launched from a Java enabled Web Browser, the OC://WebConnect Web Emulation server, and any TN or Telnet server.(See Figure a below). The OC://WebConnect product supplies the Web Emulation server and JAVA client applets for 3270, 3287,5250, or VT. The Java enabled Web Browser and TN or Telnet server must be purchased separately. The TN server can be purchased from OCS.

Figure 1: OC://WebConnect 3 Tier Model - TN/Telnet Server, Web Emulation Server, Java Applet Client

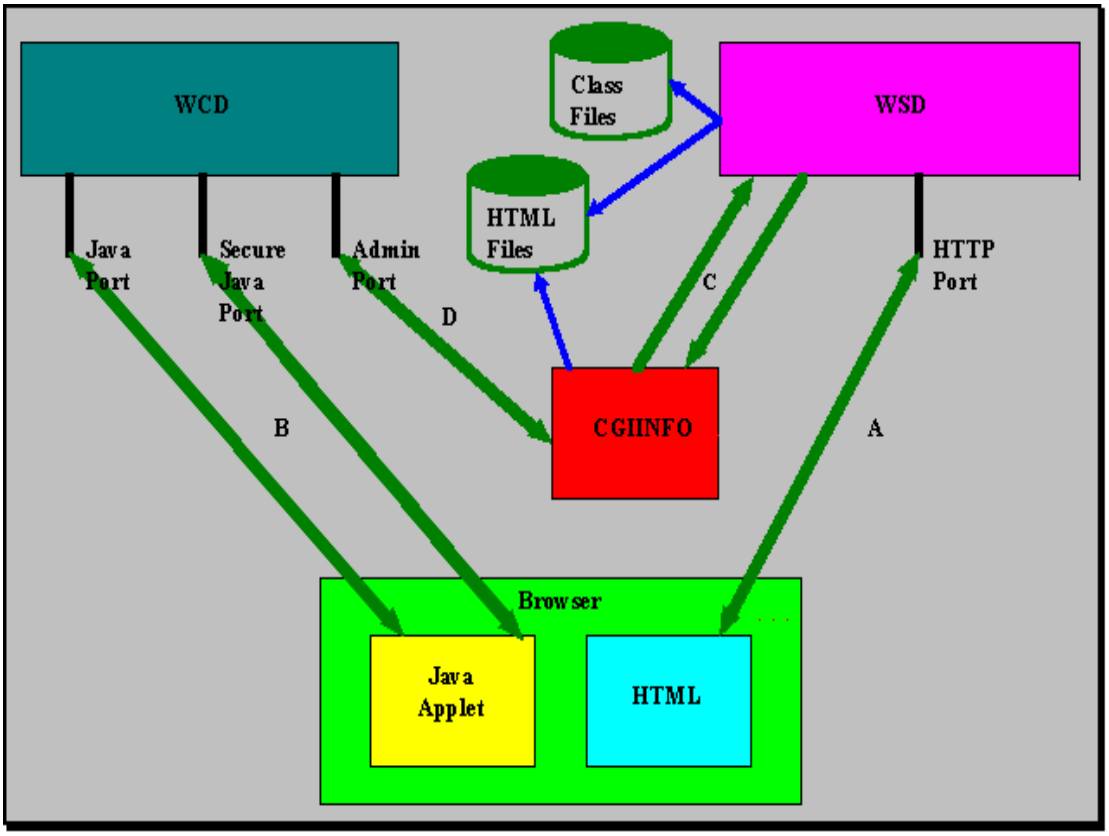


Establishing a Host Emulation Session from a Web Browser

OC://WebConnect provides persistent emulation session connectivity through the Java emulation client applets.(See Figure 1)

The JAVA enabled browser connects to the OC://WebConnect HTTP server or third-party HTTP server and downloads the Java emulation client applet as needed. The client applet functions to create and maintain an emulation client user interface (see connection A in Figure 1). The applet opens a *persistent socket connection* with the OC://WebConnect Emulation server (see connection B in Figure 1) which then completes the 3 tier model by opening a direct socket connection with the configured TN server. At this point there is a direct logical session connection from the Web Browser client to the host through the TN or Telnet server.

Figure 1: OC://WebConnect 3 Tier Model - TN/Telnet Server, Web Emulation Server, Java Applet Client



Full Featured Emulation

OC://WebConnect provides all the features of the traditional emulation clients with the centralized management of a server. Java applets are included for 3270, 3287 print, 5250 and VT emulation. Other major features include 3270E protocol support, print screen, copy/paste, hotspots, IND$FILE transfer, administrator or end user mapping of user interface options such as keyboard and color emulation. Administrative features include server session status, kill session, RTM (Response Time Monitor) and others.

More Information:

For more information about emulation features and session configuration creation, deletion, or modification refer to one of the following chapters within this document:

- Chapter 6: 3270 Session Configuration and Features
- Chapter 7: 5250 Session Configuration and Features
- Chapter 8: 3287 Print Session Configuration and Features
- Chapter 9: VT Session Configuration and Features

Installation, Configuration, and Maintenance

Easy Installation and Setup

The OC://WebConnect product has been designed for easy installation and maintenance. To deploy the OC://WebConnect product an administrator performs a one time installation of the full OC://WebConnect product on a UNIX or NT server, makes the necessary configuration changes to customize for network and host access, and starts the OC://WebConnect Emulation server and the OC://WebConnect HTTP server or 3rd party HTTP server.

The only software residing on the end user's desktop is the Java enabled browser that has already been installed for Internet or intranet access. The client software, a Java applet, is downloaded to the end user's desktop when the end user accesses OC://WebConnect HTTP server or 3rd party HTTP server via a browser and chooses to start an emulation session. *The client software is not installed on the desktop but is automatically downloaded from the server as needed only for the duration of the browser session.*

When the OC://WebConnect configuration changes or as new versions are made available, only the OC://WebConnect server platform needs to be updated. The end user's desktop will automatically be updated with the latest client software the next time a connection is made.

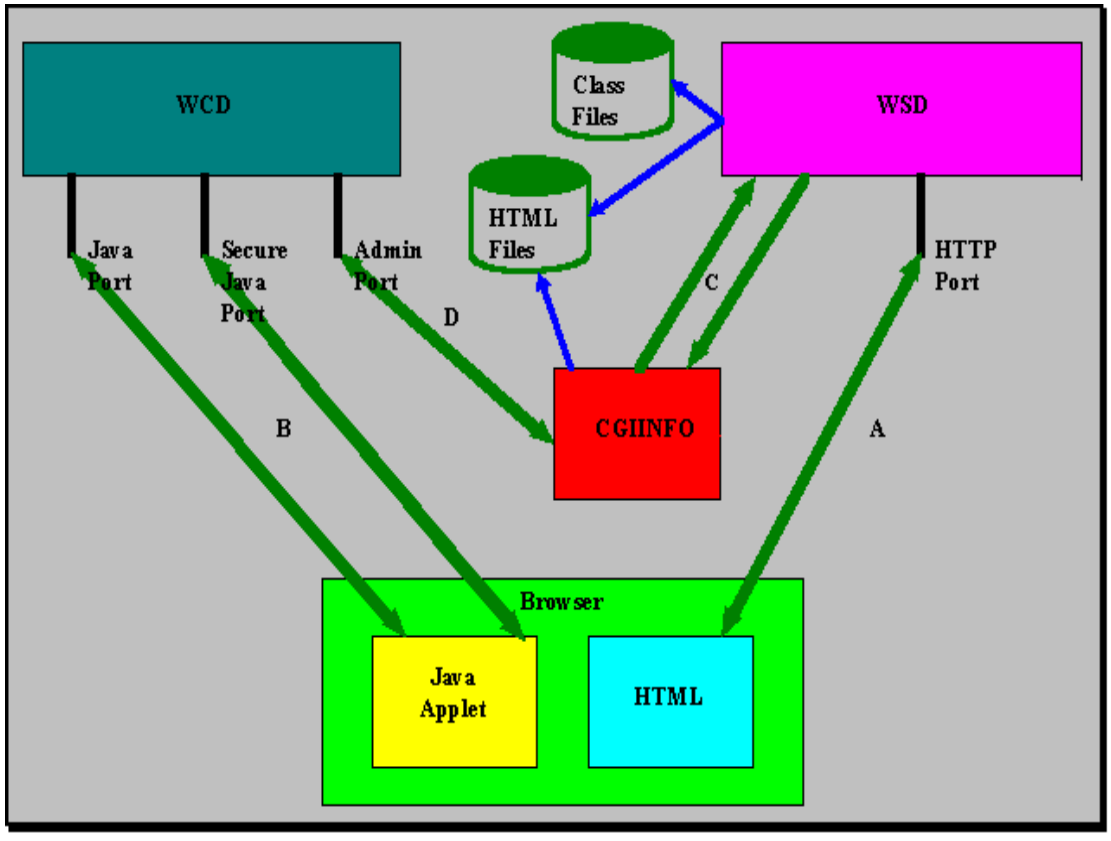
Configuration and Maintenance

OC://WebConnect includes graphical interfaces for configuration and administration task of server options such as port assignments, encryption choices, tracing, session status, session kill; as well as client session configuration for host access , keyboard mapping, color mapping and others. Server and client configuration and administration are provided via browser interface. An administrator can make the necessary server and client session configuration changes by either HTML pages or dynamically downloading a Java applet to do the same.

Dynamic Interface with HTTP Servers

In order to dynamically provide server and session information directly from the OC://WebConnect Web Emulation server with third-party HTTP servers, a method of querying data from the server is provided. The OC://WebConnect architecture uses an industry standard CGI bin interface available to most HTTP Web Servers. Requests to and data back from the OC://WebConnect Web Emulation server are handled by CGI-bin processes (see connection D for **cgiinfo** in Figure 1) which queries the OC://WebConnect Web Emulation server, and sends configuration and server data back to the Browser through the HTTP Server interface(see connection C in Figure 1). The CGI bin process is not permanent: it lasts only long enough to get and send data from a given request. Examples of the type of data requested are server status commands reporting on the connected sessions, list of configured and available sessions, hostname and encryption option information for individual session configuration, and applet download information. Use of the CGIbin interface is only required for a snapshot of the server and session status. Static HTML may be employed to download client applets.

Figure 1: OC://WebConnect 3 Tier Model - TN/Telnet Server, Web Emulation Server, Java Applet Client



Customization

The OC://WebConnect product can be customized to easily incorporate OC://WebConnect into an Intranet or Internet web site and provide the necessary centralized control of feature and host access. Some of the major areas in which OC://WebConnect can be customized are HTML user interface, session level options, emulation feature availability.

Through development of HTML pages an administrator or webmaster can control the "look and feel" of the initial OC://WebConnect interface, control the end user session configurations, control the emulation features-or client applets, which will be delivered to the end user's desktop. The initial HTML interface and download of applets can be served by either the OC://WebConnect HTTP server or a 3rd party HTTP server.

Through customized session configuration an administrator can create and make available a wide variety of session configurations that may vary in such ways as host, emulation configuration, level of session encryption, data stream compression and more. These options allow the administrator to provide a product to many end user groups, such as accounting which might require a very secure connection or customers connecting via a modem who require a quick no frills connection.
An administrator can choose to provide a range of display configurations such as keyboard mapping, color mapping or AutoGui interface or allow end users to configure their own display mappings which would be stored on their desktop. These capabilities allow an administrator to provide more end user control over their environment depending upon the variety and sophistication of the end users.

More Information:

- For more information about OC://WebConnect security refer to the *Chapter12: Customization of OC://WebConnect*

Security

OC://WebConnect provides a number of advanced security options. Some of the features included are Client Authentication, Server Authentication, data encryption and message authentication. By enabling combinations of options, varying levels of security can be established between the emulation and administration clients and OC://WebConnect Server.

OC://WebConnect Security Options:

RC4 from RSA Data Security, Inc. :

- Provides encrypted emulation sessions between client and server.
- 40 bit encrypted key·
- 128 bit encrypted key (Export Restrictions apply, US only)

Secure Socket Layer (SSL) from Netscape Communication:

- Server authentication via X.509 certificates
- Client/server encryption algorithm negotiations
- Message authentication protects against message tampering
- Cipher suites provide varying levels of security

Client Token Authentication:

- Client authentication via a secure token passing mechanism

More Information:
For more information about OC://WebConnect security features see *Chapter 15: Security Overview*

Chapter 2: Starting OC://WebConnect

Starting the OC://WebConnect Servers for UNIX

Note: If OC://WebConnect has not already been installed please refer to the *OC://WebConnect Installation Guide*.

To start the OC://WebConnect servers on a UNIX platform:

1. To start the OC://WebConnect emulation server execute ./**wcd** from the **wc** directory.

   ./wcd

1. When the emulation server has successfully started, a series of message are displayed showing the time the process started, the listening ports which have been established, the server key limit, and server process id.

   Example:

   Dec 10 11:07:00 - OC://WebConnect Started Wed Dec 10 11:07:00 1997
   Dec 10 11:07:00 - Process 27326 Started wcd
   Dec 10 11:07:01 - Key Session Limit: 8
   Dec 10 11:07:01 - Service 3270 Started jcpClient
   Dec 10 11:07:01 - Service 4224 Started apiClient
   Dec 10 11:07:01 - Process 27327 Detached wcd

3. Start the OC://WebConnect HTTP server, execute ./**wsd** from the **wc** directory.

./wsd

4. When the http server has successfully started a series of messages are displayed showing the time the process started, the listening port which have been established, and the server process id.

   Example:

   Dec 10 11:24:15 - OC://WebServer Started Wed Dec 10 11:24:15 1997

   Dec 10 11:24:15 - Process 29230 Started wsd

   Dec 10 11:24:15 - Service 2011 Started httpClient

   Dec 10 11:24:15 - Process 29231 Detached wsd

5. To access the *OC://WebConnect* HTTP Server enter the following URL (Universal Resource Locator) in a web browser:

   URL usage:

   http://[*hostname*]:[*port number of http web server*] .

   The following example uses the default setup options:

   http://host1.oc.com:2080

6. Once browser access has been established an emulation session may be started, further server and session configuration is possible, and online help is available.

Troubleshooting:

If either of the OC://WebConnect servers fails to start, use the OC://WebConnect Configuration utility, to reconfigure the port numbers and restart the servers. A port conflict may exist with another server running on the UNIX system.

To start the OC://WebConnect Configuration Utility from the OC://WebConnect directory (default directory is *wc*) type:

./configure

For more information about the configuration utility refer to the *OC://WebConnect Installation Guide* or *Chapter 5 Server Configuration and Administration.*

UNIX Command Line Options

The command line options for starting **wcd** and **wsd** are:

wcd *filename [-?] [-v] [-t filename] [-d description] [-l filename]*

or

wsd *filename [-?] [-v] [-t filename] [-d description] [-l filename]*

After OC://WebConnect initializes, it detaches from the controlling terminal and returns to the command shell. The **wcd** and **wsd** commands recognize the following command line option:

filename

If a ***filename*** is specified it is used instead of the default server configuration file (wcd.ini).. The file must have the same format as the default OC://WebConnect server configuration file, **wcd.ini**. and should be stored in the OC://WebConnect (default directory is *wc/cfgdir/ini)*. This method of an alternate configuration file may be employed to use the same server installation but specify different ports.

-?

Displays **wcd** and **wsd** usage information.

-v

Displays OC://WebConnect version information.

-t *filename*

Enables the OC://WebConnect daemon's trace facility; all trace information is written to the trace file specified by ***filename***. A trace file may be started after execution remotely by accessing the *OC://WebConnect* Server Administration Panel. For more information see *Chapter 5 Server Configuration and Administration.*

-d *description*

Inserts a ***description*** in the trace file. This option allows remarks used to identify the trace file and the circumstances to be included in the trace file . Enclose the ***description*** in quotes.

-l *filename*

Redirects the OC://WebConnect server's log file output to the file specified by ***filename*** and is stored in the OC://WebConnect log directory or *wc/logs* directory. By default, all log file output is sent to **stdout**. To direct output to a UNIX SYSLOG specify **-l SYSLOG**.

Stopping the OC://WebConnect Servers for UNIX

To stop the OC://WebConnect servers:

1. Determine each servers process id. This information was displayed when the server started or can be determined by using the UNIX *ps* command:

   ps |*grep wcd*

   and

   ps |*grep wsd*

2. kill each server by using the UNIX command:

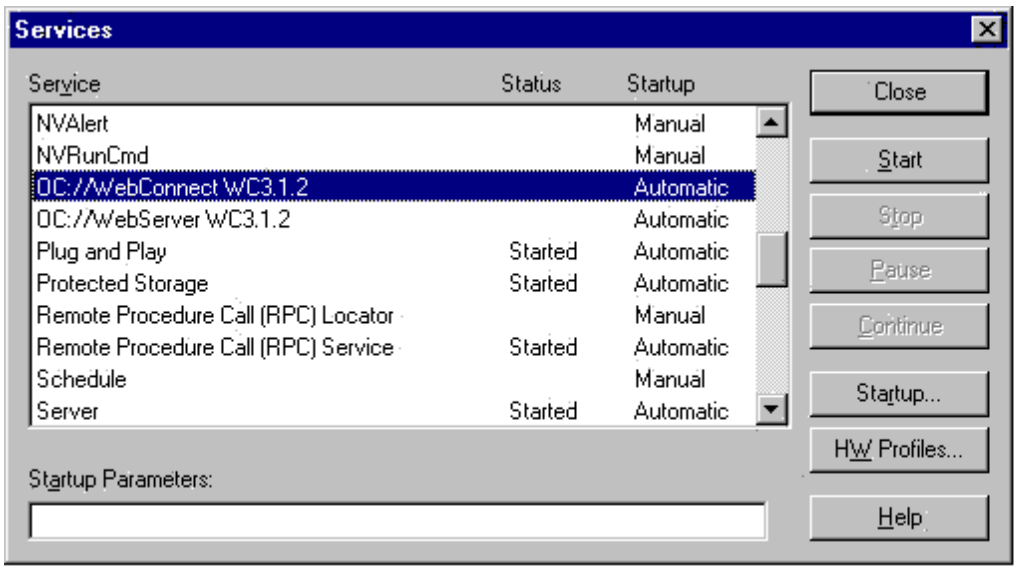
   kill *process id*

Starting the OC://WebConnect Services for Windows NT

***Note:** If OC://WebConnect has not already been installed please refer to the *OC://WebConnect Installation Guide*.

To start OC://WebConnect Emulation server on a Windows NT platform:

1. Select **Settings** from the **Start** menu on the Windows taskbar.
2. Select **Control Panel**.
3. Double click **Services**.
4. To start the *OC://WebConnect* Web Emulation server select **OC://WebConnect WC*rel #*** from the list box. Scroll down if necessary.
5. Click the **Start** button.



6. When the emulation server has successfully started the Services Panel will show the *OC://WebConnect* server status as "Started".
7. To start the *OC://WebConnect* http web server, select **OC://WebServer WC*rel #*** from the list box.
8. Click the **Start** button.
9. When the WebServer has successfully started the Services Panel will show the OC://WebServer status to be "Started"
10. Click the **Close** button to close the window.
11. To access the *OC://WebConnect* HTTP Server enter the following URL (Universal Resource Locator) in a web browser:

    URL usage:

    http://[*hostname*]:[*port number of http web server*] .

    The following example uses the default setup options:

    http://host1.oc.com:2080

12. Once browser access has been established emulation session may be started, further server and session configuration is possible, and online help is available.

Troubleshooting:

- If either of the *OC://WebConnect* services fails to start additional error messages will be logged to the Window NT Applications log. To access the Windows NT Event Viewer:
  1. Select the **Start** menu on the Windows NT task bar
  2. Select **Administrative Tools,**
  3. Select **Event Viewer**.

  Remember, *OC://WebConnect* server messages will be listed in the *Applications* log

- Use the **OC://WebConnect Configuration Utility** to resolve any server configuration problems. It may be necessary to reconfigure the server ports, if the chosen ports are being used by another service.

  To start the OC://WebConnect Configuration Utility:
  1. Select **Start** menu on the Windows NT task bar
  2. Select **OC://WebConnect rel#**
  3. Select **OC://WebConnect Configuration Utility**.

  For more information refer to the *OC://WebConnect Installation Guide* or *Chapter 5 Server Configuration and Administration*.

Logging server activity for OC://WebConnect for NT

By default the *OC://WebConnect* servers output only minimal messages, which are logged, when the services have been successfully started and stopped, to the Window NT event log. *OC://WebConnect* can log additional server activity such as logging each session connection and disconnection, each attempt to enter the Administrative tools, and each Administrative connection. To configure this option refer to the *OC://WebConnect Installation Guide* or *Chapter 5 OC://WebConnect Server Configuration and Administration.*

To access the Windows NT Event Viewer:

1. Select the **Start** menu on the Windows NT task bar
2. Select **Administrative Tools,**

3. Select **Event Viewer**.

Remember, *OC://WebConnect* server messages will be listed in the *Applications* log

Stopping the OC://WebConnect Servers for NT

To stop the *OC://WebConnect* servers:

1. Select **Settings** from the **Start** menu on the Windows taskbar.
2. Select **Control Panel**.
3. Select **Services**.
4. To stop the *OC://WebConnect* Web Emulation server select **OC://WebConnect WC*rel #*** from the list box. You may have to scroll down.
5. Click the **Stop** button.
6. To stop the *OC://WebConnect* http web server, select **OC://WebServer WC*rel #*** from the list box.
7. Click the **Stop** button.
8. Click the **Close** button to close the window.

Chapter 3: Navigating OC://WebConnect

Overview

The OC://WebConnect provides a browser based interface made up of a series of HTML pages to launch emulation sessions, HTML pages for administration and configuration, a Java applet for configuration and several Java applets for 3270, 5250, and VT emulation.

When a user or an administrator initial connects to the OC://WebConnect HTTP server, via a browser, the default index.html page is downloaded and displayed. The index.html page, or **Sessions** page, is the primary interface for end users and has links to the administration and configuration tools for an administrator. The administrative tools available from the index.html are the HTML **Administration** and **Configuration** pages, the **GUI Configurator** Java applet. For both end user and administration links are available for Context sensitive **Help** and a online **User's Guide**. Because the index.html page is HTML it can be customized as necessary.

To use OC://WebConnect for starting a session an end user will select which session configuration, set a few configuration options, and hit the **start** button.

To use OC://WebConnect for administration and configuration choose the Administration or Configuraiton buttons for HTML based tools or the GUI Config button for java based tools. Remote administration and configuration tools require an administrators password

The **Help** provide context sensitive HTML based help pages.

The **User's Guide** button provides HTML based documentation for both the administrator and end user.

General HTML Page Layout

Each OC://WebConnect HTML page is made up of at least 3 frames or sections

On the top is the ***Header*** section which shows which version of OC://WebConnect is being accessed and shows the HTML page being displayed, **Example:** Sessions, 3270 Configuration.

On the left are ***Navigation Buttons*** or links used to access other OC://WebConnect HTML pages or Java applets. These navigations buttons differ for each HTML page depending upon the page being displayed. For example the **Sessions** includes links for Administration, Configuration and GUI Configuration features as well as context sensitive help and the on-line User's Guide, a configuration page will include links to save or cancel the configuration changes and links to other more specific configuration pages.

The main section to the middle and right is the ***Selection and Input*** section. Within choices and input fields will be displaye for the end user or administrator to select options and input data. Buttons are also included to provide options such as Edit, Copy, Delete functions.

Sessions Page

The OC://WebConnect default page is broken up into three sections.

On the top is the ***Header*** section which shows which version of OC://WebConnect is being accessed and shows the HTML page being displayed, **Sessions**.

On the left are ***Navigation Buttons*** used to access other OC://WebConnect HTML pages for Administration, Configuration and GUI Configuration features as well as context sensitive help and the on-line User's Guide.

To the middle and right is the ***Session Select and Start*** section used to create, edit or delete session configurations.

Header Section

The Header section displays the OC://WebConnect log, version number, as **Sessions** to specify the HTML page.

Navigation Buttons:

- **Administration…** To perform HTML administrative functions such as to modify server settings; create, modify or delete emulation session settings.

- **Configuration…** To create, modify, or delete sessions and session features with HTML configuration. Server ports may also be reconfigured via this button.

- **GUI Config…** To create, modify, or delete sessions and session features and server ports using the Graphical Configurator.

- To access Context Sensitive **Help**

- To access Online **User's Guide**

- To refresh the list of session configurations

Session Select and Start Section

This section of the Sessions page includes a list of a available session configurations a choice of applet types, a choice to enable SSL security, and a choice of print methods for print screen or 3287 printing.

The **Sessions** list contains a list of available sessions that may be started. These include the default sessions originally configured during installation (Default 3270, 5250, VT and 3287) and any additional sessions created by the administrator. See *Chapter 4: Starting an Emulation Session* for more details.

The applet type list box contains a list of the available applet types.

- **Ultra Lite** enables all functionality available in OC://WebConnect version 2.6. A browser that supports Sun's JDK 1.1 is not required. Note: SSL is not supported for this setting.
- **Enhanced** contains the features available in OC://WebConnect version 3.1 including print, copy, and paste. A browser that supports Sun's JDK 1.1 is required.
- **Power User** contains the features available in OC://WebConnect version 3.1 and includes IND$File transfer, HotSpots, and Auto GUI features. A browser that supports Sun's JDK 1.1 is required.

The SSL list box give the end user the option to start a session using SSL authentication and encryption. The OC://WebConnect Emulation server can be configured to require the session use SSL.

- SSL Disabled
- SSL Enabled

The Print method list box alls the end user to use the default print method specified in the session configuration or override the session configuration

- **Default Print** uses whichever print option is defined in the session (.ses) file being used for that session.
- **OC://WebPrint** uses the OC://WebPrint solution for print screen functionality. OC://WebPrint must be installed on the browser platform to use this feature. This is supported on JDK 1.0 and great browsers.
- **JavaScript** is a print screen option which is embedded in some browsers. This option is supported in some JDK 1.0 and greater browsers.
- **JDK 1.1** print screen method is embedded in JDK 1.1 based browsers only.

To Start an Emulation Session

When starting a session, the user has four items to select before clicking the Start button.

1. Highlight the appropriate session to be started.
2. Choose an applet type: **Ultra Lite, Enhanced** or **Power User**.
3. Choose to use **SSL(Secure Socket Layer)** or not can be turned ON or OFF. The default is OFF. See *Chapter 13: Security Features* for more details.

4. Choose the printing option desired.

5. Hit the **Start** button to download the emulation applet and start an emulation session.

More Information:

A customized session screen (html page) may be created listing only those sessions available to users within your organization. See *Chapter 12: Customization of OC://WebConnect* for more details..

Administration Pages

The **Administration** button is used for performing a variety of OC://WebConnect HTML administration tasks.

Enter the appropriate password in the Administrator Password field and choose the OK button. The default password is "OCS". Since this password is documented, it is recommended that the administrator password be changed from the default.

If the correct Administrator password has been entered, the main Administration HTML page will be displayed.

Navigation Buttons:

- **Server…** To view server status and perform shutdown and restart of the OC://WebConnect server.
- **Sessions…** To view client session status and perform a kill on individual sessions
- **Tracing…**To perform tracing functions.

- To access Context Sensitive Help

- To access Online User's Guide

- To exit *Session Configuration*

Server

The administrator can view the server status, perform a shutdown and restart the OC://WebConnect server from this area.

Sessions

Session status may be viewed using this button. The administrator may also kill sessions by placing a check in the "kill" box. Select the Kill Sessions button and confirm when prompted.

Tracing

The **Tracing** button allows the administrator to start, stop, view, download and detele traces.

Done

Choose the **Done** button to exit the **Administration** HTML pages.

More Information:

For more information on the OC://WebConnect status pages, restart server, kill sessions, and tracings see *Chapter 5 Server Configuration and Administration*.

Configuration Pages

The **Configuration** button is used for performing configuration functions on sessions and the OC://WebServer.

Enter the appropriate password in the Administrator Password fieldl and choose the OK button. The default password is "OCS". Since this password is documented, it is recommended that the administrator password be changed from the default.

If the correct Administrator password has been entered, the main Configuration HTML page will be displayed.

More Information:

For more information about emulation features and session configuration creation, deletion, or modification refer to one of the following chapters within this document:

- Chapter 6: 3270 Session Configuration and Features
- Chapter 7: 5250 Session Configuration and Features
- Chapter 8: 3287 Print Session Configuration and Features
- Chapter 9: VT Session Configuration and Features

Navigation Buttons:

- **Sessions…** To configure an individual **Session configuration.** *This is the current page being displayed.*
- **Keyboard…** To create, modify, or delete a **Keyboard map**.
- **Attributes**… To create, modify, or delete an **Attribute and Color** map.
- **Hot Spots…** To create, modify, or delete a Hot Spot map.
- **Auto GUI…** To create, modify, or delete a Auto GUI map.
- **Servers…** To modify server settings

- To access Context Sensitive Help

- To access Online User's Guide

- To exit *Session Configuration*

Sessions, Keyboard, Attributes, Hot Spots and Auto GUI:

A list of existing session configurations is displayed. To edit, copy or delete an existing session select a session and choose a button. A group of radio buttons of the supported emulation types is displayed. To create a new session, choose an emulation type then select the New button.

Servers:

Select the Server button to edit the OC://WebConnect port configuration.

GUI Configurator Applet

After the OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) has been started, the GUI Configuration applet may be accessed by selecting the GUI Config button on any OC://WebConnect HTML page using a JDK1.1 JAVA enabled browser.

Administrator vs. User Mode

The GUI Configuration applet can be started in one of two modes, User mode or Administrative mode. This is determined by the " Allow User Configuration" option explained in detail below. Briefly, **Administrative mode** means, the "Allow User Configuration" option *has **not** been enabled*, the GUI Configuration applet is password protected, and if the correct password is entered the full GUI Configuration utility will be started. **User mode** means, the "Allow User Configuration" *has been enabled*, the user will not be prompted for a password, the GUI Configuration for end user will be displayed, all changes will be written to the browser platform not the server. The **Administrative mode** is accessible through the **User Mode** but is protected by password.

Use of the GUI Configurator in the Administrator mode

After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.

   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.

3. If the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:

OC://WebConnect Server
Modify IP addresses
Modify port numbers
Set GUI Configurator Cipher Suite
Allow User Configuration
Conceal Host Connection Information
Enable Client Token Authentication and Timeout
Set Client Are You There Timeout
Submit changes to save modifications

Password
Set New Administrator Password

License Key
Enter new software authorization key

Sessions
Create Sessions
Delete Sessions
Edit existing session configuration properties

More Information:

For more information on the OC://WebConnect Server, Password, and License Key tabs see *Chapter 5 Server Configuration and Administration*.

Help

Help displays OC://WebConnect's Context Sensitive help for the current screen. A separate HTML page opens containing information related to the current OC://WebConnect screen.

User's Guide

The **User's Guide** provides access to the OC://WebConnect's online documentation. A separate browser instance opens containing an outline with links to the various areas of the documentation.

Chapter 4: Starting an Emulation Session

Overview

OC://WebConnect provides a default HTML interface, accessed via a browser to start an emulation session. This interface allows emulation client users to select an emulation session configuration then automatically downloads the emulation software, starts a client session, and makes a connection to a S/390, AS/400, or UNIX host.

Like most web based products the initial interface to the product is an **index.html** page. By specifying the OC://WebConnect URL (host name and port) within the browser, contact can be made with the HTTP server and by default an **index.html** page is downloaded to the browser.

In the case of the OC://WebConnect product the HTTP Webserver downloads the default **index.html**, known as the **Sessions**" page. The **Sessions** page displays the available *emulation client session configuration*, the list of *emulation client applet packages*, a list to *enable or disable SSL* security, and a **Start** button to start a session.

The emulation client end user can select a session, hit the start button, a Java applet is downloaded to the browser platform, the applet is started by the browser, and a connection is made to the host configured for that session.

Buttons to access, *Context Sensitive Help*, an *Online User's Guide*, *Administrative* and *Configuration Tools* are also available on the **Sessions** HTML page.

The OC://WebConnect index.html page can be easily replaced with a customized index.html page. The default index.html page is available by default as an example and a tool for demonstrating and evaluating OC://WebConnect but may be used in a production environment. Some administrators may prefer to limit end users access to Administration and Configuration tools, restrict the ability of an end user to select session configurations, or just change the look and feel of the index.html page to blend into the corporate website. In addition to customizing the index.html page a 3rd party HTTP server may be used rather than the OC://WebConnect server.

More Information:
For more information about Customization of OC://WebConnect see *Chapter 12 Customization of OC://WebConnect.*

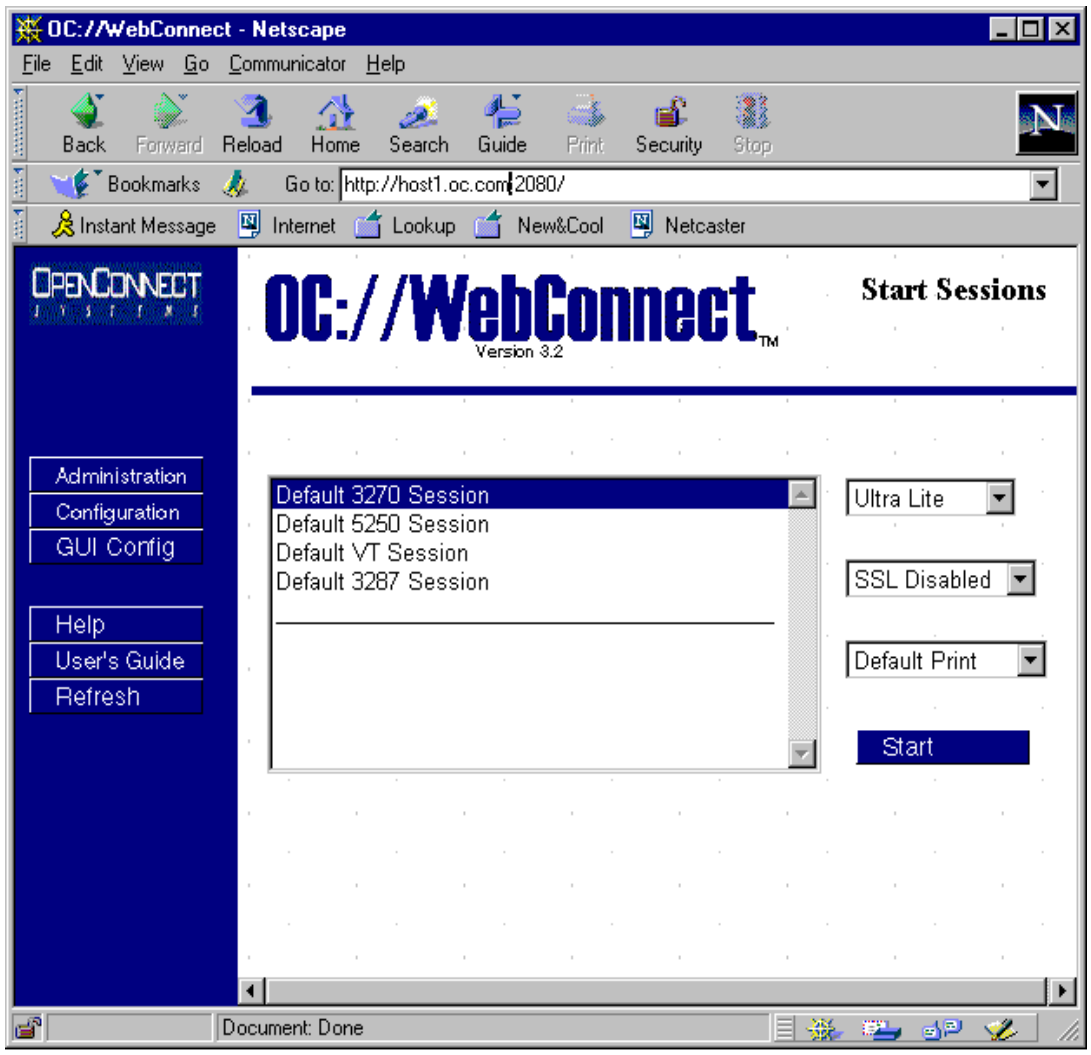
Sessions Page Layout

The **Sessions** page is made up of three sections, the *Header* section, *Navigation* section, and *Session Select and Start* section.

The *Header* section, at the top shows version information and identifies the HTML page as the "**Sessions**" page.

The ***Navigation*** section, to the left, provides links to OC://WebConnect administration and configuration tools, context sensitive help, and an online User's Guide.

The ***Session Select and Start*** section is the main section displayed to the right of the *Navigation* section. One list box shows a dynamic list of the emulation session configurations. When OC://WebConnect is first installed this is only the default session configurations. As the administrator adds, modifies and deletes session configurations this list automatically changes. Another list shows the OC://WebConnect applet types: **Ultralite, Enhanced,** and **Power User**. A list box is displayed to enable or disable **SSL** security. The last choice is a list box for choosing a print solution. An end user may override the session configuration by choosing either OC://WebPrint, JavaScript, or JDK 1.1 print. The **Start** button, downloads and starts a OC://WebConnect Java emulation client depending upon the choices made from the list boxes.


OC://WebConnect - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop
Bookmarks Go to: http://host1.oc.com:2080/
Instant Message Internet Lookup New&Cool Netcaster
OpenConnect Systems
OC://WebConnect™
Version 3.2
Start Sessions
Administration
Configuration
GUI Config
Help
User's Guide
Refresh
Default 3270 Session
Default 5250 Session
Default VT Session
Default 3287 Session
Ultra Lite
SSL Disabled
Default Print
Start
Document: Done

How to start an Emulation session

Note: The steps below describe how to start an emulation session using the **Sessions** page provided with OC://WebConnect in the form of the index.html file. If the index.html file has been customized the steps described may not work.

1. Access the Select Sessions Window by entering the URL of the OC://WebConnect HTTP server into a browser and hit return.

   Example:

   http://host1.oc.com:2080

1. The default index.html, **Sessions**, page is displayed.

2. A list of the available session configurations is displayed. Select a session configuration. The list of available session configurations is dependent upon the OC://WebConnect administrator. The administrator should create, modify, or delete sessions as necessary to meet the end users needs. The description of each session configuration listed is entered by the administrator and should be used as a description that is meaningful to the end users.

3. A list of OC://WebConnect applet packages is displayed. Choose an emulation client applet package.

   The emulation applet packages (Ultralite, Enhanced, and Power User) are available to meet the different user environments and needs.

   - **Ultra Lite** enables all functionality available in OC://WebConnect version 2.6. A browser that supports Sun's JDK 1.1 is not required. Note: SSL is not supported for this setting.
   - **Enhanced** contains the features available in OC://WebConnect version 3.1 including print, copy, and paste. A browser that supports Sun's JDK 1.1 is required.
   - **Power User** contains the features available in OC://WebConnect version 3.1 and includes IND$File transfer, HotSpots, and Auto GUI features. A browser that supports Sun's JDK 1.1 is required.

5. Choose to enable SSL authentication and encryption for this session connection or not. This choice is a security vs. time consideration. For more information about SSL see *Chapter 15 Security Overview*.

6. The Print method list box shows the available print solutions for print screen or 3287 print.

    - **Default Print** uses whichever print option is defined in the session (.ses) file being used for that session.

    - **OC://WebPrint** uses the OC://WebPrint solution for print screen functionality. OC://WebPrint must be installed on the browser platform to use this feature. This is supported on JDK 1.0 and greater browsers.

    - **JavaScript** is a print screen option which is embedded in some browsers. This option is supported in some JDK 1.0 and greater browsers.

    - **JDK 1.1** print screen method is embedded in JDK 1.1 based browsers only.

7. Hit the **Start** button.

8. A Java emulation applet is downloaded into the memory of the browser platform the 1st time the applet package is chosen during the browser session. Additional applet starts do not require another download.

    While the applet is downloading, the browser displays information about the files being downloaded in a status line usually displayed at the bottom of the browser window.

8. When the applet download is completed the browser starts the applet and the applet window is displayed.

9. Once the applet has been started the emulation client automatically makes a connection with the OC://WebConnect Emulation server.

10. If RSA encryption has been chosen encrytion keys are generated and exchanged. All data, at this point, between the emulation client applet and the OC://WebConnect Emulation server is encrypted.

11. The OC://WebConnect Emulation Server makes a connection to the S/390, AS/400, or UNIX host specified in the session configuration.

12. Emulation data begins to flow and the host data is displayed within the emulation client applet window.

More Information:
For more information about emulation client features and how to choose the appropriate emulation client type see *Chapter 11 Emulation Client Interface and Features.*

Chapter 5: Server Configuration and Administration

Overview

OC://WebConnect provides three methods for Configuration and Administration of the OC://WebConnect servers and sessions. The first, the **OC://WebConnect Configuration Utility,** is a limited stand alone UNIX or NT script. It is used prior to starting the OC://WebConnect Servers to configure server ports and IP addresses, minimum default session settings, server language, sensitive security settings, and default HTML files. The **OC://WebConnect** Configuration Utility does not require a browser.

The next method is **HTML Administration and Configuration**. The HTML Administration and Configuration is a remote method of configuration which involves access to the **OC://WebConnect** Server, which has already started, via a browser. HTML pages and a CGI Bin interface are used to configure and administer **OC://WebConnect** servers and sessions and does not require a JAVA enabled browser.

The last method, the **Graphical (GUI) Configurator**, is a java applet. The GUI Configurator is also a remote method which involves the download and execution of a java applet via a JAVA enabled (JDK 1.1) browser.

The **HTML Administration and Configuration** and **GUI Configurator** both provide the full featured configuration. Two different configuration methods are provided for the differing needs of **OC://WebConnect** users.

OC://WebConnect Configuration Utility

The **OC://WebConnect Configuration Utility** is a stand alone UNIX script or NT program used to configure the OC://WebConnect servers and default sessions prior to starting the OC://WebConnect servers. The Configuration Utility must be executed from the platform on which OC://WebConnect has been installed.

The following items may be configured using the Configuration Utility:

- OC://WebConnect Emulation Server ports
- OC://WebConnect HTTP Server port
- Host names and ports for the default 3270, 5250, 3287, and VT session configurations
- Server License Key
- Server or Administration Language (English, German, French, or Castilian Spanish).
- Generate Secure Socket Layer(SSL) key pair, set SSL password, and enable SSL. *Please note that Key generation is only provided via the Configuration Utility for security reasons.
- Modifies HTML used to access, configure, and administer the OC://WebConnect servers, and download emulation sessions.

The **OC://WebConnect Configuration Utility** is not provided through a browser connection due to security reasons.

Accessing the Configuration Utility for UNIX

To Configure OC://WebConnect for UNIX using the configuration utility:

1. Execute the configure script, by entering the following command from the OC://WebConnect directory (default is **wc)** :

   ./configure

2. The following menu will be displayed:

1)	Configure WebConnect Ports
2)	Configure Default 3270 Session
3)	Configure Default 5250 Session

4)	Configure Default VT220 Session
5)	Configure Default 3287 Session
6)	Configure License Key Information
7)	Configure Default Administration Language
8)	Configure WebServer HTTP Port
9)	Configure WebConnect SSL
0)	Exit

3. Select the number of each configuration option, as needed. See each configuration item discussed below.

4. After completing each server configuration option the OC://WebConnect HTML files will automatically be updated. This is relevant for anyone using the OC://WebConnect provided HTML files either directly or as a model for customization. All files in the OC://WebConnect html directory will be scanned for host name, port parameters, and server language then updated with the current settings. Any HTML files stored in another directory will not be updated.

   Failure to update HTML files may make it difficult to access and configure OC://WebConnect via a browser, start an emulation session, or retrieve server status information.

5. If the RETURN key is pressed each time the main menu is displayed, the configuration will auto-sequence through each of the menu items.

6. Choose menu item **0)Exit** when configuration has been completed.

7. For the changes to take affect **Restart** the OC://WebConnect servers after exiting the configuration utility.

Accessing the Configuration Utility for Windows NT

To start the OC://WebConnect Configuration Utility:

1. Select **Start** menu on the Windows NT taskbar
2. Select **OC://WebConnect rel#**
3. Select **OC://WebConnect Configuration Utility**.

   After completing each server configuration option the OC://WebConnect HTML files will automatically be updated. This is relevant for anyone using the OC://WebConnect provided HTML files either directly or as a model for customization. All files in the OC://WebConnect html directory will be scanned for host name, port parameters, and server language then updated with the current settings. Any HTML files stored in another directory will not be updated.

4. Current configuration information is displayed , press RETURN to accept the current information or change the information that needs to be modified. See each configuration item discussed below.
5. For the changes to take affect restart the OC://WebConnect servers after exiting the configuration utility.

OC://WebConnect Configuration Options

Option 1) Configure WebConnect Ports

Each OC://WebConnect service can listen to incoming requests and send data back on one or all of the network interfaces installed on a UNIX platform. The configuration utility lists all of the IP addresses for the network interfaces detected The default, 0.0.0.0, will cause OC://WebConnect servers to respond to requests from all installed network interfaces. Specify another IP address to limit OC://WebConnect to one network interface.

OC://WebConnect Emulation server can use up to three ports during operation. The default port setting may be used or enter a port number greater than 0 or less than 65,535. Root privileges are required to use a port number less than 1024. A port number of 0 will disable a port.

Default Settings:

PORT NUMBER	SERVICE	DESCRIPTION
3270	Java Server	Listening port for use by non-SSL java emulation clients. Required.
3443	Secure Java Server	Listening port for use by SSL java emulation and administration clients. (Optional if not using SSL)
4270	Java Administration	Listening port for use by the CGIbin interface to obtain configuration parameters to launch applets and for retrieving server status information. (Optional if using static html and not reporting server Status information.)

For more information about ports see the Server Ports section of the chapter.

2) Configure Default 3270 Session

This selection allows the configuration of the Domain Name Server (DNS) host name or IP address and TCP port address of a TN3270 server , TN3270E server, or gateway for mainframe emulation access. This information is used to create the default 3270 session configuration. Other default session settings and additional 3270 sessions may be configured later using the OC://WebConnect HTML Configuration or OC://WebConnect GUI Administration Client.

3) Configure Default 5250 Session

This selection allows the configuration of the DNS host name or IP address and TCP port address of a TN5250 server or gateway for midrange emulation access. This information is used to create the default 5250 session configuration. Other default session settings and additional 5250 sessions may be configured later using the OC://WebConnect HTML Configuration or OC://WebConnect GUI Administration Client.

4) Configure Default VT Session

This selection allows the configuration of the DNS host name or IP address and TCP port address of a Telnet server or gateway for ASCII terminal emulation access. This information is used to create the default VT session configuration. Other default session settings and additional VT sessions may be configured later using the OC://WebConnect HTML Configuration or OC://WebConnect GUI Administration Client.

5) Configure Default 3287 Session

This selection allows the configuration of the DNS host name or IP address and TCP port address of a TN3270 server , TN3270E server, or gateway for mainframe print emulation access. This information is used to create the default 3287 Print session configuration. Other default session settings and additional 3287 sessions may be configured later using the OC://WebConnect HTML Configuration or OC://WebConnect GUI Administration Client.

6) Configure License Key Information

OC://WebConnect comes prepackaged with a license key to enable the server for a specific number of concurrent sessions and key expiration. Press RETURN to accept the default key. If a special or replacement key has been provided enter the key at this time. The number of concurrent sessions and expiration date for the key configured can be seen when the OC://WebConnect servers are started, on the OC://WebConnect STATUS page, log file, or trace file.

7) Configure Default Administration Language

OC://WebConnect can be configured to one of four possible server languages. The OC://WebConnect Server Language is used for the HTML configuration pages, the Graphical (GUI) Configuration client, the OC://WebConnect HTML session selection pages, and Online User's Guide. When the server language is changed the HTML files provided with OC://WebConnect will automatically be updated which will include any previous configured server host names or ports. **For more information about ports see the Server Ports section of the chapter.**

8) Configure WebServer HTTP Port

The OC://WebConnect HTTP Web Server can listen to incoming requests on one or all of the network interfaces installed on a UNIX platform. The configuration utility lists all of the IP addresses for the network interfaces detected and defaults to 0.0.0.0 which will cause OC://WebConnect servers to

respond to requests from all installed network interfaces. Specify another IP address to limit OC://WebConnect to one network interface.
To use the OC://WebConnect HTTP Web server, enter a TCP port number for the HTTP service. This is the port number which is used when accessing OC://WebConnect via browser.

Example:

http://host1.oc.com:2080

The default OC://WebConnect HTTP Web server port is 2080. Many HTTP Web servers use port 80 because most browser default to port 80, therefore the browser user will only have to enter the Web server host name and not a port.

Example:

http://host1.oc.com

A port number of 0 will disable the OC://WebConnect HTTP Web Server. **For more information about ports see the Server Ports section of the chapter.**

9) Configure WebConnect SSL

To use OC://WebConnect SSL authentication and encryption features either a key pair and certificate, or a "generate a certificate request" must be generated. If a key pair and certificate is generated answer YES to enable SSL At this point SSL will be fully operational when the OC:/Webconnect server is started.

If the "generate a certificate request" is chosen, the request must be submitted a Certificate Authority(CA). SSL cannot be used until the certificate has been received from the CA, manually installed in the OC://WebConnect security directory, and the configure utility is rerun to enable SSL. When executing the configure utility, answer NO when asked to generate a new key pair, then answer YES when asked to enable SSL.

SSL Key Pair & Certificate Generation

OC://WebConnect must be setup with a key pair and certificate before the SSL features can be used. Specific information is required about the length of key and company to generate the RSA key pair and certificate, or a certificate request, for the OC://WebConnect server. Each panel will present detailed information concerning a particular question, followed by the actual question.
For the optimum performance/convenience vs. security trade-off, the default settings are recommended.

The following questions are asked:

1. Choose a value between 512 and 2048 bits for the RSA modulus length? [1024]

If 512 bit modulus is chosen, skip step 2 and proceed to directly step to 3.

2. Generate server-wide key exchange key pair (yes/no)? [yes]

This question is only relevant if exportable(40-bit) ciphers will be used with this installation. If "yes" is chosen, a 512-bit key will be used for these ciphers, rather than waiting until session startup. This will improve session connect times and help prevent the server from becoming bogged down computing keys on heavily loaded servers.

3. Store password on server system (yes/no)? [yes]

A password is used to secure the server's private key. The system administrator will need to type in this password each time OC://WebConnect is started, making unattended restarts impossible, unless the password is stored on the server system. The administrator must choose between the convenience of unattended restart or the additional security.
Regardless of whether the password is stored on the server, the OC://WebConnect security directory must be access-protected to prevent potential attackers from compromising the server. With this perspective, the slight reduction in security from storing the password on the server may be a reasonable trade-off for the increased convenience of having an automatic restart capability.

4. *The password may be any combination of displayable characters, including spaces, up to 100 characters in length.*
   Shall I turn off echo while you enter the password (yes/no)? [yes]
   Enter the password at this time:

After entering the password, the RSA key pair will be generated. This may take anywhere from a couple of seconds for shorter keys, to over an hour for extremely long keys. A 1024-bit key should normally complete within a minute or two depending on the system. A second key, 512 bits, will be generated a server-wide key exchange key pair was selected.

5. Specific site information is needed to generate a certificate request. This information pertains to the name and location of the server.

DNS NAME OF SERVER: [HOST NAME]
Company name or organization
Organizational unit, division, etc. (this field is optional)
City
State
Country (use ISO Country Code -- do not spell out): [US]

The data entered in these fields will comprise the X.500 "distinguished name" of the subject listed in the body of the certificate. If a built-in certificate generator or a private CA will be used, then what is entered in these fields is somewhat arbitrary, but is intended to uniquely identify the holder of the certificate. If a third-party CA will be used, it is important that the name be unique, and all fields accurate. The “State” should be spelled out.

6. Shall I generate the Certificate, or shall I generate a certificate. Request instead? Generate certificate (yes/no)? [yes]

Choose YES to allow the built-in certificate generator to generate a certificate, or NO to generate a PKCS #10 certificate request to be submitted to a third-party or private CA.
The default method of server authentication used by OC://WebConnect SSL clients is to compare the computed fingerprint of the server's certificate to the fingerprint received as an applet startup parameter from the web server. Therefore, it is not necessary to use a CA to generate the OC://WebConnect server certificate. The setup for server authentication will be handled automatically if the default method is chosen and the OC://WebConnect provided CGIbin for applet startup is used.

Alternatively, if No is chosen only a certificate request is generated. The request will have to submitted to a CA and manually install the certificate into the security directory and rerun configure. With a CA-generated certificate, it may be desirable for the clients to authenticate the server using the CA's certificate rather than the server's. This approach can provide a more centralized security model, but is more cumbersome to implement.
If a CA is chosen instead of the built-in certificate generator, skip to question #8.

7. Term of validity for certificate in hours: [8760 (1 yr)]

The certificate will be generated with a validation period starting at the time the certificate is generated. The period of time entered here will determine the expiration date of the certificate.
At this point the certificate will be generated without asking question #8.

8. More information is needed concerning the person responsible for receiving the certificate from the CA.

E-MAIL ADDRESS
Phone number

Fill out these fields and press RETURN. Third party and private CA's will use the phone number or e-mail address in the request to contact the person responsible, if additional information is needed, or if problems arise. The completed certificate will typically be delivered via e-mail, in base 64 encoding, to the e-mail address provided in the request.

The server certificate must be stored with the CA certificate(s), all base 64-encoded, in a file named cert.txt, with server certificate first to root CA certificate last order, in the security sub-directory of the OC://WebConnect home directory. After the certificate has been installed, rerun the "configure" utility to enable SSL.

To make OC://WebConnect clients validate to the CA rather than the server certificate, HTML will have to be created for applet startup with the following applet parameter:

<param name="certfpca" value="xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx">

where the x's represent the fingerprint (MD5 hash) of the CA certificate in hexadecimal.

OC://WebConnect HTML Administration

The OC://WebConnect HTML Administration and Configuration pages are HTML pages which provide the ability to administer and configure the OC://WebConnect Emulation server and emulation sessions. The OC://WebConnect Emulation server must be running for these tools to work. By using the HTML Administration pages it is possible to remotely configure and administer the following options:

- Restart OC://WebConnect Emulation server
- Enable OC://WebConnect tracing
- Shutdown OC://WebConnect Emulation server
- Display the status of all current sessions
- Kill individual Emulation sessions

More Information:

The Administration button can be removed from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet. See *Chapter 12 Customization of OC://WebConnect* in this document for more information.

To Access the HTML Administration pages

After the OC://WebConnect server *(See Chapter 2: Starting OC://WebConnect)* has been started, the HTML Administration may be accessed via a browser by selecting the Administration link on the main OC://WebConnect HTML pages.

1. Connect to the OC://WebConnect HTTP Webserver. Enter the host name and tcp port number in the URL of a browser

   Example:

   http://host1.oc.com:2080

2. Choose the **Administration** button displayed on the left side of the **Sessions** page.

3. A prompt will appear for the **Administrator Password**.

4. Enter the appropriate password and choose the **OK** button. The default password is "OCS". Since this password is documented it is recommended that the administrator password be changed from the default as soon as possible.

5. The HTML Administration page should be displayed.

HTML Administration Page Layout

The HTML Adminstration page is made up of three frames or sections, the ***header, Navigation Buttons***, and ***Adminstration*** sections (**Server Status, Session Status, Tracing**).

On the top is the ***Header*** section which shows which version of OC://WebConnect is being accessed and show the HTML page being displayed. In this case it should display **Administration Server Status** in the upper right corner.

On the left are **Navigation Buttons** used to access other OC://WebConnect HTML pages for current sessions, tracing and documentation.

To the middle and right is Administration section used to display information about the current server status and current users, restart the server, shutdown the server, kill current sessions and enable tracing.

Navigation Buttons:

- **Server…** To display the OC://WebConnect Emulation server status, restart the server, or shutdown the server.
- **Sessions…** To display the sessions currently connected to the OC://WebConnect server or kill individual sessions.
- **Tracing…** To enable or disable tracing..
- To access **Context Sensitive Help**
- To access **Online User's Guide**
- To **exit** Administration pages

Server Status

The server status page includes information about the OC://WebConnect Emulation server and the ability to Restart or Shutdown the OC://WebConnect Emulation Server.

Information included:

FIELD	DESCRIPTION
Host	The DNS host name or IP address where the OC://WebConnect Emulation Server is running.
Server Version	The version of the OC://WebConnect Emulation server.
Started	The time which the current instance of the OC://WebConnect Emulation server was started.
UpTime	The amount of time between the current time and the time the OC://WebConnect Emulation server was started.
Key Expiration	The date on which the user license for OC://WebConnect will expire.
Process ID	The UNIX process id for the OC://WebConnect Emulation server.
Session Limit	The limit to the number of concurrent emulation sessions.
Active Sessions	The number of sessions current connected to the OC://WebConnect Emulation server. This includes emulation and configuration sessions.

Session Status

The Session Status page includes information about all sessions currently connected to the OC://WebConnect Emulation server and the ability to one or more session connections.

Information included:

FIELD	DESCRIPTION
Kill	This box marks a session to be killed. Use the kill session button to kill all the marked sessions.
ID	The OC://WebConnect ID for this session. This ID corresponds to the ID in trace files.
Type	The type of session connection. Example CGI-BIN, 3270, 5250.
IP Address	The IP address of the workstation on which the emulation or configuration session is running.
Connect Time	The date and time the session connection was made.
Last Response	The last time a response from the session was received by the server.
Bytes Sent	The number of bytes sent from the session to the OC://WebConnect server.
Bytes Received	The number of bytes received by the session from the OC://WebConnect server.

Tracing

The Tracing page provides a facility for starting and stopping an OC://WebConnect trace while the server is running. On this HTML page trace files may also be viewed, downloaded and deleted.

Display Server Status

To display the OC://WebConnect Emulation server status:

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Server status page should display

Restart the Server

To **Restart** the OC://WebConnect Emulation server:

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Server status should display
4. Select the **Restart Server** button.

   Note: All sessions connected to the OC://WebConnect server will be disconnect when the server is restarted, including the Administration session.

Shutdown the Server

To Shutdown the OC://WebConnect Emulation server:

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Server status should display
4. Select the **Shutdown Server** button.
5. All sessions connected to the OC://WebConnect server will be disconnect when the server is restarted, including the Adminstration session.
6. The server will have to be restarted from the UNIX command line or the NT Services panel.

Display Current Session Status

To display the status of the sessions currently connected to the OC://WebConnect server :

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. Choose the **Sessions**… button
4. The Session status page should display

Kill a Current Session

To kill a session that is currently connected to the OC://WebConnect server:

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. Choose the **Sessions**… button
4. The Session status page should display
5. In the **Kill** column next to each session mark the sessions to be killed.
6. Choose the **Kill Session** button.

Start an OC://WebConnect Trace

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. Choose the **Tracing**… button
4. The Tracing page should display
5. Enter a filename for the new trace file.
6. Choose the **Start Tracing** button.

To view an OC://WebConnect Trace

1. Select the **Administration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. Choose the **Tracing**… button
4. The Tracing page should display
5. Choose a trace file from the list box of trace files
6. Choose what data to view: JCP, Telnet/RUI, HTTP.
7. Choose the **View Tracing** button.

OC://WebConnect HTML Configuration

The OC://WebConnect HTML Configuration pages are HTML pages which provide the ability to configure the OC://WebConnect Emulation server and emulation sessions. The OC://WebConnect Emulation server must be running for these tools to work. By using the HTML Configuration pages it is possible to remotely configure and administer the following options:

- Modify the OC://WebConnect Emulation Server ports
- Modify OC://WebConnect Administration ports
- Select Cipher suites
- Configure to Conceal Host Connection Information from Client Emulation users.
- Enable and configure Client Token authentication
- Modify the Administration Password
- Modify the Server License Key

More Information:

The Configuration button can be removed from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet. See *Chapter 12 Customization of OC://WebConnect* in this document for more information.

HTML Configuration Page Layout

After the OC://WebConnect server *(See Chapter 2: Starting OC://WebConnect)* has been started, the HTML Administration and Configuration pages may be accessed, via a browser by selecting the Administration or Configuration link on the main OC://WebConnect HTML pages.

The HTML Server Configuration page is made up of 3 frames or sections, the ***header, Navigation Buttons***, and ***Configuration*** sections.

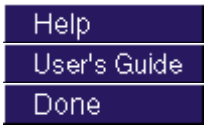
On the top is the ***Header*** section which shows which verson of OC://WebConnect is being accessed and show the HTML page being displayed. In this case it should display **Server Configuration** in the upper right corner.

On the left are ***Navigation Buttons*** used to access other OC://WebConnect HTML pages for configuration Server IP address, ports, security features, and administrator password

To the middle and right is the ***Server Configuration*** section used to make server configuration choices and input server configuration data.

Navigation Buttons:

- **Services…** To input the IP address and port for the JCP, Secure JCP, and Adminstration services. JCP is the service for non SSL emulation sessions. Secure JCP is the service for emulation session that are secured by SSL. The Admin service is for the CGI-BIN connections used to query the server for current server and session information.
- **Security…** To enable client token authentication, suppress host information, set the SSL cipher suite, and set the adminstrator password..
- **Misc…** To set the OC://WebConnect license key and enable JCP Client Are You There.



- To access **Context Sensitive Help**
- To access **Online User's Guide**
- To **exit** *Server Configuration pages*

To set Server IP addresses and ports

1. Select the **Configuration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Session Configuration page is displayed
4. Choose the **Servers…** button on the left
5. The Configuration Servers page is displayed
6. Select the server to configure and press the **Edit** button
7. The Server Configuration page is displayed
8. Input the IP address for each service.

   Each OC://WebConnect service can listen to incoming requests and send data back on one or all of the network interfaces installed on a UNIX platform. The configuration utility lists all of the IP addresses for the network interfaces detected The default, 0.0.0.0, will cause OC://WebConnect servers to respond to requests from all installed network interfaces. Specify another IP address to limit OC://WebConnect to one network interface.

9. Input the Port number for each service.

   OC://WebConnect Emulation server can use up to three ports during operation. The default port setting may be used or enter a port number greater than 0 or less than 65,535. Root privileges are required to use a port number less than 1024. A port number of 0 will disable a port.
10. Choose the save button
11. For the changes to take affect restart the OC://WebConnect server. HTML files used to access OC://WebConnect will have to be updated. Use the OC://WebConnect Configure Utility to update these files or manually edit the HTML files.

To set Server Security options

1. Select the **Configuration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Session Configuration page is displayed
4. Choose the **Servers…** button on the left
5. The Configuration Servers page is displayed
6. Select the server to configure and press the **Edit** button
7. The Server Configuration page is displayed
8. Choose the **Security…** button
9. The Security page is displayed
10. Make the necessary security changes
11. Choose the **Save** to save changes made.

To set the License Key

1. Select the **Configuration** button from the **Sessions** HTML page.
2. Enter the Administrator's password.
3. The Session Configuration page is displayed
4. Choose the **Servers…** button on the left
5. The Configuration Servers page is displayed
6. Select the server to configure and press the **Edit** button
7. The Server Configuration page is displayed
8. Choose the **Misc…** button
9. The input box for the License key is displayed with the current key.
10. Input the new license key
11. Choose the **Save** button

OC://WebConnect GUI Configuration

The OC://WebConnect GUI Configuration utility is a JDK 1.1 java applet which is accessed via a JAVA enabled (JDK 1.1) browser. The OC://WebConnect Emulation server must be running for these tools to work. By using this java applet, it is possible to remotely configure the following options:

- Modify the OC://WebConnect Emulation Server ports
- Modify OC://WebConnect Administration ports
- Select Cipher suites
- Set "Allow User Configuration" to allow users to configure their own User Interface features such as keyboard map and color map
- Configure to Conceal Host Connection Information from Client Emulation users.
- Enable and configure Client Token authentication
- Configure Client "Are You There?"
- Modify the Administration Password
- Modify the Server License Key
- Restart OC://WebConnect Emulation server
- Enable OC://WebConnect tracing
- Shutdown OC://WebConnect Emulation server
- Kill individual Emulation sessions
- Create, delete, or modify all features of all session configurations including mapping keyboards, colors, attributes, etc.

After the OC://WebConnect server *(See Chapter 2: Starting OC://WebConnect)* has been started, the GUI Configuration applet may be accessed by selecting the Administration button on any OC://WebConnect HTML page using a JDK1.1 JAVA enabled browser.

More Information:

The Administration button can be removed from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet. See *Chapter 12 Customization of OC://WebConnect* in this document for more information.

The GUI Configuration applet can be started in one of two modes, User mode or Administrative mode. This is determine by the " Allow User Configuration" option explained in detail below. Briefly, **Administrative mode** means, the "Allow User Configuration" option *has **not** been enabled*, the GUI Configuration applet is password protected, and if the correct password is entered the full GUI Configuration utility will be started. **User mode** means, the "Allow User Configuration" *has been enabled,* the user will not be prompted for a password, the GUI Configuration for end user will be displayed, all changes will written to the browser platform not the server. The **Administrative mode** is accessible through the **User Mode** but is protected by password.

Logging on as an Administrator

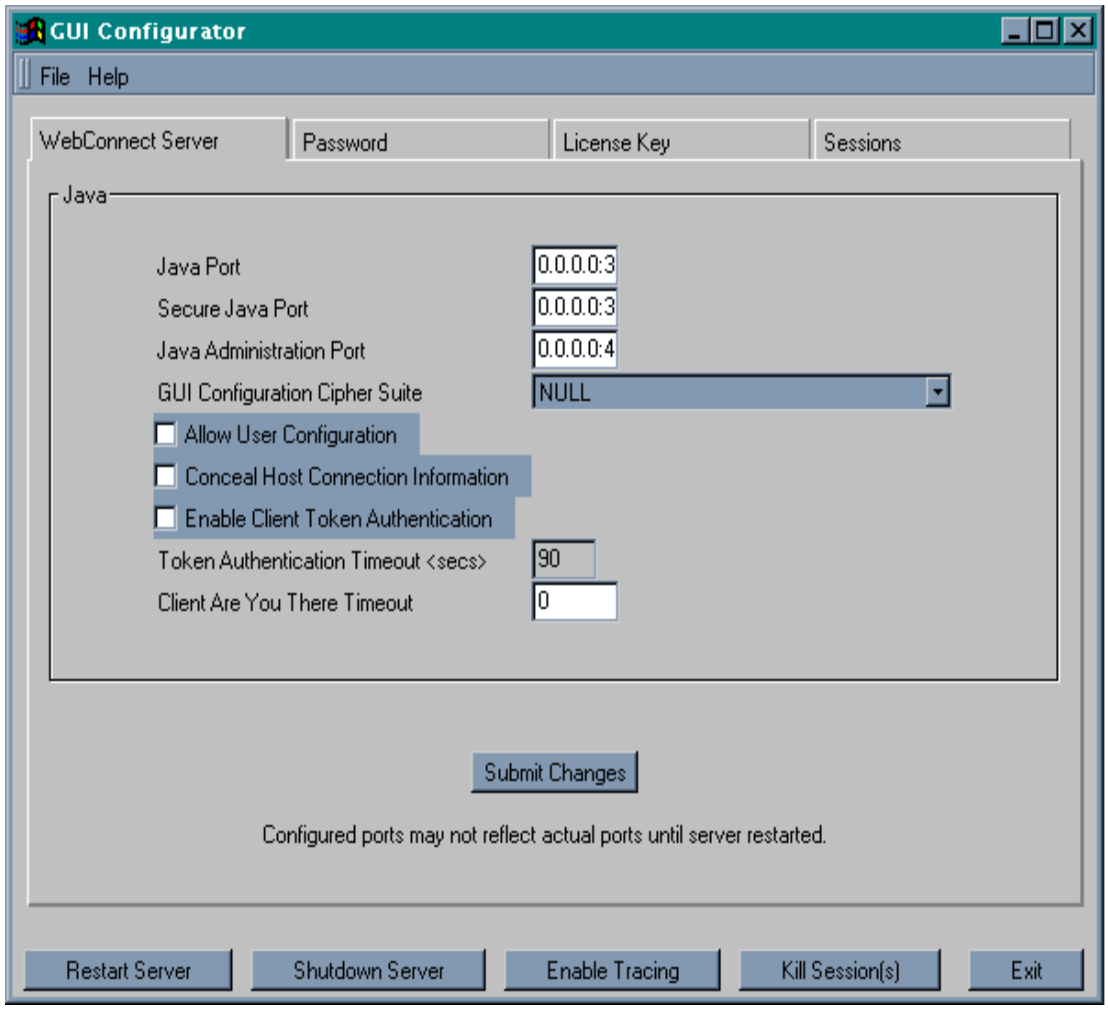
After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.
   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.
3. If the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:

   - OC://WebConnect Server
   - Password
   - License Key
   - Sessions

Using the WebConnect Server Tab

The WebConnect Server tab is available only if using Administrative mode. Use this tab to configure server ports and to enable Secure Socket Layer (SSL) protocol for the server.



Making server configuration changes

1. Make the desired change in the OC://WebConnect server configuration. Each configuration option is discussed in more detail below.
2. Save changes to the OC://WebConnect server by selecting the **Submit Changes** button. A write successful dialog will appear to indicate the changes have been made.
3. Some changes made to the OC://WebConnect server require the OC://WebConnect server be restarted for the changes to take affect. Use the **Restart Server** button to use the new settings.

   The settings that require the OC://WebConnect server to be restarted include **Java Port** IP Address and port, **Secure Java Port** IP Address and port, **Java Administration Port** IP Address and Port.

Configuring Server Ports

The OC://WebConnect Server uses three ports during operation, one for each service as shown below:

Default Settings:

IP ADDRESS AND PORT NUMBER	SERVICE	DESCRIPTION
0.0.0.0:3270	Java Server	Listening port for use by non-SSL Java emulation clients. Required
0.0.0.0:3443	Secure Java Server	Listening port for use by SSL Java emulation and administration clients. (Optional if not using SSL)
0.0.0.0:4270	Java Administration	Listening port for use by the CGIbin interface to obtain configuration parameters to launch applets and for retrieving server status information. (Optional if using static HTML and not reporting server Status information.)

Server IP Addresses and Ports

Each OC://WebConnect service can listen to incoming requests and send data back on one or all of the network interfaces installed on a UNIX platform. The configuration utility lists all of the IP addresses for the network interfaces detected The default, 0.0.0.0, will cause OC://WebConnect servers to respond to requests from all installed network interfaces. Specify another IP address to limit OC://WebConnect to one network interface.

OC://WebConnect Emulation server can use up to three ports during operation. The default port setting may be used or enter a port number greater than 0 or less than 65,535. Root privileges are required to use a port number less than 1024. A port number of 0 will disable a port.

Specify the Server IP Address and Port number in the following format:

ip address:port number

Example:

0.0.0.0:2080

For the changes to take affect restart the OC://WebConnect server. HTML files used to access OC://WebConnect will have to be updated. Use the OC://WebConnect Configure Utility to update these files or manually edit the HTML files.

Configuring GUI Configuration Cipher Suite

To use SSL for the connection between the OC://WebConnect server and the GUI Configuration applet select a cipher suite from the **GUI Configuration Cipher Suite** list box. Cipher Suites specify the algorithms to be used for authentication, data encryption, data compression, and verification of message integrity when normal session traffic begins. If NULL is selected for the cipher suite, the GUI Configuration applet will use the non-SSL port to connect to the OC://WebConnect Server.

More Information:
SSL and GUI Configuration Cipher Suites are explained in more detail in *Chapter 15 Security Overview* in the document.

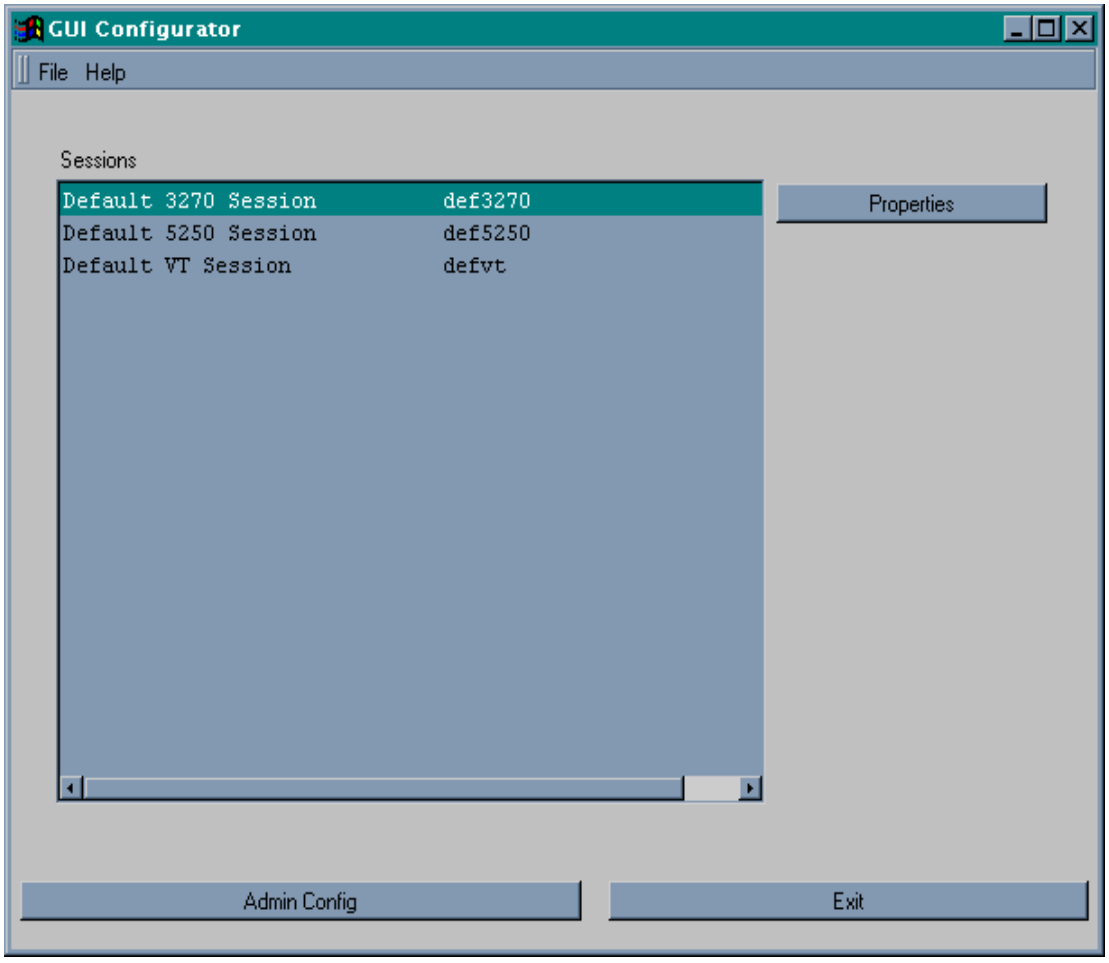
Conceal Host Connection Information

Enable **Conceal Host Connection Information** to prevent the display of the Host name, IP address, and port number of the configured and connected TN server or UNIX system within the emulation client sessions. By default this information is displayed under the Help Desk menu option on all emulation client interfaces. Submit changes.

Allow User Configuration

Enable the **Allow User Configuration** checkbox to allow “End Users” or those users of OC://WebConnect who do not have administrative access to be able to configure the Client Interface Settings for each type of emulation client interface. The “End User” will be allowed to configure their own personal keyboard map, color map, attribute map, hotspot map, and Auto GUI map for each type of emulation: 3270, 5250, and VT. These maps will be based off the default maps configured with the server but will not affect the maps on the server because the maps are stored in the /webconnect directory in the browser HOME directory of the browser system. For example a user 3270 emulation keyboard map file would be userdef3270.kbm.

User configuration files are used when a Emulation client applet is downloaded and started on the browser system. A search is made of the browser system to find any end user configuration files. If a file is found that keyboard map or color map, etc. overlays the map downloaded from the server.



Enable Client Token Authentication

Enable the **Client Token Authentication** option to use the OC://WebConnect option which uses a token to verify that the emulation client applet connecting the the OC://WebConnect server is genuine

Token Authentication Timeout <sec>

Specify, in seconds, the amount of time to wait for token authentication before a Client connection is rejected by the OC://WebConnect server.

A token is issued when an emulation session applet is initiated. The time out value is the amount of time allowed after the token is issued and before it is used for session startup. If the time out value is exceeded a host connect will not be allowed. This safeguards against a token value being stolen and used at a later time to gain host access.

More Information:

Client Token Authentication is explained in more detail in *Chapter 15 Security Overview* the document.

Client Are You There Timeout

Enable the **Client Are You There Timeout** option to prompt the server to conduct an "are you there" check to determine if a session is active. If not active, the server closes down the session between the Java client and the OC://WebConnect server. The timeout value is in minutes. Specify a **0** to disable the option.

Using the Password Tab

Use the Password tab on the OC://WebConnect **GUI Configurator** window to set the administrator password. This password is used to allow access to this GUI Configurator and HTML configuration pages.

1. Select the **Password** tab on the OC://WebConnect **GUI Configurator** window.
2. Type the new administrator password in the **Set Password** field. (The default password is "OCS."). It is recommended that the administrator password be changed from the default.

3. Type the password again in the **Confirm** field. Each character displays as an asterisk as it is entered.
4. Submit the password change by selecting the **Set Password** button.

Note:

- If the passwords in both boxes are not identical, both boxes clear when the **Set Password** button is used.

Using the License Key Tab

The OC://WebConnect License Key is used to license the OC://WebConnect server for the number of concurrent sessions and server expiration. Use the License Key tab to set the license key.

1. Select the **License** tab on the OC://WebConnect **GUI Configurator** window.
2. Enter the authorized license key in the **License Key** field.
3. Press the **Set Key** button.

Using the Sessions Tab

Use the **Sessions** tab on the OC://WebConnect **GUI Configurator** window to

- Create session configurations
- Delete session configurations
- View and modify session properties

The **Sessions** tab displays a list of defined sessions, a **Create** button, a **Delete** button, and a **Properties** button. Use the **Create** button to create a new session. The **Delete** button allows administrators to delete a defined session. The **Properties** button displays a configuration window that allows the properties and associated map files for a selected session to viewed or modified.

More Information:

For more information about Session configuration creation, deletion, or modification refer to one of the following chapters within this document:

- Chapter 6 3270 Session Configuration and Features
- Chapter 7 5250 Session Configuration and Features
- Chapter 8 3287 Print Session Configuration and Features
- Chapter 9 VT Session Configuration and Features

Using Administration Tab Buttons

Restarting Server Ports

1. Select the **WebConnect Server** tab on the **GUI Configurator** window.
2. Select the **Restart Server** button to restart OC://WebConnect's server ports and confirm restart.

Caution:

- All active Java client sessions will be killed!

Enabling Tracing

1. Select the **OC://WebConnect Server** tab on GUI Configurator window.
2. Select the **Enable Tracing** button. The **Start Trace** dialog window displays.
3. Type a **.trc* trace filename in the **Filename** box in the fourth quadrant. Do not use an extension.
4. Type a brief description or reason for the trace in the **Reason** box.
5. Use the **Start Tracing** button to begin recording communication information exchanged between the OC://WebConnect server, the host, and the Java client.
6. Trace files are stored in the OC://WebConnect directory in the **logs** directory (default is **wc/logs**).

Note:

- The **Start Tracing** button toggles to **Stop Tracing** when tracing begins. Tracing will remain on for all sessions until disabled.

Shutting Down the Server

Use the **Shutdown Server** button to shut down the OC://WebConnect server and confirm shutdown.

Caution:

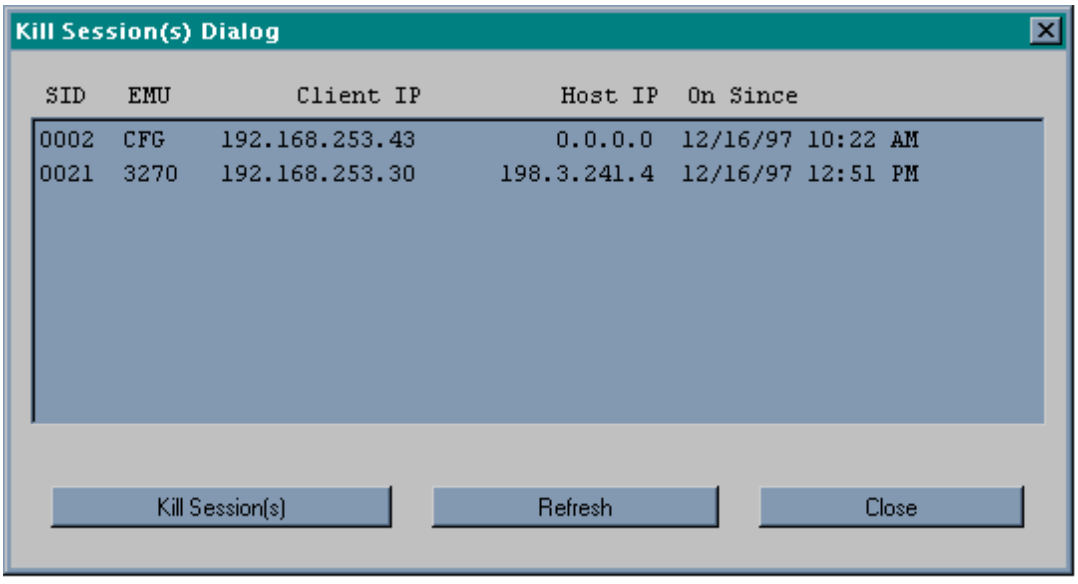
- All active clients will be killed! The OC://WebConnect Emulation server will have to be started from a UNIX command prompt or the NT services panel.

Killing a Session

A hung session or multiple sessions can be terminated at the administration level by using the **Kill Session** button on the **GUI Configurator** window. The terminated session displays with a red line at the administration and client levels.

To kill a session or multiple sessions:

1. Use the **Kill Session** button on the **GUI Configurator** window from admin mode. The **Kill Session (s) Dialog** displays.



2. Select the session or sessions to terminate.
3. Select the **Kill Session** button on the **Kill Session Dialog** window. OC://WebConnect disconnects the sessions selected.
4. The list of sessions is a snapshot of the sessions connected, the **Refresh** button updates the list of open sessions if there have been any connections or disconnects since the dialog was displayed.

Server Ports

In a TCP/IP communication, a port is a number assigned to an application program running in a destination computer. The number is used to link the incoming data to the correct application. There are many de facto standard port addresses; for example, port 80 is used for HTTP traffic (Web traffic). In the case of OC://WebConnect, there are four possible ports with only one port required:

IP ADDRESS AND PORT NUMBER	SERVICE	DESCRIPTION
0.0.0.0:3270	Java Server	Listening port for use by non-SSL Java emulation clients. Required
0.0.0.0:3443	Secure Java Server	Listening port for use by SSL Java emulation and administration clients. (Optional if not using SSL)
0.0.0.0:4270	Java Administration	Listening port for use by the CGIbin interface to obtain configuration parameters to launch applets and for retrieving server status information. (Optional if using static HTML and not reporting server Status information.)
0.0.0.0:2080	HTTP server	Serves HTML traffic for HTML configuration, Selecting sessions, and downloading client applets. (Optional if using a 3rd party HTTP server)

Chapter 6: 3270 Emulation Configuration and Features

Overview

OC://Webconnect combines feature rich 3270 emulation clients with the centralized configuration and administration on the server.

3270 emulation features include 3270E, 3278 and 3279 terminal types, Models 2 - 5, 3287 Print, IND$File transfer. Client interface features include keyboard, color, and attribute mapping , print screen, copy/paste, etc. All emulation sessions can be protected by either RSA encryption or SSL authentication and encryption.

Session configuration is provided via centralized configuration and administration tools that can be accessed remotely through a browser. OC://Webconnect provides two methods for configuration of OC://Webconnect emulation session configurations.

The two configuration tools included with OC://Webconnect provide the ability to create, delete, and modify emulation sessions configuration. An OC://Webconnect server administrator can control the important session configuration and management features like host access, security, session negotiation rules, emulation interface configuration, etc. If desired, control can be given to the end user to configure their own keyboard, color, and attribute mapping.

The first method, the **HTML Configuration**, is a series of HTML pages which accesses the OC://Webconnect Server via a CGIbin interface to create, modify and delete emulation session configurations via a browser. The HTML Configuration does not require a JAVA enabled browser. The second method, the **Graphical (GUI) Configurator**, is a java applet downloaded to the browser platform and executed via a JAVA enabled (JDK 1.1) browser.

The HTML **Configuration** and **GUI Configurator** are both full featured remote session configuration tools. Two different configuration tools are only provided for the differing needs of OC://Webconnect users.

Session Configuration Using HTML Configure

The OC://Webconnect **HTML Configuration** is a series of HTML pages that retrieve the current server and session information from the OC://Webconnect server through an administrative connection. An administrator can then modify server settings and create, modify or delete emulation session settings. The OC://Webconnect server *(See Chapter 2: Starting OC://WebConnect)* must be active to access the HTML configuration utility.

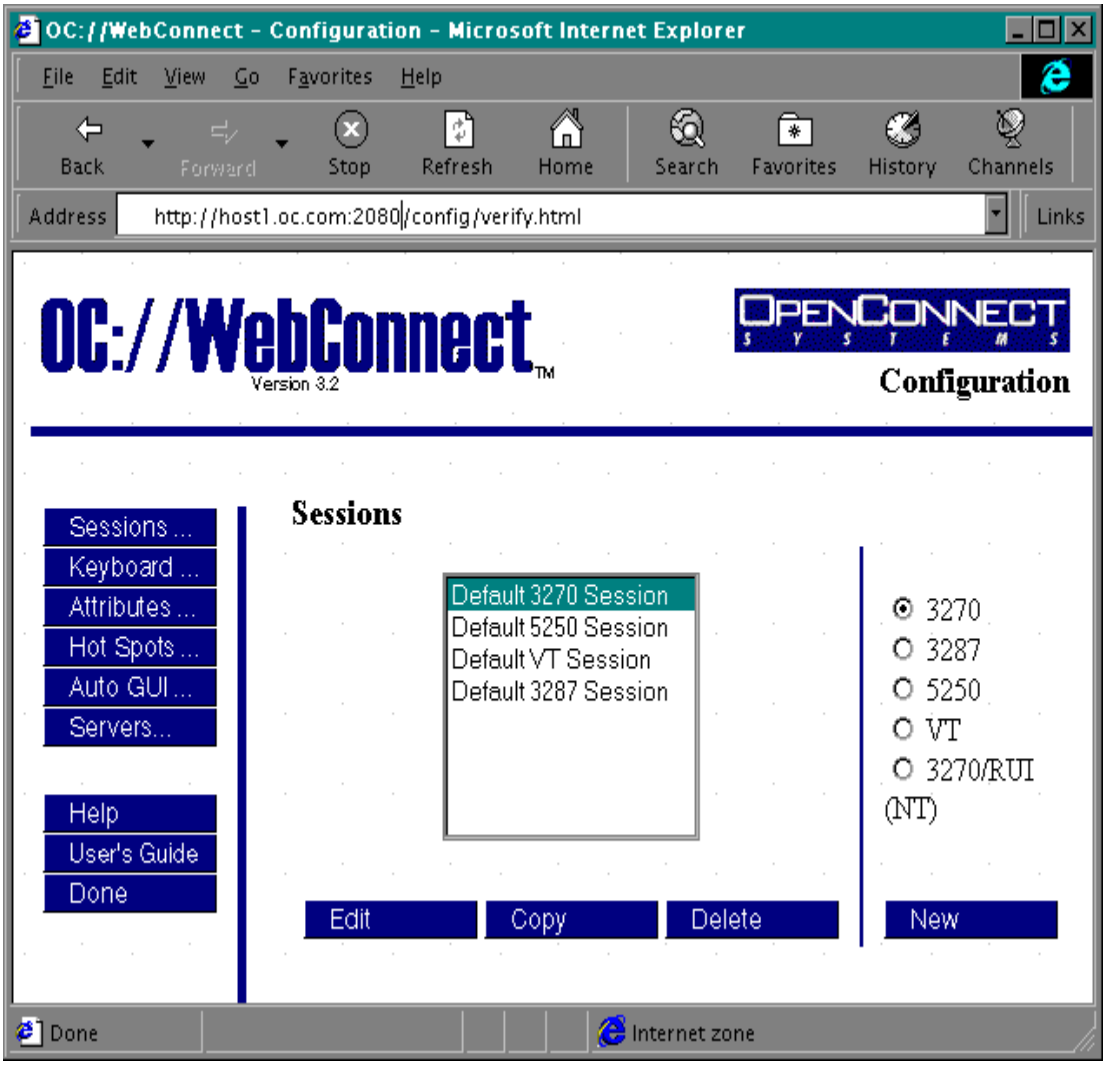
To access the 3270 HTML Session Configuration page

1. Connect to the OC://Webconnect HTTP Webserver. Enter the host name where the OC://Webconnect HTTP Webserver is running and the TCP port number in the URL of a browser.

   Example:

   Location: http://host1.oc.com:2080

2. Select the **Configure** button displayed on the left pf the **Sessions** page.
3. A prompt will appear for the **Administrator Password.**
4. Enter the appropriate password in the **Administrator Password** field and choose the **OK** button. The default password is "OCS.". Since this password is documented it is recommended that the administrator password be changed from the default.
5. If the correct Administrator password has been entered the main **Configuration** HTML page will be displayed. The main configuration page is broken up into three sections.



3270 HTML Session Configuration Page Layout

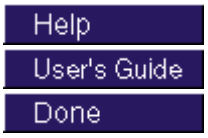
To access the **Session Configuration** page select the Configuration button from the **Sessions** page. The main **Session Configuration** page is broken up into three sections.
On the top is the ***Header*** section which shows which verson of OC://Webconnect is being accessed and show the HTML page being displayed. In this case it should display **Configuration** in the upper right corner.

On the left are **Navigation Buttons** used to access other OC://Webconnect HTML pages for context sensitive help, online User's Guide, to exit this page, and to access configuration pages for mapping other OC://Webconnect features.

To the middle and right is Session Configuration section used to create, edit, or delete session configurations.

Navigation Buttons:

- **Sessions…** To configure an individual **Session configuration.** *This is the current page being displayed.*
- **Keyboard…** To create, modify, or delete a **Keyboard map**.
- **Attributes**… To create, modify, or delete an **Attribute and Color** map.
- **Hot Spots…** To create, modify, or delete a Hot Spot map.
- **Auto GUI…** To create, modify, or delete a Auto GUI map.
- **Servers…** To modify server settings



- To access Context Sensitive Help
- To access Online User's Guide
- To exit *Session Configuration*

Edit or Create Existing Session Configurations:

A list of the existing session configurations is displayed. To edit, copy or delete an existing session select a session and choose a button.

- To edit an existing session configuration.
- To create a new session configuration from an existing session configuration.
- To delete an existing session configuration.

Create a New Session Configuration:

A group of radio buttons of the supported emulation types is displayed . Choose an emulation type to **create a new session** then select the **New** button.

- To create a new session configuration.

To create a New 3270 or 3270/RUI session configuration using HTML

1. Choose the 3270 or 3270/RUI emulation type radio button.
2. Select the New button.
3. A new session configuration page will appear with the *default* session settings. The buttons on the left are the different sections of session configuration information.
4. Enter a unique session description and filename.
5. Select each to modify the different emulation session features.
6. Each configuration option is discussed in more detail in 3270 Emulation Features section of this chapter.
7. To save new 3270 or 3270/RUI session configuration choose the **Save** button on the left side. To abort the creation of a new 3270 or 3270/RUI session configuration choose the **Cancel** button on the left hand side.

More Information:
For more information about 3270 or 3270 RUI emulation features see the 3270 features or 3270 RUI features sections in this chapter.

To Edit an existing 3270 or 3270/RUI session configuration using HTML

1. Select an existing session configuration
2. Choose the Edit button.
3. The first session configuration page will appear with the chosen session description. The buttons provide access to the different sections of session configuration information.
4. Select each to modify the different emulation session features.
5. Each configuration option is discussed in more detail in 3270 Emulation Features section of this chapter.
6. To save the changes to an existing3270 or 3270/RUI session configuration choose the **Save** button on the left side. To cancel the changes made to an existing 3270 or 3270/RUI session configuration choose the **Cancel** button on the left hand side.

More Information:
For more information about 3270 or 3270 RUI emulation features see the 3270 features or 3270 RUI features sections in this chapter.

To Copy an existing 3270 or 3270/RUI session configuration using HTML

1. Select an existing session configuration
2. Choose the **Copy** button.
3. A new session configuration page will appear with the chosen session settings.

4. Enter a unique session description and filename.
5. The buttons on the left are the different sections of session configuration information. Select each button to modify the different emulation session features.
6. Each configuration option is discussed in more detail in the 3270 features Section of this chapter.
7. To save new 3270 or 3270/RUI session configuration choose the **Save** button on the left side. To abort the creation of a new 3270 or 3270/RUI session configuration choose the **Cancel** button on the left hand side.

More Information:
For more information about 3270 or 3270 RUI emulation features see the 3270 features or 3270 RUI features sections in this chapter.

To Delete an existing 3270 or 3270/RUI session configuration using HTML

1. Select an existing session configuration
2. Choose the Delete button.
3. Confirm the session deletion

Note:
Default session configurations should not be deleted.

Using the GUI Configurator for 3270 Session Configuration

After the OC://Webconnect server (See *Chapter 2: Starting OC://WebConnect*) has been started, the GUI Configuration applet may be accessed by selecting the Administration button on any OC://Webconnect HTML page using a JDK1.1 JAVA enabled browser.

Use the **Sessions** tab on the OC://WebConnect **GUI Configurator** window to

- Create sessions
- Delete sessions
- Edit existing session configuration properties

The **Sessions** tab displays a list of existing emulation sessions configurations, a **Create** button, a **Delete** button, and a **Properties** button. Use the **Create** button to create a new session. The **Delete** button allows administrators to delete a defined session. The **Properties** button displays a

configuration window that allows properties and associated map files for a selected session to viewed or modified.

More Information:

Access to the GUI Configuration applet by removing the button from any of the of the OC://Webconnect HTML pages or custom HTML may be written to access the GUI configuration applet.

Accessing the GUI Configurator for 3270 Session Configuration

After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.

   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.

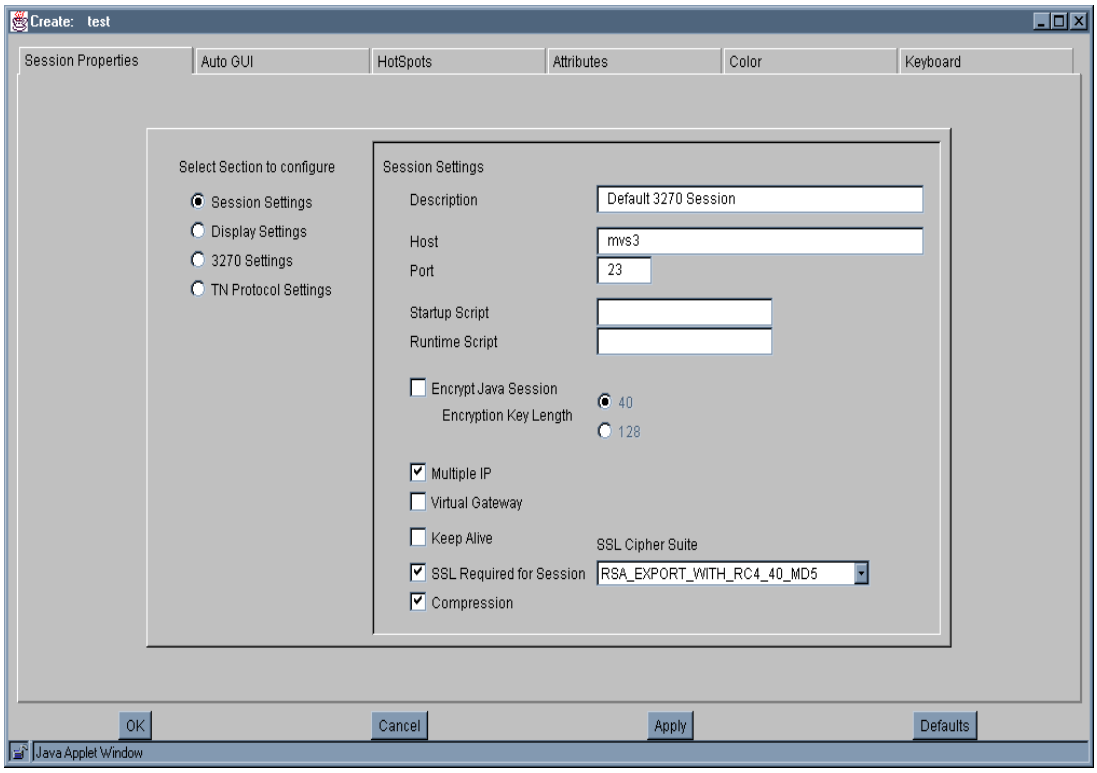
3. If the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:
   - OC://Webconnect Server
   - Password
   - License Key
   - Sessions

4. For session configuration choose the **Sessions** tab.

More Information:
For more information on the OC://Webconnect Server, Password, and License Key tabs see *Chapter 5 Server Configuration and Administration.*

To Create a New 3270 or 3270/RUI emulation session configuration

1. Choose the **Sessions** tab on the OC://WebConnect **GUI Configurator** window. A list of defined sessions displays.
2. Select the **Create** button for to create a new session configuration. The **Select Session Type** window displays.
3. Type a unique file name for the session to be created. Do not use an extension.
4. Select either the 3270 or 3270 (RUI/LUA) emulation type from the list.
5. Choose the **OK** button. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
6. Each configuration option is discussed in more detail in 3270 Emulation Features section of this chapter.
7. Choose the **OK** button to save the session configuration to a session file (**.ses*).



Notes:

- The number of tabs displayed depends on the mode the configuration applet is in and the emulation type of the session selected. For example, if the applet is in user mode, the Sessions Properties tab will not display.
- When choosing a filename for a session configuration, consider that files are listed on the **Sessions** window in alphabetical order.
- To restore the form to default display values, select the **Defaults** button.

Editing a 3270 or 3270/RUI Session Configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be editted.
3. Press the Properties button.
4. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
5. Each configuration option is discussed in more detail in 3270 Emulation Features section of this chapter.
6. Click the **OK** button to save the session configuration

To Delete a 3270 or 3270/RUI emulation session configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be deleted.
3. Press the **Delete** button.
4. A confirmation dialog is displayed, choose the **OK** button to confirm the deletion. The session file is deleted.

Note:

- Default session files cannot be deleted.

3270 Emulation session features and settings

Description…

Field	Procedure
Description	Enter a brief description for the session configuration. This description will appear on screens used to select this session to be started, modified, and deleted.
File Name	Enter a unique filename in which to store the emulation session settings.

3270 Network Settings…

Field	Procedure
Host	Enter the a host name or IP address of the gateway or TN server to be used to access the S/390 host.
Port	Enter the TCP/IP port number the gateway or TN server uses for emulation connections.
TCP Keep Alive	Enable this option to instruct OC://WebConnect to send "keepalive" messages to the host to keep the connection between the OC://Webconnect server and the gateway or telnet server alive during periods of user or host inactivity.
Multiple IP Address	Enable this option if the host name being used for host connectivity refers to multiple DNS addresses. Multiple DNS addresses are used to provide the OC://Webconnect server a choice of gateways or TN servers when a server is busy or the type of session is not available. OC://WebConnect will evaluate the DNS addresses serially to make a host connection. The methods that OC://Webconnect uses to evaluate whether the correct connection can be made depends upon the TN Server or gateway being used. If using a **OCS II gateway** evaluation can be based upon the availability of the specified Model and LU type, a specific LU name, a specific LU number, and/or access to a specific SAC LU Pool. If using a **TN3270E server** evalution can be based upon the model and LU type or a specific LU name. If using a general **TN server** evaluation is based upon model and/or LU type. **For example**: Host name XY has been configured for 2 IP addresses (gateway X and gateway Y). OC://Webconnect wants a 3279 LU, all the 3279 lus on gateway X are being

Field	Procedure
	used, so OC://Webconnect automatically will attempt to connect to a 3279 LU on gateway Y.
Virtual Gateway	Enable this option to instruct OC://WebConnect to access an OpenConnect Systems virtual gateway to look up the host gateway to access for this client location. This option requires an OCS II gateway.
Data Compression	Enable this option to compress the data flowing between the OC://Webconnect server and java client compression for data streams flowing. Data compression will reduce the amount of data flowing over the network between the OC://Webconnect server and Java clients. Be aware that the trade off for decreased network traffic flow is time compressing and uncompressing data.

3270/RUI Network Settings…

Field	Procedure
LU Pool Name	Enter the LU or LU pool name of the NT SNA server to which this session will connect.
LU Type	Check the appropriate box to indicate whether this is a 3270 or LUA LU/POOL.

Security settings…

Field	Procedure
Diffie Hellman/RC 4 Encrypt	Enable this option to encrypt session data between the OC://Webconnect server and Java client session.
Encryption Key Length	Select 40-bit encryption or 128-bit encryption. 128 bit encryption is not available outside the US. If 128 bit encryption is selected for an non-US version the session will default to 40 bit encryption. The encryption method for an specific emulation can be seen by selecting the Help Desk from the emulation client Help menu.
SSL(Secure Socket Layer)	Enable SSL to use an SSL cipher suite to provide authentication and/or encryption of data between the OC://Webconnect server and the Java client session. This option require that the OC://Webconnect Server Secure Java Port be configured and active. See *Chapter 5 OC://Webconnect Server Configuration and Administration* for more information about the **Secure Java Port**.

	Select **Optional** if the emulation client users will have the option to use SSL. Select **ALWAYS** to always force the use of SSL session configuration
SSL Cipher Suite	Select an SSL Cipher Suite. The selection of a cipher suite depends upon the level of security desired.
Limit # of Sessions per applet	Enable this option to restrict the number of **New** sessions which can be started from an emulation session that has already be connected. Each Java emulation client has a File->New menu item which allows for a new emulation session to be spawned from the existing connection. By default a emulation client user can start as many sessions as the OC://Webconnect License key will allow.
Sessions per applet	Specify the number of sessions that may be spawned from an emulation applet. Zero will also disable this option.

Display Settings

Field	Procedure
Language	Select the language of the Java client.
Attribute Map	Enter the .atm file name for the session being edited. The default is *def3270.atm*.
Keyboard Map	Enter the .kbm file name for the session being edited. The default is *def3270.kbm*.
Color Map	Enter the .clm file name for the session being edited. The default is *def3270.clm*.
Hotspot Map	Enter the .hsp file name for the session being edited. The default is *def3270.hsp*.
AutoGUI Map	Enter the .agu file name for the session being edited. The default is *def3270.agu*.
Host Code Page Number	Enter the number of the code page for the target host. Values range from 37 to 61712.
Code Page Transform Type •	Select the code page transform type from the list box. **Note**: If using the Single/Double Byte EBCDIC to Unicode option, the ability to switch the single-byte code pages using a default key is available.
Font Point Size	Enter the number indicating the font point size to use. This is the initial fonts size which dictates the initial client window size.
Display Click Pad	Enable the Display Clickpad. To initially show the Clickpad when a emulation client is started.
AutoGUI Config	Enable the AutoGUI feature for this session. This allows the emulation client user to toggle on/off the AUTO GUI display option. If this option is disabled the emulation client user will not been given the AUTO GUI option.

More Information:
For more information about mapping User interface and display options see *Chapter 10 Emulation Display Options Configuration and Features.*

3270 Settings

Field	Procedure
Device Type	Click the arrow to display the available IBM device types that can be emulated in the session.
Default Screen Size	Click the arrow to display the IBM model type that to be used for the default screen size (in the applet window for the session). **2** -- 24 rows x 80 columns **3** -- 32 rows x 80 columns **4** -- 43 rows x 80 columns **5** -- 27 rows x 132 columns
Alternate Screen Size	Click the arrow to display the IBM model type that will be used for the alternate screen size (in the applet window for the session). (see "Default Screen Size" for value descriptions.)
Monochrome	Click the checkbox to display session data in monochrome. Otherwise, select No to use color.
File Transfer Command	Enter the file transfer command to use for IND$FILE or APVUFILE transfers.
File Transfer Map Table	Select a default file transfer map file for to be used for IND$FILE or APVUFILE transfers.

TN Protocol Settings

Field	Procedure
Allow TN3270E	Check this box if the gateway or host TN server supports the enhanced TN3270 protocol, TN3270E. Enabling TN3270E will allow the use of the Associate 3287 Printer feature.
Associate TN3287 Printer	Selecting this feature will add a File menu option to the 3270 session emulation client that will allow a 3287 printer session to be started from a 3270 session. To utilize this feature, the gateway or host TN server must support the TN3270E Associate feature and must be setup with the desired display to printer association.

Field	Procedure
AutoFit	This option influences the size of the print font. If the AutoFit option is not enabled, the size of the print font will be effectively fixed such that 80 column documents will fit a portrait page setting, and 132 column documents will fit a landscape page setting. In this mode, 132 column documents will likely run off the edge of a page for portrait. Setting the AutoFit feature will cause the 3287 applet to select a font to for whatever the current page setting, ensuring that lines are never truncated.
Device Name	Type one or more LU or Pool names of the OCS gateway, separated by a space.
Allow Demotion	Click the checkbox to allow OC://WebConnect to sequentially negotiate model types below the alternate screen size. The negotiation continues until a model is selected for the session. Otherwise, select No to allow normal default and alternate screen sizes to be negotiated between OC://WebConnect and the gateway.
Client IP Passthru	This parameter specifies whether IP passthru is enabled. **On** - IP passthru is enabled and displays the negotiated IP address. **Off** - IP passthru is not enabled. **Notes**: RTM support and IP passthru require an OCSII Gateway version 3.8 or greater. Any other gateway must have IP passthru disabled. If the OCSII gateway has IP Health Check enabled, IP passthru is required, and RTM support is optional.
RTM Support	Click the checkbox to extend Response Time Monitoring (RTM) from the OC://WebConnect server to the client. **Notes**: RTM support and IP passthru require an OCSII Gateway version 3.8 or greater. Any other gateway must have IP passthru disabled. If the OCSII gateway has IP Health Check enabled, IP passthru is required, and RTM support is optional.
Are You There	Click the checkbox to instruct OC://WebConnect to send "Are You There" messages to the host. Otherwise, select No to prevent the messages from transmitting to the host.
AYT Time Out	Enter a value in whole minutes. This value represents the amount of time that OC://WebConnect waits for a return AYT response from the remote host.

TCL Script settings

Field	Procedure
Startup Script	Type the name of the TCL script that automatically runs after a 3270 emulation session has been connected.
Arguments	Specify an input argument for the specified Startup TCL script. An example of arguments would be a userid and password that the script should use to logon to a host application. Arguments should be space delimited. When using the GUI Configurator specify the arguments in the Startup script field. For more information see Chapter 13 TCL Scripting.
Runtime Script	Type the name of the script file that indicates run time for a 3270 or 5250 session. Press the **CTRL** key and type **r** to start the script.
Arguments	Specify an input argument for the specified Startup TCL script. An example of arguments would be a userid and password that the script should use to logon to a host application. Arguments should be space delimited. When using the GUI Configurator specify the arguments in the Startup script field. For more information see Chapter 13 TCL Scripting.

Print Settings

Field	Procedure
OC://Webprint	Enable this option for print screen functionality using the OC://Webprint solution. OC://Webprint must be installed on the browser platform to use this solution. This option is supported on JDK 1.0 and greater browsers.
Javascript	Enable this option for print screen using Javascript, which is embedded in some browsers. This option is supported in some JDK 1.0 and greater browsers.
JDK 1.1	Enable this option for print screen using JDK 1.1 print methods embedded in JDK 1.1 based browsers.
Disable	Enable this option to disable print screen functionality.

Troubleshooting:
If the print method chosen is not supported by the browser or has not been installed or enabled on the browser platform error messages may occur.

Chapter 7: 5250 Emulation Configuration and Features

Overview

OC://WebConnect includes 5250 emulation clients with the centralized configuration and administration on the server.

Session configuration is provided via centralized configuration and administration tools that can be accessed remotely through a browser. OC://WebConnect provides two methods for configuration of OC://WebConnect emulation session configurations.

The two configuration tools included with OC://WebConnect provide the ability to create, delete, and modify emulation sessions configuration. An OC://WebConnect server administrator can control the important session configuration and management features like host access, security, session negotiation rules, emulation interface configuration, etc. If desired, control can be given to the end user to configure their own keyboard, color, and attribute mapping.

The first method, the **HTML Configuration**, is a series of HTML pages which accesses the OC://WebConnect Server via a CGIbin interface to create, modify and delete emulation session configurations via a browser. The HTML Configuration does not require a JAVA enabled browser. The second method, the **Graphical (GUI) Configurator**, is a java applet downloaded to the browser platform and executed via a JAVA enabled (JDK 1.1) browser.

The HTML **Configuration** and **GUI Configurator** are both full featured remote session configuration tools. Two different configuration tools are only provided for the differing needs of OC://WebConnect users.

5250 Session Configuration Using HTML Configure

The OC://WebConnect **HTML Configuration** is a series of HTML pages that retrieve the current server and session information from the OC://WebConnect server through an administrative connection. An administrator can then modify server settings and create, modify or delete emulation session settings. The OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) must be active to access the HTML configuration utility.

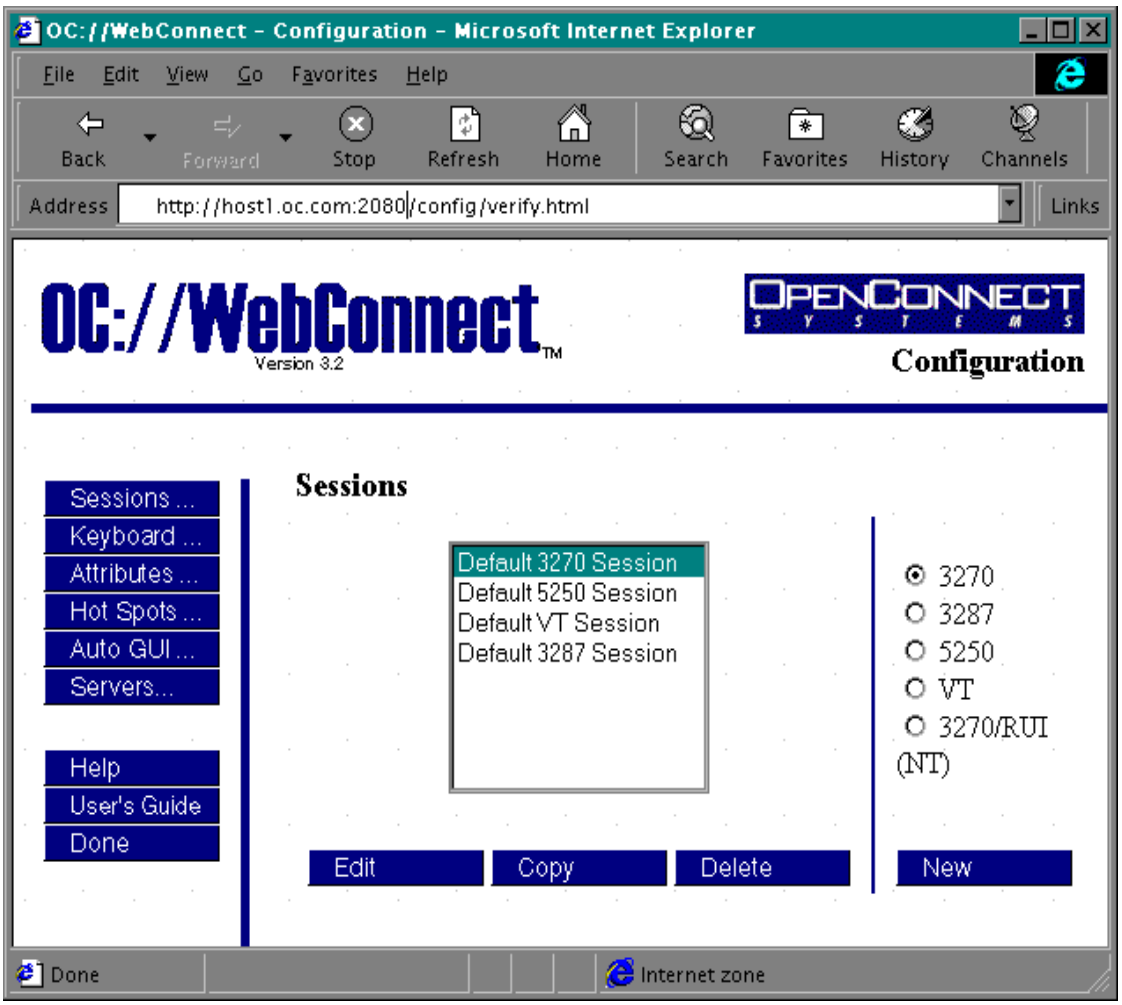
To access the 5250 HTML Session Configuration

1. Connect to the OC://WebConnect HTTP Webserver. Enter the host name where the OC://WebConnect HTTP Webserver is running and the TCP port number in the URL of a browser.

   Example:

   Location: http://host1.oc.com:2080

2. The OC://WebConnect Sessions page should be displayed.
3. Choose the **Configuration** button on the left side.
4. The OC://WebConnect administrator password verification screen will be displayed. Enter the correct admin If the correct Administrator password has been entered the main **Configuration** HTML page will be displayed.



5250 HTML Session Configuration Page Layout

To access the **Session Configuration** page select the Configuration button from the **Sessions** page. The main **Session Configuration** page is broken up into three sections.

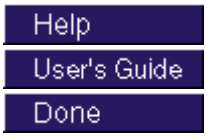
On the top is the ***Header*** section which shows which verson of OC://WebConnect is being accessed and show the HTML page being displayed. In this case it should display **Configuration** in the upper right corner.

On the left are **Navigation Buttons** used to access other OC://WebConnect HTML pages for context sensitive help, online User's Guide, to exit this page, and to access configuration pages for mapping other OC://WebConnect features.

To the middle and right is Session Configuration section used to create, edit, or delete session configurations.

Navigation Buttons:

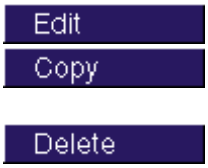
- **Sessions…** To configure an individual **Session configuration.** *This is the current page being displayed.*
- **Keyboard…** To create, modify, or delete a **Keyboard map**.
- **Attributes**… To create, modify, or delete an **Attribute and Color** map.
- **Hot Spots…** To create, modify, or delete a Hot Spot map.
- **Auto GUI…** To create, modify, or delete a Auto GUI map.
- **Servers…** To modify server settings



- To access Context Sensitive Help
- To access Online User's Guide
- To exit *Session Configuration*

Edit or Copy Existing Session Configurations:

A list of the existing session configurations is displayed. To edit, copy or delete an existing session select a session and choose a button.



- To edit an existing session configuration.
- To create a new session configuration from an existing session configuration.
- To delete an existing session configuration.

Create a New Session Configuration:

A group of radio buttons of the supported emulation types is displayed . Choose an emulation type to **create a new session** then select the **New** button.

- To create a new session configuration.

Creating a New 5250 emulation session configuration using HTML

1. Choose the 5250 emulation type.
2. Select the **New** button.
3. A new session configuration page will appear with the *default* session settings. The buttons on the left are the different sections of session configuration information.
4. Enter a unique session description and filename.
5. Select each to modify the different emulation session features.
6. Each configuration option is discussed in more detail in 5250 Emulation Features section of this chapter.
7. To save the new 5250 configuration choose the **Save** button on the left hand side. To abort the creation of a new 5250 session configuration choose the **Cancel** button on the left hand side.

To Edit an existing 5250 emulation session configuration using HTML

1. Select an existing session configuration
2. Choose the **Edit** button.
3. The first session configuration page will appear with the chosen session description. The buttons provide access to the different sections of session configuration information.
4. Select each to modify the different emulation session features.
5. Each configuration option is discussed in more detail in 5250 Emulation Features section of this chapter.
6. To save the changes made to an existing 5250 configuration choose the **Save** button on the left hand side. To cancel the changes made to an existing 5250 session configuration choose the **Cancel** button on the left hand side.

To Copy an existing 5250 emulation session configuration using HTML

1. Select an existing session configuration
2. Choose the **Copy** button.
3. A new session configuration page will appear with the chosen session settings.
4. Enter a unique session description and filename.
5. The buttons on the left are the different sections of session configuration information. Select each button to modify the different emulation session features.
6. Each configuration option is discussed in more detail in 5250 Emulation Features section of this chapter.
7. To save the new 5250 configuration choose the **Save** button on the left hand side. To abort the creation of a new 5250 session configuration choose the **Cancel** button on the left hand side.

To Delete an existing 5250 emulation session configuration using HTML

1. Select an existing session configuration
2. Choose the **Delete** button.
3. Confirm the session deletion

Note:
Default session configurations should not be deleted.

5250 Session Configuration Using the GUI Configurator

After the OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) has been started, the GUI Configuration applet may be accessed by selecting the Administration button on any OC://WebConnect HTML page using a JDK1.1 JAVA enabled browser.
Use the **Sessions** tab on the OC://WebConnect **GUI Configurator** window to

- Create sessions
- Delete sessions
- Edit existing session configuration properties

The **Sessions** tab displays a list of existing emulation sessions configurations, a **Create** button, a **Delete** button, and a **Properties** button. Use the **Create** button to create a new session. The **Delete** button allows administrators to delete a defined session. The **Properties** button displays a configuration window that allows properties and associated map files for a selected session to viewed or modified.

More Information:
Access to the GUI Configuration applet by removing the button from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet. See *Chapter 12 Customization of OC://WebConnect* in this document for more information.

Accessing the GUI Configurator for 5250 Session Configuration

After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.

   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.

3. If the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:

- OC://WebConnect Server
- Password
- License Key
- Sessions

4. For session configuration choose the **Sessions** tab.

More Information:
For more information on the OC://WebConnect Server, Password, and License Key tabs see *Chapter 5 Server Configuration and Administration*.

To Create a New 5250 emulation session configuration

1. Choose the **Sessions** tab on the OC://WebConnect **GUI Configurator** window. A list of defined sessions displays.
2. Select the **Create** button for to create a new session configuration. The **Select Session Type** window displays.
3. Type a unique file name for the session to be created. Do not use an extension.
4. Select the 5250 emulation type from the list.
5. Choose the **OK** button. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
6. Each configuration option is discussed in more detail in 5250 Emulation Features section of this chapter.
7. Choose the **OK** button to save the session configuration to a session file (**.ses*).

Notes:

- The number of tabs displayed depends on the mode the configuration applet is in and the emulation type of the session selected. For example, if the applet is in user mode, the Sessions Properties tab will not display.
- When choosing a filename for a session configuration, consider that files are listed on the **Sessions** window in alphabetical order.
- To restore the form to default display values, select the **Defaults** button.

Editing a 5250 Session Configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be editted.
3. Press the Properties button.
4. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
5. Each configuration option is discussed in more detail in 5250 Emulation Features section of this chapter.
6. Click the **OK** button to save the session configuration

To Delete a 5250 emulation session configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be deleted.
3. Press the **Delete** button.
4. A confirmation dialog is displayed, choose the **OK** button to confirm the deletion. The session file is deleted.

Note:

- Default session files cannot be deleted.

5250 Emulation Features

Description…

Field	Procedure
Description	Enter a brief description for the session configuration. This description will appear on screens used to select this session to be started, modified, and deleted.
File Name	Enter a unique filename in which to store the emulation session settings.

5250 Network Settings…

Field	Procedure
Host Name	Enter the a host name or IP address of the gateway or TN server to be used to access the AS/400 host.
Port	Enter the TCP/IP port number the gateway or TN server uses for emulation connections.
TCP/IP Keep Alive	Enable this option to instruct OC://WebConnect to send "keepalive" messages to the host to keep the connection between the OC://WebConnect server and the gateway or telnet server alive during periods of user or host inactivity.
Multiple IP Address	Enable this option if the host name being used for host connectivity refers to multiple DNS addresses. Multiple DNS addresses are used to provide the OC://WebConnect server a choice of gateways or TN servers when a server is busy or the type of session is not available. OC://WebConnect will evaluate the DNS addresses serially to make a host connection. The methods that OC://WebConnect uses to evaluate whether the correct connection can be made depends upon the TN Server or gateway being used. If using a **OCS II gateway** evaluation can be based upon the availability of the specified Model and LU type, a specific LU name, a specific LU number, and/or access to a specific SAC LU Pool. If using a general **TN server** evaluation is based upon model and/or LU type. **For example**: Host name XY has been configured for 2 IP addresses (gateway X and gateway Y). OC://WebConnect wants a 3279 LU, all the 3279 lus on gateway X are being used, so OC://WebConnect automatically will attempt to connect to a 3279 LU on gateway Y.

Data Compression	Enable this option to compress the data flowing between the OC://WebConnect server and java client compression for data streams flowing. Data compression will reduce the amount of data flowing over the network between the OC://WebConnect server and Java clients. Be aware that the trade off for decreased network traffic flow is time compressing and uncompressing data as well as an increase in the CPU utilization fo the OC://WebConnect server.

Security settings…

Field	Procedure
Diffie Hellman/RC 4 Encrypt	Enable this option to encrypt session data between the OC://WebConnect server and Java client session.
Encryption Key Length	Select 40-bit encryption or 128-bit encryption. 128 bit encryption is not available outside the US. If 128 bit encryption is selected for an non-US version the session will default to 40 bit encryption. The encryption method for an specific emulation can be seen by selecting the Help Desk from the emulation client Help menu.
SSL(Secure Socket Layer)	Enable SSL to use an SSL cipher suite to provide authentication and/or encryption of data between the OC://WebConnect server and the Java client session. This option require that the OC://WebConnect Server Secure Java Port be configured and active. See *Chapter 5 OC://WebConnect Server Configuration and Administration* for more information about the **Secure Java Port**. Select **Optional** if the emulation client users will have the option to use SSL. Select **ALWAYS** to always force the use of SSL with this session configuration
SSL Cipher Suite	Select an SSL Cipher Suite. The selection of a cipher suite depends upon the level of security desired.
Limit # of Sessions per applet	Enable this option to restrict the number of **New** sessions which can be started from an emulation session that has already been connected. Each Java emulation client has a File->New menu item which allows for a new emulation session to be spawned from the existing connection. By default a emulation client user can start as many sessions as the OC://WebConnect License key will allow.
Sessions per applet	Specify the number of sessions that may be spawned from an emulation applet. Zero sessions will disable this option and allow unlimited number of sessions to be spawned.

More Information:
For more information about OC://WebConnect security features see *Chapter 15 Security Overview.*

5250 Settings

Field	Procedure
Device Type	Click the arrow to choose the IBM device type to be emulated.
OC Server	Enable this option if an OpenConnect Systems gateway is to be used for host connections.
AS400 V2R1 Support	Enable this option if the AS/400 host is using operating system version 2.0, release 1.0 or higher
Auto Help	Enable this option to display error messages in the session's applet window.
PTS override	Enable this option to instruct OC://WebConnect to send passthru screen parameters to the 5250 host.
Remote Location Name	Enter a remote location name for the passthru screen.
Mode Name	Enter a mode name for the passthru screen.
Virtual Controller	Enter a virtual controller definition for the passthru screen.
Virtual Device	Enter the name of the AS/400 virtual device.
Remote Network ID	Type a remote network identifier for the passthru screen.
Local PU Name	Type a local PU name for the passthru screen.
Local LU Name	Type a local LU name for the passthru screen.

Display Settings…

Field	Procedure
Language	Select the language of the Java client.
Attribute Map	Enter the .atm file name for the session being edited. The default is *def5250.atm*.
Keyboard Map	Enter the .kbm file name for the session being edited. The default is *def5250.kbm*.
Color Map	Enter the .clm file name for the session being edited. The default is *def5250.clm*.
Hotspot Map	Enter the .hsp file name for the session being edited. The default is *def5250.hsp*.
AutoGUI Map	Enter the .agu file name for the session being edited. The default is *def5250.agu*.
Host Code Page Number	Enter the number of the code page for the target host. Values

Field	Procedure
	range from 37 to 61712.
Code Page Transform Type	• Select the code page transform type from the list box. **Note**: If using the Single/Double Byte EBCDIC to Unicode option, the ability to switch the single-byte code pages using a default key is available.
Font Point Size	Enter the number indicating the font point size to use. This is the initial fonts size which dictates the initial client window size.
Display Click Pad	Enable the Display Clickpad. To initially show the Clickpad when a emulation client is started.
AutoGUI Config	Enable the AutoGUI feature for this session. This allows the emulation client user to toggle on/off the AUTO GUI display option. If this option is disabled the emulation client user will not been given the AUTO GUI option.

More Information:
For more information about mapping User interface and display options see *Chapter 10 Emulation Display Options Configuration and Features.*

Print Settings

Field	Procedure
OC://Webprint	Enable this option for print screen functionality using the OC://Webprint solution. OC://Webprint must be installed on the browser platform to use this solution. This option is supported on JDK 1.0 and greater browsers.
Javascript	Enable this option for print screen using Javascript, which is embedded in some browswers. This option is supported in some JDK 1.0 and greater browsers.
JDK 1.1	Enable this option for print screen using JDK 1.1 print methods embedded in JDK 1.1 based browsers.
Disable	Enable this option to disable print screen functionality.

Troubleshooting:
If the print method chosen is not supported by the browser or has not been installed or enabled on the browser platform error messages may occur.

For more information about OC://WebConnect printing solutions see *Chapter 17 OC://WebConnect Print Solutions.*

TCL Script settings

Field	Procedure
Startup Script	Type the name of the TCL script that automatically runs after a *5250* emulation session has been connected.
Arguments	Specify an input argument for the specified Startup TCL script. An example of arguments would be a userid and password that the script should use to logon to a host application. Arguments should be space delimited.
Runtime Script	Type the name of the script file that indicates run time for a 5250 session. Press the **CTRL** key and type **r** to start the script.
Arguments	Specify an input argument for the specified Startup TCL script. An example of arguments would be a userid and password that the script should use to logon to a host application. Arguments should be space delimited.

More Information:
For more information see *Chapter 13 TCL Scripting* .

Chapter 8: 3287 Print Emulation Configuration and Features

Overview

OC://WebConnect combines feature rich 3287 Print emulation client with the centralized configuration and administration on the server.

Session configuration is provided via centralized configuration and administration tools that can be accessed remotely through a browser. OC://WebConnect provides two methods for configuration of OC://WebConnect emulation session configurations.

The two configuration tools included with OC://WebConnect provide the ability to create, delete, and modify emulation sessions configuration. An OC://WebConnect server administrator can control the important session configuration and management features like host access, security, session negotiation rules, emulation interface configuration, etc. If desired, control can be given to the end user to configure their own keyboard, color, and attribute mapping.

The first method, the **HTML Configuration**, is a series of HTML pages which accesses the OC://WebConnect Server via a CGIbin interface to create, modify and delete emulation session configurations via a browser. The HTML Configuration does not require a JAVA enabled browser. The second method, the **Graphical (GUI) Configurator**, is a java applet downloaded to the browser platform and executed via a JAVA enabled (JDK 1.1) browser.

The HTML **Configuration** and **GUI Configurator** are both full featured remote session configuration tools. Two different configuration tools are only provided for the differing needs of OC://WebConnect users.

3287 Session Configuration Using HTML Configure

The OC://WebConnect **HTML Configuration** is a series of HTML pages that retrieve the current server and session information from the OC://WebConnect server through an administrative connection. An administrator can then modify server settings and create, modify or delete emulation session settings. The OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) must be active to access the HTML configuration utility.

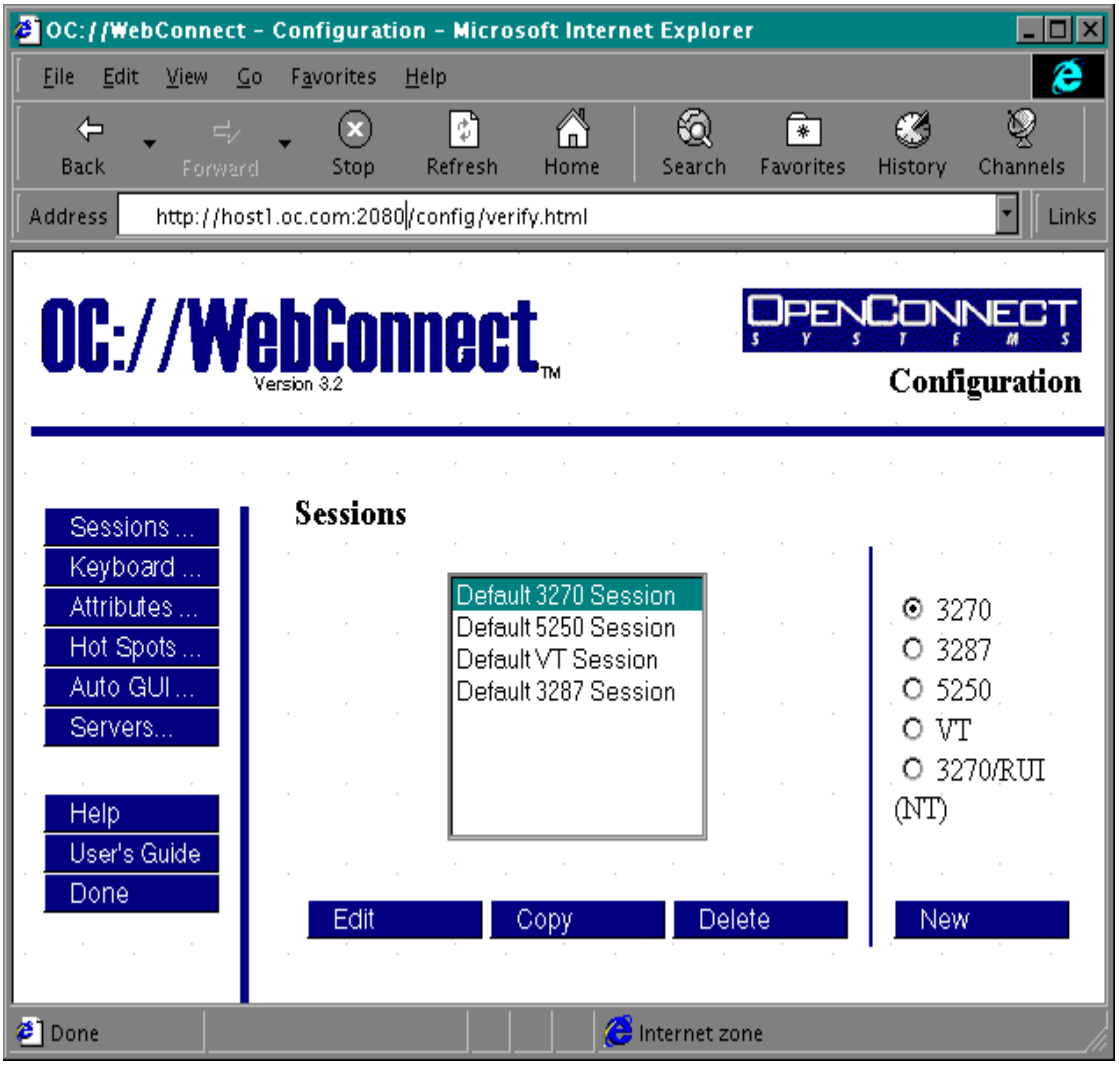
To access the 3287 HTML Session Configuration page

1. Connect to the OC://WebConnect HTTP Webserver. Enter the host name where the OC://WebConnect HTTP Webserver is running and the TCP port number in the URL of a browser.

   Example:

   Location: http://host1.oc.com:2080

3. Select the **Configure** button displayed on the left pf the **Sessions** page.
4. A prompt will appear for the **Administrator Password.**
5. Enter the appropriate password in the **Administrator Password** field and choose the **OK** button. The default password is "OCS.". Since this password is documented it is recommended that the administrator password be changed from the default.
6. If the correct Administrator password has been entered the main **Configuration** HTML page will be displayed. The main configuration page is broken up into three sections.



3287 HTML Session Configuration Page Layout

To access the **Session Configuration** page select the Configuration button from the **Sessions** page. The main **Session Configuration** page is broken up into three sections.

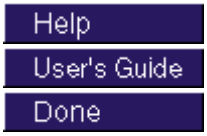
On the top is the ***Header*** section which shows which version of OC://WebConnect is being accessed and show the HTML page being displayed. In this case it should display **Configuration** in the upper right corner.

On the left are ***Navigation Buttons*** used to access other OC://WebConnect HTML pages for context sensitive help, online User's Guide, to exit this page, and to access configuration pages for mapping other OC://WebConnect features.

To the middle and right is ***Session Configuration*** section used to create, edit, or delete session configurations.

Navigation Buttons:

- **Sessions…** To configure an individual **Session configuration.** *This is the current page being displayed.*
- **Keyboard…** To create, modify, or delete a **Keyboard map**.
- **Attributes**… To create, modify, or delete an **Attribute and Color** map.
- **Hot Spots…** To create, modify, or delete a Hot Spot map.
- **Auto GUI…** To create, modify, or delete a Auto GUI map.
- **Servers…** To modify server settings



- To access Context Sensitive Help
- To access Online User's Guide
- To exit *Session Configuration*

Edit or Create Existing Session Configurations:

A list of the existing session configurations is displayed. To edit, copy or delete an existing session select a session and choose a button.

- To edit an existing session configuration.
- To create a new session configuration from an existing session configuration.
- To delete an existing session configuration.

Create a New Session Configuration:

A group of radio buttons of the supported emulation types is displayed . Choose an emulation type to **create a new session** then select the **New** button.

- To create a new session configuration.

To create a New 3287 Print session configuration using HTML

1. Choose the 3287 emulation type radio button.
2. Select the **New** button.
3. A new session configuration page will appear with the *default* session settings. The buttons on the left are the different sections of session configuration information.
4. Enter a unique session description and filename.
5. Select each to modify the different emulation session features.
6. Each configuration option is discussed in more detail in the 3287 features section of this chapter.
7. To save the new 3287 session configuration choose the **Save** button on the left side. To abort the creation of a new 3287 session configuration choose the **Cancel** button on the left hand side.

To Edit an existing 3287 Print session configuration using HTML

1. Select an existing session configuration
2. Choose the **Edit** button.
3. The first session configuration page will appear with the chosen session description. The buttons provide access to the different sections of session configuration information.
4. Select each to modify the different emulation session features.
5. Each configuration option is discussed in more detail in the 3287 features section of this chapter.
6. To save the changes to an existing 3287 session configuration choose the **Save** button on the left side. To cancel the changes made to an existing 3287 session configuration choose the **Cancel** button on the left hand side.

To Copy an existing 3287 Print session using HTML configuration

1. Select an existing session configuration
2. Choose the **Copy** button.
3. A new session configuration page will appear with the chosen session settings.
4. Enter a unique session description and filename.
5. The buttons on the left are the different sections of session configuration information. Select each button to modify the different emulation session features.
6. Each configuration option is discussed in more detail in the 3287 features section of this chapter.
7. To save the new 3287 session configuration choose the **Save** button on the left side. To abort the creation of a new 3287 session configuration choose the **Cancel** button on the left hand side.

To Delete an existing 3287 Print session configuration using HTML

1. Select an existing session configuration
2. Choose the Delete button.
3. Confirm the session deletion

Note:
Default session configurations should not be deleted.

3287 Session Configuration Using the GUI Configurator

After the OC://WebConnect server *(See Chapter 2: Starting OC://WebConnect)* has been started, the GUI Configuration applet may be accessed by selecting the Administration button on any OC://WebConnect HTML page using a JDK1.1 JAVA enabled browser.

Use the **Sessions** tab on the OC://WebConnect **GUI Configurator** window to

- Create sessions
- Delete sessions
- Edit existing session configuration properties

The **Sessions** tab displays a list of existing emulation sessions configurations, a **Create** button, a **Delete** button, and a **Properties** button. Use the **Create** button to create a new session. The **Delete** button allows administrators to delete a defined session. The **Properties** button displays a configuration window that allows properties and associated map files for a selected session to viewed or modified.

More Information:
Access to the GUI Configuration applet by removing the button from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet.

Accessing the GUI Configurator for 3287 Print Session Configuration

After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

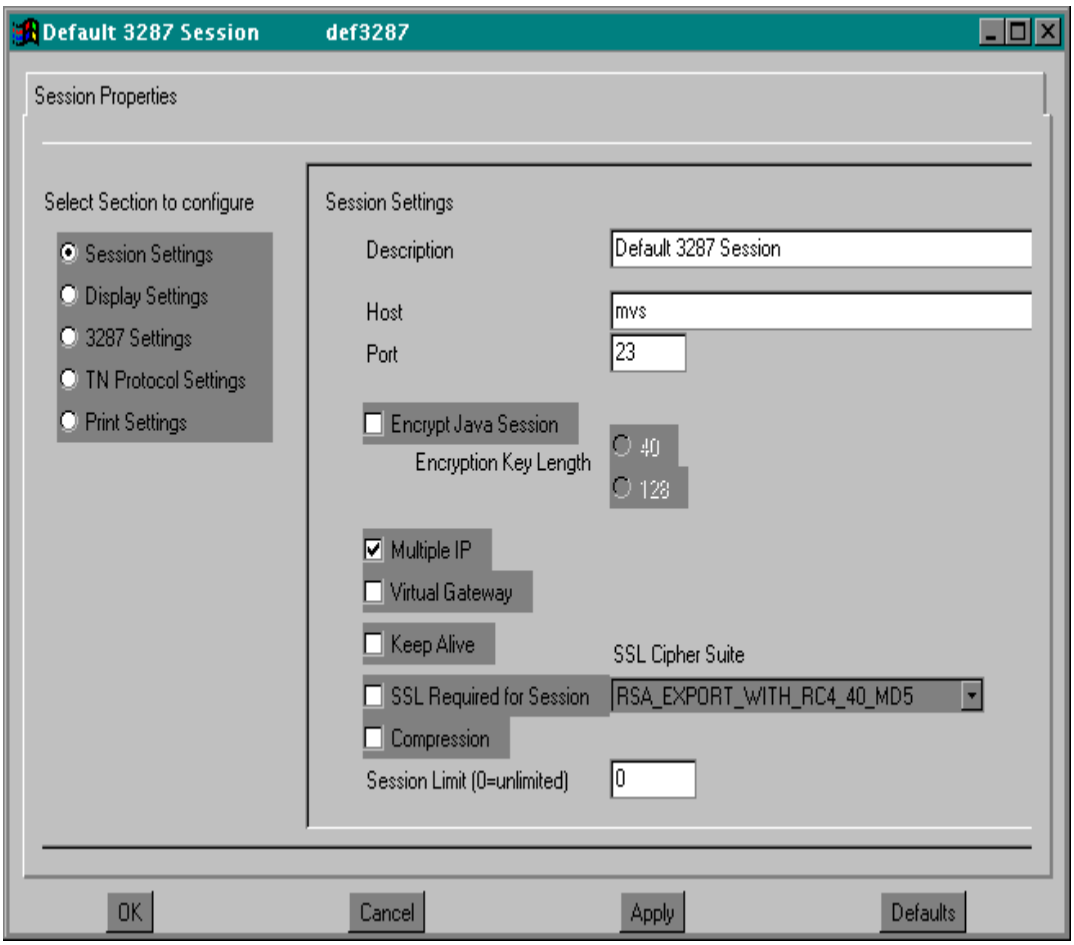
1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.

   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.
3. If the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:
   - OC://WebConnect Server
   - Password
   - License Key
   - Sessions
4. For session configuration choose the **Sessions** tab.

More Information:
For more information on the OC://WebConnect Server, Password, and License Key tabs see *Chapter 5 Server Configuration and Administration.*

To Create a New 3287 Print emulation session configuration using the GUI Configurator

1. Choose the **Sessions** tab on the OC://WebConnect **GUI Configurator** window. A list of defined sessions displays.
2. Select the **Create** button for to create a new session configuration. The **Select Session Type** window displays.
3. Type a unique file name for the session to be created. Do not use an extension.
4. Select either the 3287 Print Session emulation type from the list.
5. Choose the **OK** button. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
6. Each configuration option is discussed in more detail in the 3287 features section of this chapter.
7. Choose the **OK** button to save the session configuration to a session file (*.*ses*).



Notes:

- The number of tabs displayed depends on the mode the configuration applet is in and the emulation type of the session selected. For example, if the applet is in user mode, the Sessions Properties tab will not display.
- When choosing a filename for a session configuration, consider that files are listed on the **Sessions** window in alphabetical order.
- To restore the form to default display values, select the **Defaults** button.

Editing a 3287 Print Session Configuration using the GUI Configurator

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be editted.
3. Press the Properties button.
4. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
5. Each configuration option is discussed in more detail in the 3287 features section of this chapter.
6. Click the **OK** button to save the session configuration

To Delete a 3287 Print emulation session configuration using the GUI Configurator

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be deleted.
3. Press the **Delete** button.
4. A confirmation dialog is displayed, choose the **OK** button to confirm the deletion. The session file is deleted.

Note:

- Default session files cannot be deleted.

3287 Print Features

On the **Session Properties** tab, click the radio button to select the section of the default configuration file to modify. The properties for the section of the default configuration file selected displays in the box to the right of the radio buttons. These properties correspond to some properties listed in sections of the default session file (*def3287.ses*).

Enter the appropriate settings for a 3287 session:

Description…

Field	Procedure
Description	Enter a brief description for the session configuration. This description will appear on screens used to select this session to be started, modified, and deleted.
File Name	Enter a unique filename in which to store the emulation session settings. Keep in mind that the filename will control the order in which sessions appear on select session list boxes.

Network Settings…

Field	Procedure
Host	Enter the a host name or IP address of the gateway or TN server to be used to access the S/390 host.
Port	Enter the TCP/IP port number the gateway or TN server uses for emulation connections.
TCP Keep Alive	Enable this option to instruct OC://WebConnect to send "keepalive" messages to the host to keep the connection between the OC://WebConnect server and the gateway or telnet server alive during periods of user or host inactivity.
Multiple IP Address	Enable this option if the host name being used for host connectivity refers to multiple DNS addresses. Multiple DNS addresses are used to provide the OC://WebConnect server a choice of gateways or TN servers when a server is busy or the type of session is not available. OC://WebConnect will evaluate all DNS addresses to make a host connection. This option requires an OCS II gateway. **For example**: Host name XY has been configured for 2 IP addresses (gateway X and gateway Y). OC://WebConnect wants a 3279 LU, all the 3279 lus on gateway X are being used, so OC://WebConnect automatically will attempt to connect to a 3279 LU on gateway Y.
Virtual Gateway	Enable this option to instruct OC://WebConnect to access an OpenConnect Systems virtual gateway to look up the host gateway to access for this client location.
Data Compression	Enable this option to compress the data flowing between the

Field	Procedure
	OC://WebConnect server and java client compression for data streams flowing. Data compression will reduce the amount of data flowing over the network between the OC://WebConnect server and Java clients. Be aware that the trade off for decreased network traffic flow is time compressing and uncompressing data.

Security settings…

Diffie Hellman/RC 4 Encrypt	Enable this option to encrypt session data between the OC://WebConnect server and Java client session.
Encryption Key Length	Select 40-bit encryption or 128-bit encryption. 128 bit encryption is not available outside the US. If 128 bit encryption is selected for an non-US version the session will default to 40 bit encryption. The encryption method for an specific emulation can be seen by selecting the Help Desk from the emulation client Help menu.
SSL(Secure Socket Layer)	Enable SSL to use an SSL cipher suite to provide authentication and/or encryption of data between the OC://WebConnect server and the Java client session. This option require that the OC://WebConnect Server Secure Java Port be configured and active. Select **Optional** if the emulation client users will have the option to use SSL. Select **ALWAYS** to always force the use of SSL session configuration
SSL Cipher Suite	Select an SSL Cipher Suite. The selection of a cipher suite depends upon the level of security desired.
Limit # of Sessions per applet	Enable this option to restrict the number of **New** sessions which can be started from an emulation session that has already be connected. Each Java emulation client has a File->New menu item which allows for a new emulation session to be spawned from the existing connection. By default a emulation client user can start as many sessions as the OC://WebConnect License key will allow.
Sessions per applet	Specify the number of sessions that may be spawned from an emulation applet. Zero will also disable this option.

More Information:
For more information about configuring the OC://WebConnect server for SSL and the Secure Java Port see *Chapter 5 Server Configuration and Administration*.

Telnet Protocol Settings

Field	Procedure
Enable TN3270E	Check this box if the gateway or host TN server supports the enhanced TN3270 protocol, TN3270E.
Client IP Passthru	This parameter specifies whether IP passthru is enabled. **On** - IP passthru is enabled and displays the negotiated IP address. **Off** - IP passthru is not enabled. **Notes**: RTM support and IP passthru require an OCSII Gateway version 3.8 or greater. Any other gateway must have IP passthru disabled. If the OCSII gateway has IP Health Check enabled, IP passthru is required, and RTM support is optional.
RTM Support	Click the checkbox to extend Response Time Monitoring (RTM) from the OC://WebConnect server to the client. **Notes**: RTM support and IP passthru require an OCSII Gateway version 3.8 or greater. Any other gateway must have IP passthru disabled. If the OCSII gateway has IP Health Check enabled, IP passthru is required, and RTM support is optional.
Telnet Are You There (AYT)	Click the checkbox to instruct OC://WebConnect to send "Are You There" messages to the host. Otherwise, select No to prevent the messages from transmitting to the host.
AYT Time Out	Enter a value in whole minutes. This value represents the amount of time that OC://WebConnect waits for a return AYT response from the remote host.
Device Name	Type one or more LU or Pool names of the OCS gateway, separated by a space.

3287 Settings

Field	Procedure
AutoFit	This option allows the print client to compute a font size to match the print job to the page size. If the AutoFit option is not enabled, the size of the print font will be effectively fixed such that 80 column documents will fit a portrait page setting, and 132 column documents will fit a landscape page setting. In this mode, 132 column documents will likely run off the edge of a page for portrait. Setting the AutoFit feature will cause the 3287 applet to select a font to for whatever the current page setting, ensuring that lines are never truncated.

Display Settings

Field	Procedure
Language	Select the language of the Java client.
Host Code Page	Enter the number of the code page for the target host. Values range from

Number	37 to 61712.
Code Page Transform Type	Select the code page transform type from the list box. **Note**: If using the Single/Double Byte EBCDIC to Unicode option, the ability to switch the single-byte code pages using a default key is available.

Print Settings

Field	Procedure
OC://Webprint	Enable this option for print screen functionality using the OC://Webprint solution. OC://Webprint must be installed on the browser platform to use this solution. This option is supported on JDK 1.0 and greater browsers.
Javascript	Enable this option for print screen using Javascript, which is embedded in some browsers. This option is supported in some JDK 1.0 and greater browsers.
JDK 1.1	Enable this option for print screen using JDK 1.1 print methods embedded in JDK 1.1 based browsers.
Disable	Enable this option to disable print screen functionality.

Troubleshooting:

If the print method chosen is not supported by the browser or has not been installed or enabled on the browser platform error messages may occur.

Chapter 9: VT Emulation Configuration and Features

Overview

OC://WebConnect combines feature VT Java emulation clients with the centralized configuration and administration on the server.

Client interface features include keyboard, color, and attribute mapping, print screen, copy/paste, etc. All emulation sessions can be protected by either RSA encryption or SSL authentication and encryption.

Session configuration is provided via centralized configuration and administration tools that can be accessed remotely through a browser. OC://WebConnect provides two methods for configuration of OC://WebConnect emulation session configurations.

The two configuration tools included with OC://WebConnect provide the ability to create, delete, and modify emulation sessions configuration. An OC://WebConnect server administrator can control the important session configuration and management features like host access, security, session negotiation rules, emulation interface configuration, etc. If desired, control can be given to the end user to configure their own keyboard, color, and attribute mapping.

The first method, the **HTML Configuration**, is a series of HTML pages which accesses the OC://WebConnect Server via a CGIbin interface to create, modify and delete emulation session configurations via a browser. The HTML Configuration does not require a JAVA enabled browser.

The second method, the **Graphical (GUI) Configurator**, is a java applet downloaded to the browser platform and executed via a JAVA enabled (JDK 1.1) browser.
The HTML **Configuration** and **GUI Configurator** are both full featured remote session configuration tools. Two different configuration tools are only provided for the differing needs of OC://WebConnect users.

Session Configuration Using HTML Configure

The OC://WebConnect **HTML Configuration** is a series of HTML pages that retrieve the current server and session information from the OC://WebConnect server through an administrative connection. An administrator can then modify server settings and create, modify or delete emulation session settings. The OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) must be active to access the HTML configuration utility.

To access the VT HTML Session Configuration page

1. Connect to the OC://WebConnect HTTP Webserver. Enter the host name where the OC://WebConnect HTTP Webserver is running and the TCP port number in the URL of a browser.

   Example:

   Location: http://host1.oc.com:2080

3. Select the **Configure** button displayed on the left of the **Sessions** page.
4. A prompt will appear for the **Administrator Password.**
5. Enter the appropriate password in the **Administrator Password** field and choose the **OK** button. The default password is "OCS". Since this password is documented it is recommended that the administrator password be changed from the default.
6. If the correct Administrator password has been entered the main **Configuration** HTML page will be displayed. The main configuration page is broken up into three sections.

VT HTML Session Configuration Page Layout

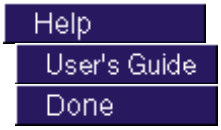
To access the **Session Configuration** page select the Configuration button from the **Sessions** page. The main **Session Configuration** page is broken up into three sections.

On the top is the ***Header*** section which shows which verson of OC://WebConnect is being accessed and show the HTML page being displayed. In this case it should display **Configuration** in the upper right corner.

On the left are **Navigation Buttons** used to access other OC://WebConnect HTML pages for context sensitive help, online User's Guide, to exit this page, and to access configuration pages for mapping other OC://WebConnect features.
To the middle and right is Session Configuration section used to create, edit, or delete session configurations.

Navigation Buttons:

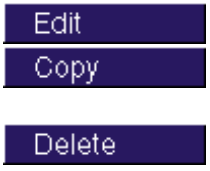
- **Sessions…** To configure an individual **Session configuration.** *This is the current page being displayed.*
- **Keyboard…** To create, modify, or delete a **Keyboard map**.
- **Attributes**… To create, modify, or delete an **Attribute and Color** map.
- **Hot Spots…** To create, modify, or delete a Hot Spot map.
- **Auto GUI…** To create, modify, or delete a Auto GUI map.
- **Servers…** To modify server settings



- To access **Context Sensitive Help**
- To access **Online User's Guide**
- To **exit** *Session Configuration*

Edit or Create Existing Session Configurations:

A list of the existing session configurations is displayed. To edit, copy or delete an existing session select a session and choose a button.



- To edit an existing session configuration.
- To create a new session configuration from an existing session configuration.
- To delete an existing session configuration.

Create a New Session Configuration:

A group of radio buttons of the supported emulation types is displayed . Choose an emulation type to **create a new session** then select the **New** button.

- To create a new session configuration.

Creating a New VT emulation session configuration using HTML

1. Choose the VT emulation type.
2. Select the New button.
3. A new session configuration page will appear with the *default* session settings. The buttons on the left are the different sections of session configuration information.
4. Enter a unique session description and filename.
5. Select each to modify the different emulation session features.
6. Each configuration option is discussed in more detail in VT Features section of this chapter.
7. To save the new VT session configuration choose the **Save** button on the left side. To abort the creation of a new VT session configuration, choose the **Cancel** button on the left hand side.

To Edit an existing VT emulation session configuration using HTML

1. Select an existing session configuration.
2. Choose the Edit button.
3. The first session configuration page will appear with the chosen session description. The buttons provide access to the different sections of session configuration information.
4. Select each to modify the different emulation session features.
5. Each configuration option is discussed in more detail in VT Features section of this chapter.
6. To save changes made to the existing VT session configuration choose the **Save** button on the left side. To cancel the changes made to the existing VT session configuration, choose the **Cancel** button on the left hand side.

To Copy an existing VT emulation session configuration using HTML

1. Select an existing session configuration.
2. Choose the **Copy** button.
3. A new session configuration page will appear with the chosen session settings.
4. Enter a unique session description and filename.
5. The buttons on the left are the different sections of session configuration information. Select each button to modify the different emulation session features.
6. Each configuration option is discussed in more detail in VT Features section of this chapter.
7. To save the new VT session configuration, choose the **Save** button on the left side. To abort the creation of a new VT session configuration, choose the **Cancel** button on the left hand side.

To Delete an existing VT emulation session configuration

1. Select an existing session configuration.
2. Choose the Delete button.
3. Confirm the session deletion.

Note:
Default session configurations should not be deleted.

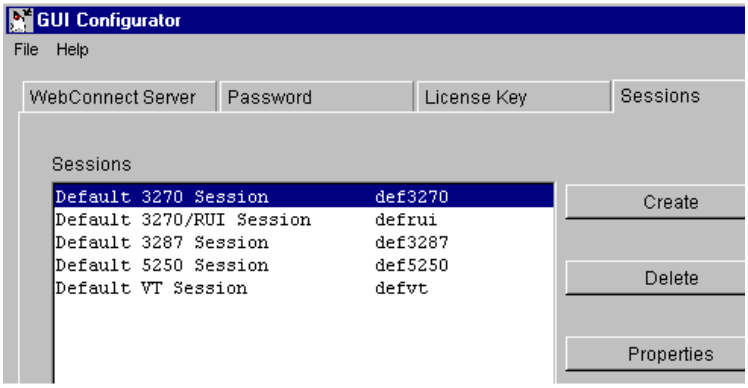
Session Configuration Using the GUI Configurator

After the OC://WebConnect server (See *Chapter 2: Starting OC://WebConnect*) has been started, the GUI Configuration applet may be accessed by selecting the Administration button on any OC://WebConnect HTML page using a JDK1.1 JAVA enabled browser.

Use the **Sessions** tab on the OC://WebConnect **GUI Configurator** window to

- Create sessions
- Delete sessions
- Edit existing session configuration properties

The **Sessions** tab displays a list of existing emulation session configurations, a **Create** button, a **Delete** button, and a **Properties** button. Use the **Create** button to create a new session. The **Delete** button allows administrators to delete a defined session. The **Properties** button displays a configuration window that allows properties and associated map files for a selected session to viewed or modified.



More Information:

Access to the GUI Configuration applet by removing the button from any of the of the OC://WebConnect HTML pages or custom HTML may be written to access the GUI configuration applet. See *Chapter 12 Customization of OC://WebConnect* in this document for more information.

Accessing and Using the GUI Configurator for VT Session Configuration

After the GUI Configuration applet has been downloaded to the browser the **Configuration Permissions Dialog** window displays asking for the Administrator password.

1. Enter the appropriate password in the **Administrator Password** field.
2. Choose the **OK** button. The OC://WebConnect **GUI Configurator** window displays in Administrative mode.

   The **Cancel** button will return the focus to the main OC://WebConnect HTML page.
3. I f the correct Administrator password has been entered the GUI Configuration applet will appear with the current configuration information including four tabs:

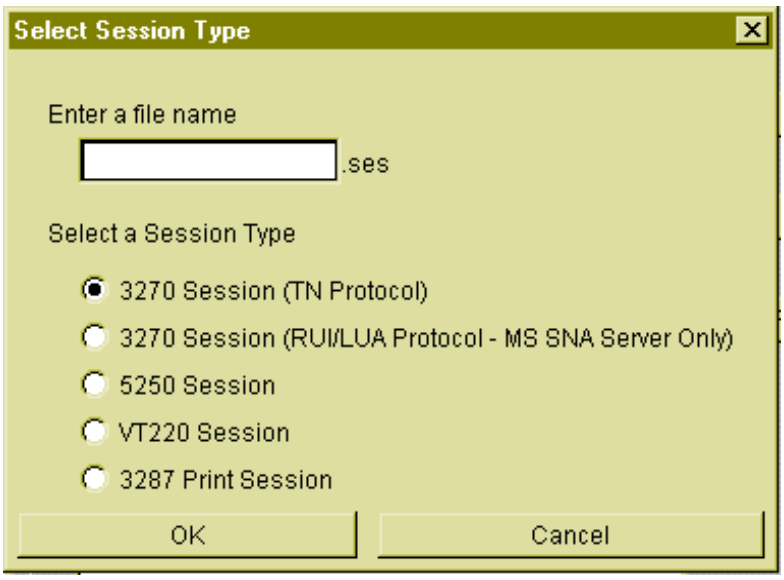
   - OC://WebConnect Server
   - Password
   - License Key
   - Sessions

4. For session configuration choose the **Sessions** tab.

More Information:
For more information on the OC://WebConnect Server, Password, and License Key tabs see *Chapter 5 Server Configuration and Administration*.

To Create a New VT emulation session configuration

1. Choose the **Sessions** tab on the OC://WebConnect **GUI Configurator** window. A list of defined sessions displays.
2. Select the **Create** button for to create a new session configuration. The **Select Session Type** window displays.



3. Type a unique file name for the session to be created. Do not use an extension.
4. Select the VT emulation type from the list.
5. Choose the **OK** button. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
6. Each configuration option is discussed in more detail in VT Features section of this chapter.
7. Choose the **OK** button to save the session configuration to a session file (**.ses*).

Notes:

- The number of tabs displayed depends on the mode the configuration applet is in and the emulation type of the session selected. For example, if the applet is in user mode, the Sessions Properties tab will not display.
- When choosing a filename for a session configuration, consider that files are listed on the **Sessions** window in alphabetical order.
- To restore the form to default display values, select the **Defaults** button.

Editing a VT Session Configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window.
2. Select the session configuration to be edited.
3. Press the Properties button.
4. The **GUI Configurator** window displays the **Session Properties** tab for the selected emulation type.
5. Each configuration option is discussed in more detail in VT Features section of this chapter.
6. Click the **OK** button to save the session configuration.

To Delete a VT emulation session configuration

1. Choose the **Session** tab on the OC://WebConnect **GUI Configurator** window .
2. Select the session configuration to be deleted.
3. Press the **Delete** button.
4. A confirmation dialog is displayed, choose the **OK** button to confirm the deletion. The session file is deleted.

Note:

Default session files cannot be deleted.

VT Emulation Features

Description…

Field	Procedure
Description	Enter a brief description for the session configuration. This description will appear on screens used to select this session to be started, modified, and deleted.
File Name	Enter a unique filename in which to store the emulation session settings. Keep in mind that the filename will control the order in which sessions appear on select session list boxes.

Network Settings…

Field	Procedure
Host	Enter the a host name or IP address of the gateway or TN server to be used to access the S/390 host.
Port	Enter the TCP/IP port number the gateway or TN server uses for emulation connections.
TCP Keep Alive	Enable this option to instruct OC://WebConnect to send "keepalive" messages to the host to keep the connection between the OC://WebConnect server and the gateway or telnet server alive during periods of user or host inactivity.
Multiple IP Address	Enable this option if the host name being used for host connectivity refers to multiple DNS addresses. Multiple DNS addresses are used to provide the OC://WebConnect server a choice of gateways or TN servers when a server is busy or the type of session is not available. OC://WebConnect will evaluate all DNS addresses to make a host connection. **For example**: Host name XY has been configured for 2 IP addresses (gateway X and gateway Y). OC://WebConnect wants a 3279 LU, all the 3279 lus on gateway X are being used, so OC://WebConnect automatically will attempt to connect to a 3279 LU on gateway Y.
Data Compression	Enable this option to compress the data flowing between the OC://WebConnect server and java client compression for data streams flowing. Data compression will reduce the amount of data flowing over the network between the OC://WebConnect server and Java clients. Be aware

Field	Procedure
	that the trade off for decreased network traffic flow is time compressing and uncompressing data.

Security settings…

Field	Procedure
Diffie Hellman/RC 4 Encrypt	Enable this option to encrypt session data between the OC://WebConnect server and Java client session.
Encryption Key Length	Select 40-bit encryption or 128-bit encryption. 128 bit encryption is not available outside the US. If 128 bit encryption is selected for an non-US version the session will default to 40 bit encryption. The encryption method for an specific emulation can be seen by selecting the Help Desk from the emulation client Help menu.
SSL(Secure Socket Layer)	Enable SSL to use an SSL cipher suite to provide authentication and/or encryption of data between the OC://WebConnect server and the Java client session. This option require that the OC://WebConnect Server Secure Java Port be configured and active. Select **Optional** if the emulation client users will have the option to use SSL. Select **ALWAYS** to always force the use of SSL session configuration
SSL Cipher Suite	Select an SSL Cipher Suite. The selection of a cipher suite depends upon the level of security desired.
Limit # of Sessions per applet	Enable this option to restrict the number of **New** sessions which can be started from an emulation session that has already be connected. Each Java emulation client has a File->New menu item which allows for a new emulation session to be spawned from the existing connection. By default a emulation client user can start as many sessions as the OC://WebConnect License key will allow.
Sessions per applet	Specify the number of sessions that may be spawned from an emulation applet. Zero will also disable this option.

More Information:
For more information about configuring the OC://WebConnect Server for security and the secure Java port see *Chapter 5 Server Configuration and Administration*.

Display Settings…

Field	Procedure
Language	Select the language of the Java client.
Attribute Map	Enter the .atm file name for the session being edited. The default is *defvt.atm*.
Keyboard Map	Enter the .kbm file name for the session being edited. The default is *defvt.kbm*.
Color Map	Enter the .clm file name for the session being edited. The default is *defvt.clm*.
Host Code Page Number	Enter the number of the code page for the target host. Values range from 37 to 61712.
Code Page Transform Type	Select the code page transform type from the list box. **Note**: If using the Single/Double Byte EBCDIC to Unicode option, the ability to switch the single-byte code pages using a default key is available.
Font Point Size	Enter the number indicating the font point size to use. This is the initial fonts size which dictates the initial client window size.
Display Click Pad	Enable the Display Clickpad. To initially show the Clickpad when a emulation client is started.

More Information:
For more information about mapping User interface and display options see *Chapter 10 Display Options Configuration and Features*.

VT Settings

Field	Procedure
Number of Columns	Select the number of columns from the list to set the session display width.
Number of Rows(Lines)	Select the number of rows to be displayed from the list.
Tab Spacing	Enter the tab settings. Example: Tabs = T T T T T T T T
Auto Wrap	Click the checkbox to wrap text automatically.
Warning Bell	Click the checkbox to activate the margin bell. The warning bell sounds when the cursor is moved within eight characters of the maximum number of columns set.

Print Settings

Field	Procedure
OC://Webprint	Enable this option for print screen functionality using the OC://Webprint solution. OC://Webprint must be installed on the browser platform to use this solution. This option is supported on JDK 1.0 and greater browsers.
Javascript	Enable this option for print screen using Javascript, which is embedded in some browsers. This option is supported in some JDK 1.0 and greater browsers.
JDK 1.1	Enable this option for print screen using JDK 1.1 print methods embedded in JDK 1.1 based browsers.
Disable	Enable this option to disable print screen functionality.

Troubleshooting:
If the print method chosen is not supported by the browser or has not been installed or enabled on the browser platform error messages may occur. For more information about OC://WebConnect printing solutions see *Chapter 17 OC://WebConnect Print Solutions.*

Chapter 10: Display Options Configuration and Features

Accessing Display Options

This section discusses the display options that can be modified by an administrator or by the user if the administrator has enabled "Allow User Configuration" from the WebConnect tab in the GUI configuration.

You can log on to OC://WebConnect without administrative privileges in user mode and can change the display options locally on the browser machine. However, as an administrator, modifying display options changes server files. User mode displays a list of defined sessions and a **Properties** button. The **Properties** button displays a configuration window that allows you to view and modify the key, color, attribute, hotspot, and auto GUI map files associated with the selected session. The properties of the session are not available when the configuration applet is in user mode.

Switching from User to Admin Mode

1. Log on to OC://WebConnect.
2. Click the Admin Config button. The Configuration Permissions Dialog window displays.
3. Enter the appropriate password in the **Administrator Password** field.
4. Click the **OK** button. The OC://WebConnect configuration applet displays in admin mode.

Note:

You can set display options in admin mode by clicking the **Properties** button on the **Sessions** tab on the **configuration applet** window.

Using the Auto GUI Tab

The Auto GUI tab creates an automatic graphical-user interface (GUI) to replace host screens. When you enable the Auto GUI , your host's "green-on-black" screens are replaced with a GUI containing labels and text fields.

Notes:

- Auto GUI can be configured only by the administrator in Admin Mode.
- Screen sizes vary and may blink when using labels and text fields because their components occupy different amounts of space.
- Using Java, text fields do not accept colors the same as labels. Therefore, certain colors are not available for text fields.

Setting GUI Options

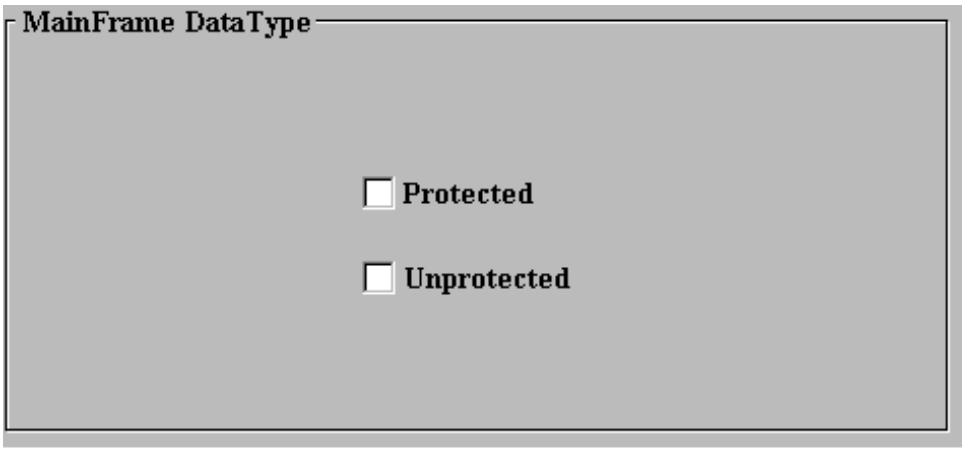
To set GUI options

1. Click the **Auto GUI** tab in the **OC://WebConnect configuration applet** window.

Notes:

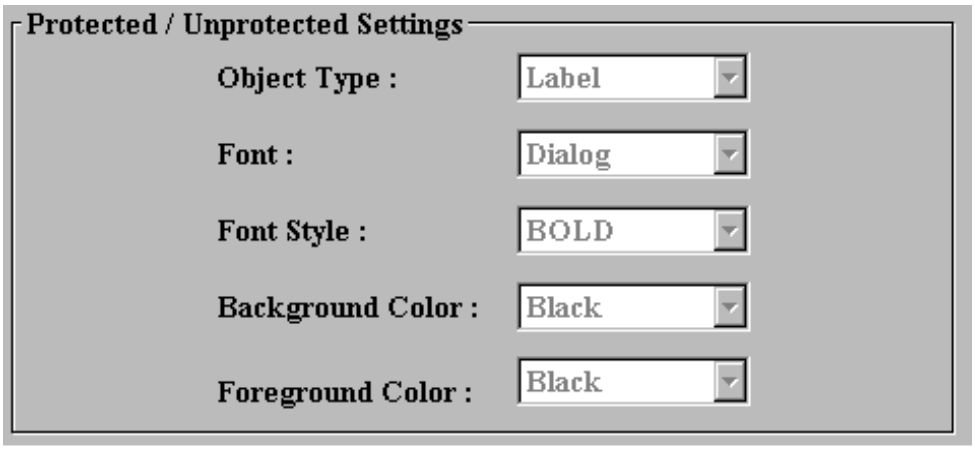
- Auto GUI is available for 3270 and 5250 sessions only.

2. Click the **Protected** or **Unprotected** checkbox in the **MainFrame Data Type** box. Click **Protected** to assign options to host fields that cannot be changed or edited. Click **Unprotected** to assign options to host fields to which you can input data or change.



Note:

The **Object Type** field in the **Protected/Unprotected Settings** box automatically displays **Label** if you choose **Protected** in the **Mainframe Data Type** box. The **Object Type** field in the **Protected/Unprotected Settings** box automatically displays **Text Field** if you choose **Protected** in the **Mainframe Data Type** box.



3. Select a font type from the **Font** list box in the **Protected/Unprotected Settings** box. Font types vary based on your platform.

4. Select a font style from the Font Style list box. Values are:

   Bold
   Plain
   Italics

5. Select a background color from the **Background Color** list box. Values are:

   Black
   Blue
   Gray
   Green
   Magenta
   Red
   Turquoise
   White
   Yellow

6. Select a foreground color from the **Foreground Color** list box. Values are:

   Black
   Blue
   Gray
   Green
   Magenta
   Red
   Turquoise
   White
   Yellow

7. Select a font size for the protected and unprotected fields in the **Font Size** list box on the **Main Panel Settings** box.

Note:

The Preview Settings box displays a sample of the options you choose on the Auto GUI Configurator tab.

The auto GUI settings you choose display in the Preview Settings box.

Using the Hotspots Tab

Use the **Hotspots** tab on the OC://WebConnect **configuration applet** window to map buttons that initiate PF keys. The Hotspots tab displays if you are configuring properties for a 3270 or 5250 session.

Setting Hotspots

1. Type in the **Match Text** field in the **Hotspot** box the text to display on the button that will map to and initiate an emulator key (such as a PF key).
2. Click the emulator key to which you are mapping in the list box.
3. Click the **Add Entry** button. The text and emulator key are listed in the **Hotspot Current Settings** box.
4. Click the **OK** button to save the changes.
   Click the **Cancel** button to exit without saving your changes.
   Click the **Defaults** button to delete your changes and replace them with default options.
   Click the **Apply** button to activate your changes.

Caution:

The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your hotspot options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your hotspot options to the default options, you cannot restore your options!

Note:

Hotspots are available for 3270 and 5250 sessions only.

Modifying Hotspots

1. Select **3270** or **5250** from the **Session** menu on the **Hotspots** tab. The hotspot options display for the selected emulation type.
2. Select the text and emulator key entry in the **Hotspot Current Settings** box. The text displays in the **Hotspot Configuration** box, and the emulator key displays in the **Match Text** box.
3. Remap the text to another emulation key in the **Match Text** box, or type different text in the **Hotspot Configuration** box to initiate the emulator key.

4. Click the **Replace Entry** button. The new text and emulator key entry displays in the **Hotspot Current Settings** box.
5. Click the **OK** button to save the changes.
   Click the **Cancel** button to exit without saving your changes.
   Click the **Defaults** button to delete your changes and replace them with default options.
   Click the **Apply** button to activate your changes.

Caution:

The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your hotspot options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your hotspot options to the default options, you cannot restore your options!

Deleting Hotspots

1. Select **3270** or **5250** from the **Session** menu on the **Hotspots** tab. The hotspot options display for the selected emulation type.
2. Click the text and emulator key entry you want to delete in the **HotSpot Current Settings** box.
3. Click the **Delete Entry** button. The entry is deleted.
4. Click the **OK** button to save the changes.
   Click the **Cancel** button to exit without saving your changes.
   Click the **Defaults** button to delete your changes and replace them with default options.
   Click the **Apply** button to activate your changes.

Caution:

The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your hotspot options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your hotspot options to the default options, you cannot restore your options!

Displaying Hotspots

To display hotspots:

Select the **Hotspots** checkbox from the **Settings** menu on your emulation screen.

Using the Attributes Tab

Use the **Attributes** tab on the OC:/WebConnect **configuration applet** window to assign display and emulation attributes for host screens.

Many applications use IBM base and extended attributes to identify characters and fields for special functions. Examples of IBM attributes are fields that are protected (cannot be edited) or that display blinking text.

The Attributes tab lists IBM attributes in the Application Attributes list. Emulation attributes displayed by your application can be assigned or mapped to the following:

Foreground color (selectable)

Background color (selectable)

Blink

Underline

Reverse video

High intensity

Transparency

Hotspots

OC://WebConnect has default attribute map files (*def3270.atm*, *def5250.atm*, and *defvt.atm*). Each file contains a production-set attribute map. You can use these default attribute maps or change them to suit your needs. You also can save the modified attribute maps to the original default filename or a new filename.

Mapping Display Attributes

To display the current attribute map for a listed application attribute

1. Select a session type from the **Session** menu and click on the **Properties** button.

Notes:

- Each session type, either 3270, 5250, VT, or 3287, has different application attributes.
- You must have a session open to view a session's attributes.

2. Click on the **Attributes** tab.
3. Select an application attribute from the **Application Attributes** box. OC://WebConnect shows the display attributes in the **Display Attributes** box.
4. Choose the display attributes you want from the **Display Attributes** box.
5. Click the **OK** button to save the changes.
   Click the **Apply** button to apply the attribute mapping changes.
   Click the **Cancel** button to cancel the changes.
   Click the **Default** button to replace your settings with the default settings.

Caution:

The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your attribute options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your attribute options to the default options, you cannot restore your options!

Application Attribute Considerations

Lightpen Attributes

Some host applications might use lightpen fields. OC://WebConnect uses production-set parameters to display lightpen-selectable fields in red. All non-selectable fields display in green. However, you can use the Attributes tab to map other display attributes to an application's lightpen fields.

Hotspot Attributes

You can map display attributes to Hotspots. Select the Hotspot - Selectable application attribute from the scroll list and map desired display attributes. The hotspots appear with the attributes you map to them.

Display Attribute Considerations

Foreground and Background Color Attributes

You can set the attribute's foreground and background colors and activate other characteristics, such as toggle buttons. When you select foreground and background colors, the **Display Color Sample** line shows a sample of the colors.This representation is the display attribute or the way the application attribute is to appear in the workspace.

Dotted Underline Attribute

This attribute instructs OC://WebConnect to mark column separators with decimal characters. A decimal character between fields shows the column boundary.

Button Attribute

This attribute allows the recognized application attribute to display with bordering that is similar to a physical key.

Inset Attribute

This attribute marks recognized application attributes as buttons that are “depressed” into the surrounding background.

Using the Color Tab

Colors that appear in OC://WebConnect's workspace are display colors that represent colors specified in the session’s data stream. The session’s color specifications are emulation colors that are mapped to the display colors available from your windowing system. Normally, the colors displayed by OC://WebConnect should allow you to perform the tasks you want. However, in some cases you might want to map new display colors to particular emulation colors so that information appears in the color you want.

The Colors tab shows a list of standard emulation colors. When you select an emulation color name, the display color that is mapped to the emulation color displays.

When you load a session file into memory, the session’s color map file also is loaded. OC://WebConnect automatically opens the color map file that is referenced in the session file.

OC://WebConnect has default color files (*def3270.clm*, *defvt.clm*, and *def5250.clm*). Each file contains a production-set color map. For example, a color map file can include an emulation color, such as blue, that is mapped to a light blue display color. In this case, OC://WebConnect displays light blue in its workspace when the application specifies blue in the session datastream.

Mapping Colors

During a session, you can replace current colors with new display colors.

1. Select a session type from the **Session** menu on the **Select Session Type** window, and click the **Properties** button.
2. Click the **Colors** tab.

Notes:

- You must have a session open to view a session's attributes.
- You can add a new color or remove a color by selecting the **Add Custom Color** or **Remove Custom Color** buttons.

3. Select an emulation color. The emulation color's assigned display color appears in the **Sample Display** box.
4. Select a standard color from the Standard Color list.
5. Click the **OK** button on the **Colors** tab to apply the changes to the host session in the workspace.
   Click the **Cancel** button to cancel the changes.
   Click the **Default** button to replace your settings with the default settings.
   Click the **Apply** button to activate your changes.

Caution:

The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your color options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your color options to the default options, you cannot restore your options!

Using the Keyboard Tab

Mapping Key Combinations

You can map keys to perform specific actions.

1. Select a session type from the **Session** menu on the **Sessions** menu.
2. Click the Pro**p**erties button.
3. Click the **Keyboard** tab.
4. Select a key action from the **Action Keys** list box on the **Keyboard** tab, such as **Tab**.
5. Click in the **Mapped To** list box.
6. Select the key(s) that you want to map to that action from the keyboard displayed on the **Keyboard** tab. The sequence will display in the **Mapped To** list box.
   Example: <CTRL+ALT+DEL>
7. Click the **Map** button
8. Click the **OK** button to save the key sequences.
   Click the **Apply** button to apply the attribute mapping changes.
   Click the **Cancel** button to cancel the changes.
   Click the **Default** button to replace your settings with the default settings.

Notes:

- OC://WebConnect saves the key map sequences to the server if you logged in with administration-level access. If you logged on locally, OC://WebConnect saves your key map sequences locally.
- The key map sequences are saved with a *.kbm* extension. The default *.kbm* files for OC://WebConnect are *def3270.kbm*, *def3287.kbm*, *def5250.kbm*, and *defvt.kbm*.
- Although it is not inhibited by the GUI Keyboard Mapping tool, mapping functions to alphanumeric keys is not allowed. For example, it is not possible to map 3270 PF4 to the L key on the PC keyboard.

Remapping Keys

You can remap keys to perform different actions.

1. Select a session type from the **Session** menu on the **Sessions** tab.
2. Click the **Properties** button.
3. Click the **Keyboard** tab.
4. Select the key map action you want to remap from the **Keyboard Map Action Keys** list box.
5. Click the **Unmap** button. The key mapping is removed from the **Mapped To** list box.
6. Select the key(s) that you want mapped to that action on the keyboard displayed on the Keyboard tab. The mapped sequence displays in the **Mapped To** list box.
7. Click the **Map** button when all keys have been selected.
8. Click the **OK** button to save the key sequences.
   Click the **Cancel** button to disregard the key sequences.

Notes:

- OC://WebConnect saves the key map sequences to the server if you logged in with administration-level access. If you logged on locally, OC://WebConnect saves your key map sequences locally.
- The key map sequences are saved with a *.kbm* extension. The default *.kbm* files for OC://WebConnect are *def3270.kbm*, *def5250.kbm*, and *defvt.kbm*.

Caution:

- The **Defaults** button eliminates your changes. If you accidentally click the **Defaults** button and delete your keyboard map options, you can restore them by clicking the **Cancel** button. If you click the **Save** button after you accidentally change your keyboard map options to the default options, you cannot restore your options!

Saving Display Options

OC://WebConnect saves the display options to the server if you logged in with administration-level access. If you logged on in user mode, OC://WebConnect saves your display options locally

Chapter 11: Emulation Client Applet Features and Interface

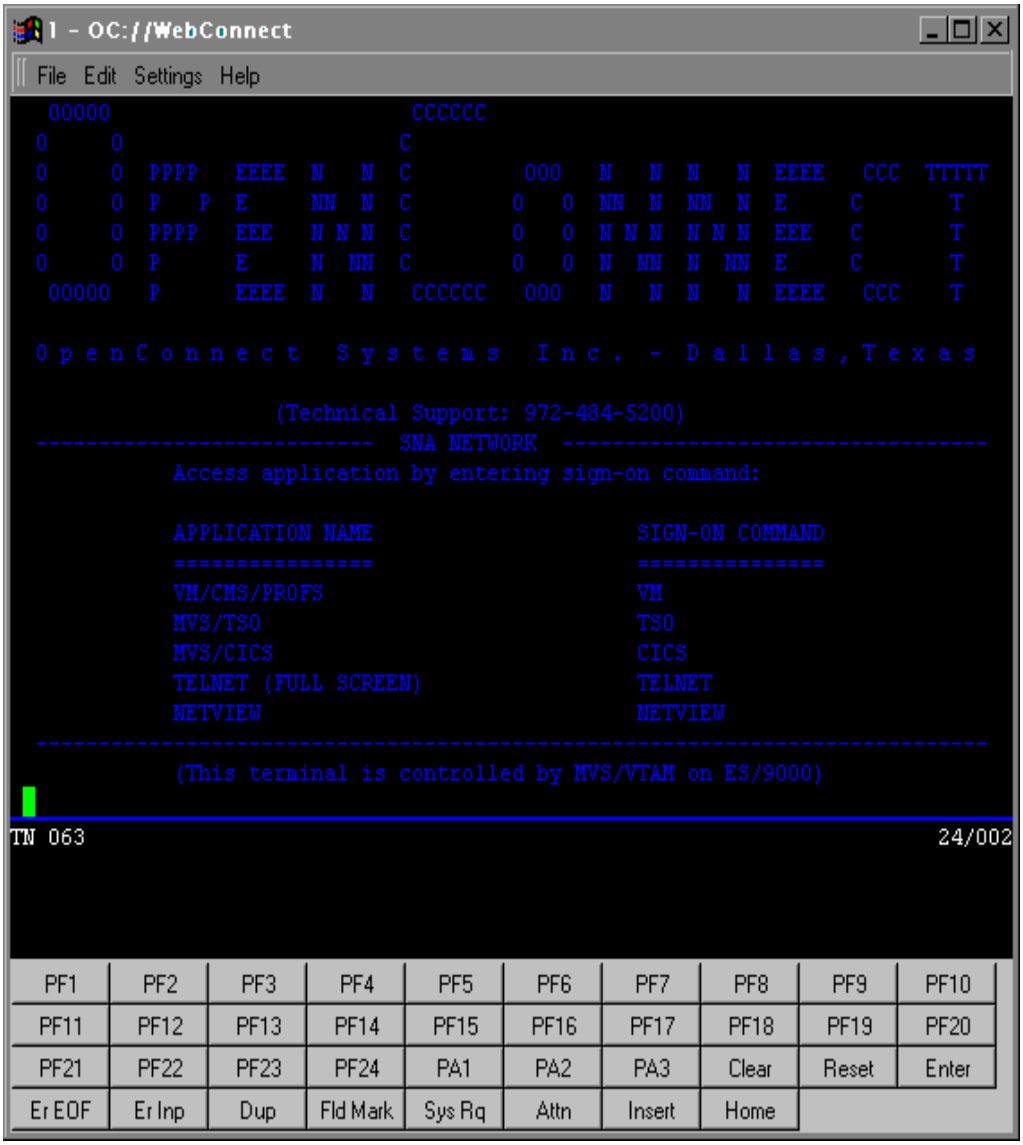
Overview

OC://WebConnect includes emulation clients for 3270, 3287, 5250 and VT emulation. The emulation support provided by the OC://WebConnect Emulation server and individual Java emulation client applets. Beyond the basic emulation support, OC://WebConnect applets provide a rich Graphical User Interface(GUI) which support the end user features usually provided by a traditional desktop emulation package.

A java applet is an application program written in Java which is executed within a java enabled browser interface. Because the major portion of the application user interface is provided by the browser and applet is fairly small and is ideal for use on the Internet or a corporate intranet. OC://WebConnect provides three Java emulation client applet packages: Ultralite, Enhanced and Power User. The packages are grouped according to browser support and feature sets.

3270, 5250 and VT Emulation Client User Interface Features

The 3270, 5250, and VT emulation client applet have similar user interfaces an share many user interface features. The client window is made up of a menu bar, emulation space, and an option clickpad. The menu options provide a variety of features from the ability to spawn a new applet to Help options. The emulation space displays information as a host connection is being made and is the area in which the host data is presented. The clickpad is a group of buttons to give mouse access to emulation functions. The interface features are explained in detail below.



FILE FEATURES	DESCRIPTION
New	The New menu option allows end users to spawn another java emulation session from the current session. The applet is downloaded againg. This option can be limited through the session configuration option “sessions per applet”.
Print Screen •	Print screen allows an end user to print locally the emulation screen. The print screen functionality is provided through 1 of 3 methods: OC://Webprint installed on JDK 1.0 or greater enabled browsers

FILE FEATURES	DESCRIPTION
	• Javascript included with some browsers. • JDK 1.1 print functionality included with JDK 1.1 enabled browsers. Ultralite supports only OC://Webprint and Javascript Enhanced and Power User support all three print solutions.
Ind$File or APVUFile transfer	File Tranfer options allow end users to send and receive files from the S/390 host to the browser system. This functionality is only available with the 3270 Power User applet.
Associate Print	Allow a 3287 print session to be spawned from a 3270 emulation session. The 3287 session options are configured within the 3270 session configuration.
Exit	Terminate the connection to the host and quit the execution of the applet. The applet may still be cached by the browser and will not need to be downloaded from the server as long as the browser task is still running.

EDIT FEATURES	DESCRIPTION
copy	The copy feature allows an end user to mark a stream of host text and copy it to the system clipboard. This functionality is provided by one of two methods: • OC://Webprint installed on an JDK 1.0 or greater java enabled browser • JDK 1.1 copy paste functionality included with a JDK 1.1 Java enabled browser.
Paste	The paste feature allows an end user to paste text, in the system clipboard, to the emulation area or to any other clipboard enabled window. Example: Copy text from an emulation session to Windows Wordpad. This functionality is provided by one of two methods: • OC://Webprint installed on an JDK 1.0 or greater java enabled browser JDK 1.1 copy paste functionality included with a JDK 1.1 Java enabled browser.

SETTINGS	DESCRIPTION
clickpad	The clickpad is a group of buttons that appear at the bottom of an emulation screen that allows the end users to click on a button to send an emulation aid key (Enter, F1, etc) to the host. The initial display of the clickpad is controlled via a session configuration setting. While the applet is running the end user can use this menu option to display or hide the clickpad.

SETTINGS	DESCRIPTION
Show printer dialog	The Show Printer Dialog allows the instructs the applet to display a dialog, when a print job created, which allows the end user to choose a printer, select landscape or portrait, etc. If this option is not enabled the print job will be sent to the default printer and be in portrait mode. This option is available with OC://Webprint only.
Hotspots	The Hot spots option allow the end user to enabled the display of a hot spot button over text that has been defined as a hotspot. The end user can then use a mouse to send the aidkey associated with that string of text. See Chapter 10 for more details. This option is available with Enhanced and Power User applets.
GUI Screen	The GUI screen option can be enabled to instruct the applet to automatically render the emulation screen to a graphical look and feel. This option is only available with 3270 and 5250 Power User applets.

FEATURE	DESCRIPTION
Font Size	The Ultralite applet allows end users to change the size of the font used for the emulation text and the size of the applet window. The initial font and window size is determined from a setting in the session configuration. This option isn't available for Enhanced and Power User applets.

HELP	DESCRIPTION
Help desk	The Help desk option displays emulation, session, security, and connection information. The display of host information may be limited by the OC://WebConnect server setting "Suppress Host Information".
Key map	Displays the currently mapped key mapping of browser platform keys to emulation functions.
Java Logging	Outputs java messages to the browser's java console. Used for debugging.
About	Displays the applet type and version number.

EMULATION SPACE	DESCRIPTION
Connection Messages	Messages are output to the emulation space: • detailing attempts to connect to the OC://WebConnect server • detailing attempts to connect to the S/390 host.
Security Messages	Messages are ouput to the emulation space about the generation of encryption keys. This is only if encryption is being used.
Host Data	After a host connection has been established Host data is displayed

EMULATION SPACE	DESCRIPTION
	according to host data within the Emulation space of the applet window. The data is displayed according to host data attributes unless Hotspots or AutoGUI is being used.

FEATURE	DESCRIPTION
Window resizing	The Enhanced and Power User applets allow the client window to be resized by a mouse click and drag along the applet borders. When the applet window is resized the font may change to best fit the session window.

More Information:

For more information about Session configuration creation, deletion, or modification refer to one of the following chapters within this document:

- Chapter 6 3270 Session Configuration and Features
- Chapter 7 5250 Session Configuration and Features
- Chapter 8 3287 Print Session Configuration and Features
- Chapter 9 VT Session Configuration and Features

3287 User Interface Features

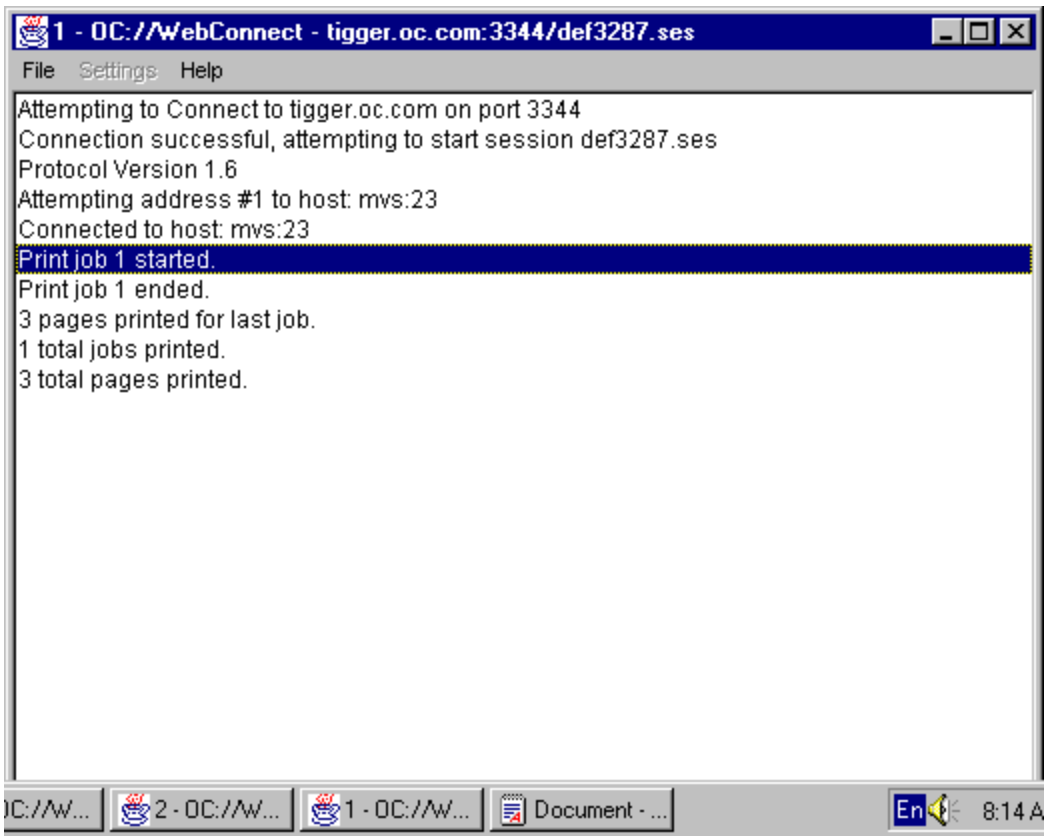
The 3287 session user interface is a simple dialog which displays the progress of 3287 print jobs. The end user does have a few options to effect the printed output. The 3287 client user interface features are explained below.

FILE	DESCRIPTION
Exit	Terminate the connection to the host and quit the execution of the applet. The applet may still be cached by the browser and will not need to be downloaded from the server as long as the browser task is still running.

SETTINGS	DESCRIPTION
Font Autofit	The Font Autofit option instructs the Java applet to compress the font to fit the line of print on the printed paper.This is useful if the print data stream defaults to 132 columns of print but is in reality just 80 columns of print. This option is available with OC://WebPrint only.
Show printer dialog	The Show Printer Dialog allows the instructs the applet to display a dialog, when a print job created, which allows the end user to choose a printer, select landscape or portrait, etc. If this option is not enabled the print job will be sent to the default printer and be in portrait mode. This option is available with OC://Webprint only.

HELP	DESCRIPTION
Help desk	The Help desk option displays emulation, session, security, and connection information. The display of host information may be limited by the OC://WebConnect server setting "Suppress Host Information".
About	Displays the applet type and version number.

PRINTER DIALOG	DESCRIPTION
Connection Messages	Messages are output to the 3287 session dialog • detailing attempts to connect to the OC://WebConnect server • detailing attempts to connect to the S/390 host.
Security Messages	Messages are ouput to the 3287 Print dialog about the generation of encryption keys. This is only if encryption is being used.
Print Job Messages	Messages are output to the 3287 session dialog: • reporting the status of a print job (e.g Started, Ended) • reporting the number of print jobs completed • reporting the number of pages printed



More Information:

For more about emulation features refer to the following chapter within this document:

- Chapter 8 3287 Print Session Configuration and Features

Choosing an Emulation Client Applet Package: Ultralite, Enhanced, or Power User

A java applet is an application program written in Java which is executed and run within a java enabled browser interface. Because the major portion of the application user interface is provided by the browser and applet is fairly small and is ideal for use on the Internet or a corporate intranet. The selection of which OC;//Webconnect emulation client java applet package to use should be made according to the browser being used, the emulation features needed, and the network environment.

Browser Environment and Java Support

Because OC://WebConnect emulation clients are Java applets the version of browser being used is important in the selection of an applet package. Different browsers and browser versions support different Java features. A Java Virtual Machine (JVM) is included with all Java enabled browsers. The version of the JVM will indicate which Java features are supported. The level of Java support is sometimes referred to as the level of JDK (Java Development Kit) support. For example browsers which support only JDK 1.0 features do support direct printing or copy/paste but require the use of a coprocess or Javascript to print or copy/paste.

In the case of OC://WebConnect Java emulation client applets Ultralite supports JDK 1.0 enabled browsers. Enhanced and Power User applets support JDK 1.1 and greater enabled browsers.

Applet Feature Sets

OC://WebConnect packages are grouped according to the level of features sets required. The Ultralite packages provide basic emulation along with print and copy/paste functionality. The Enhanced applets provide all Ultralite features as well as JDK 1.1 direct print, copy/paste, and local configuration files. Power User applets support all Ultralite and Enhanced features as well as file transfer and AutoGUI. 3287 is available as both an Ultralite and Enhanced applet.

Network Environment

The network environment in which OC://WebConnect operates and is accessed plays a roll in the decision to choose one applet over another or to use OC://WebConnect security features. OC://WebConnect applets are downloaded to the browser platformthe first time the applet is started during a browser session. When the browser user exits the browser the applet no longer resides on the browser platform. This allows the administrator to maintain software and software configurations on the server while the client platform is dynamically updated.

The more features available in an applet increases the size of the applet. Because the applets are downloaded the size of the applet affects the amount of time and network traffic to download that applet. If an end user is on a corporate intranet the amount of time to download a client emulation applet might not be important. If an end user is connecting to OC://WebConnect via the Internet and has a slow modem connection the size of the applet might be very important.

The use of OC://WebConnect security features may be of a necessity depending upon the sensitivity of data being accessed and the security of the network being used. The use of security features requires the generation of encryption keys and the encryption and decryption of data. The choice of security features is a speed vs. security decision.

If an end user is using the corporate intranet and isn't accessing sensitive data RSA or SSL may not be required. If an end user is using the Internet and accessing sensitive corporate data security may be a necessity.

Questions To Ask?

1. **What is the enduser environment?**

   Browser - What browser are the end user' using? A JDK 1.0 or JDK 1.1 enabled browser?

   Language - Do the end user require a single byte or double byte client? Enhanced or Power User?

2. **What end users features are needed?**

   What emulation do the end users need? 3270, 5250, VT?

   Is 3287 print a requirement? OC://Webprint, Javascript, or JDK 1.1 print

   What emulation features do the end users need?- file transfer? Power User or Not?

   Do the end users need copy/paste functionality OC://Webprint or JDK 1.1

   Do the end users need screen print? OC://Webprint, Javascript?, JDK 1.1 print?

   Do the users need to configure their own key, color, attribute, etc maps? Allow User configuration?

3. **What are the administrative needs?**

   Is security a concern? Is security a paramount concern? RSA, SSL?

   Does the administrator need to limit the number of sessions an end user can start? # of session allowed?

   Does the administrator need to monitor response time? RTM or Not?

   Does the administrator need to automatically disconnect defunct sessions? Are You There?

4. **What is the network environment?**

   Is the end user connecting over the Internet or Intranet? Is applet size important?

   Is the time to download the applets important? Is applet size important?

   Is the connect time important? Encrypt or Not?

   Is the time to send and receive data important? Data Compression or Not?

More Information:

For more information about security features see *Chapter 15 Security Overview*.

For more information about customizing the use of OC://WebConnect see *Chapter 12 Customizing OC://WebConnect*.

Browser Support of Applets

BROWSER SUPPORT	ULTRALITE	ENHANCED	POWER USER
Netscape 2.x	Yes	No	No
Netscape 3.x	Yes	No	No
Netscape 4.x	Yes	Yes	Yes
Internet Explorer 3.x	Yes	No	No
Internet Explorer 4.x	Yes	Yes	Yes
HotJava 1.x	Yes	Yes	Yes

Emulation Applet Feature Breakdown

EMULATION SUPPORT	ULTRALITE	ENHANCED	POWER USER
3270	Yes	Yes	Yes
3270E	Yes	Yes	Yes
5250	Yes	Yes	Yes
VT	Yes	Yes	Yes
3287	Yes, Using OC://Webprint or Javascript	Yes	No

EMULATION AND CONNECTIVITY FEATURES	ULTRALITE	ENHANCED	POWER USER
IND$File Transfer	No	No	Yes
APVUFile Transfer	No	No	Yes
Associate Print	Yes, Using OC://Webprint or	Yes, Using JDK 1.1 Print, OC://Webprint,	Yes, Using JDK 1.1 Print, OC://Webprint,

EMULATION AND CONNECTIVITY FEATURES	ULTRALITE	ENHANCED	POWER USER
	Javascript	or Javascript	or Javascript
RTM (Response Time Monitoring)	No	Yes	Yes
IP PassThru	Yes	Yes	Yes
Multiple IP Address	Yes	Yes	Yes
Keep Alive	Yes	Yes	Yes
Telnet Are You There	Yes	Yes	Yes
Client Are You There	Yes	Yes	Yes
Virtual Gateway Support	Yes	Yes	Yes
terminal type demotion	Yes	Yes	Yes
model 2,3,4,5	Yes	Yes	Yes
configured device names	Yes	Yes	Yes

PRINT SUPPORT	ULTRALITE	ENHANCED	POWER USER
3287 Print	Yes, Using OC://Webprint or Javascript	Yes, Using JDK 1.1 Print, OC://Webprint, or Javascript	Yes, Using JDK 1.1 Print, OC://Webprint, or Javascript
Print Screen	Yes, Using OC://Webprint or Javascript	Yes, Using JDK 1.1 Print, OC://Webprint, or Javascript	Yes, Using JDK 1.1 Print, OC://Webprint, or Javascript
Print Autofit	No	Yes	Yes

USER INTERFACE FEATURES	ULTRALITE	ENHANCED	POWER USER
choose font	Yes	No, font changes dynamically when the window is resized.	No, font changes dynamically when the window is resized.
Window resize	No, size of window is determined by font size.	Yes	Yes
optional click pad	Yes	Yes	Yes
copy/paste	Yes, using OC://Webprint	Yes	Yes
hot spots	No	Yes	Yes
auto gui	No	No	Yes
keymap display	Yes	Yes	Yes
user defined key, color, attribute, hotspot and	No	Yes	Yes

USER INTERFACE FEATURES	ULTRALITE	ENHANCED	POWER USER
autogui maps.			
administrator defined keymap	Yes	Yes	Yes
administrator defined color, attribute, hot spot map	No	Yes	Yes
administrator defined autogui map	No	No	Yes

SECURITY FEATURES	ULTRALITE	ENHANCED	POWER USER
RC4 40 bit encryption	Yes	Yes	Yes
RC4 128 bit encryption	No	Yes	Yes
SSL server authentication	No	Yes	Yes
Client token authentication	No	Yes	Yes
Limit Number of New sessions	Yes	Yes	Yes
data compression	No	Yes	Yes
Conceal Host Information	Yes	Yes	Yes

LANGUAGE SUPPORT	ULTRALITE	ENHANCED	POWER USER
4 Server languages: English., French, German, Castillian Spanish	Yes	Yes	Yes
12 Single Byte Language Client Interface and data stream: list languages	Yes	Yes	Yes
Double Byte Language Client Interface and keyboard input and data stream: Chinese Chinese	No	Yes	Yes

Japanese Korean			
Keyboard Input and Data Stream only: Turkish	No	Yes	Yes

MISCELLANEOUS	ULTRALITE	ENHANCED	POWER USER
TCL Scripting	Yes	Yes	Yes
Java Logging	Yes	Yes	Yes

JDK 1.1 Applet Certificates and Granting Local Files System Access

The Enhanced and Power User applets use JDK 1.1 features that are available when using JDK 1.1 java enabled browsers. The features require additional security measures because the applet must access the browser platform's file system.

Security is provides using certificates included with the applets and verified by the browserss and applets work by using certificates. When the browser attempts to execute the applet it encounters a certificate, delivered with the applet, that details the developer of the applet and what type of access to the local file system is required. The browser then displays a dialog prompting the user to grant the local file system access or not. If the user chooses to *trust* the applet the applet begins and all functionality is available for use. The browser user can choose to grant the privileges on a temporary or permanent basis. If privileges are granted temporarily future applet sessions will require the user to grant privileges again. If privileges are granted permanently subsequent downloads and use of the applet will be automatically granted privileges. The certificate and privileges can be removed from the browser under the Security or Certificates section of each browser.

These features which require access to the local file system are

- the ability to write to the disk to store User configuration files for key maps, color maps, etc.;
- the ability to use the local print spooler for print screens and/or 3287 print.
- the ability to access the local clipboard for copy/paste functionality.
- the ability to access the local file system to retrieve and store files for file transfer operations.

Chapter 12: Customization of OC://WebConnect

Overview

OC://WebConnect can be easily incorporated into an intranet or Internet web site to provide the necessary centralized control of feature and host access and provide an easy user interface. The product comes with a default user interface that can be used by both the end user and administrator in a production environment. If changes are needed in the interface the product has been designed to make customization as easy as possible.

The OC://WebConnect product can be broken down into several pieces some of which can be modified and some can even be replaced by a 3rd party tool. The basic OC://WebConnect is the OC://WebConnect Emulation server and the java client applets. The HTML session page, the HTML administration and configuration pages, the GUI configuration applet, the HTTP server, CGI-BIN administrative interface, on line User's guide, etc., are the additional utilities that bring the pieces together and deliver a cohesive product that can be used by the Emulation client end users as well as the OC://WebConnect Administrator. All of these make up the product called OC://WebConnect.

OC://WebConnect User Interface Architecture

The OC://WebConnect default user interface is made up of seven sections:

End User Interface:

- Session page (index.html)
- end user applets (Ultralite, Enhanced, Power User)

Administrator's Interface:

- Administration - html administration pages
- Configuration - html configuration pages for server and session configuration.
- GUI Config - Java applet for server and session configuration

Online Documentation:

- User's Guide - Online User's Guide
- Help - Context Sensitive Help

End User Interface

Of the seven sections, only two are used by the Client emulation user (end user). The **Session** page is used to allow the end user to choose emulation options and download the necessary applet files. The session page is several HTML pages that use a CGI-BIN utility and macros to list available sessions, give the end user a few configuration options and deliver the applet to the end user's browser. The pages can be customized in many ways. Knowledge of HTML and OC://WebConnect is usually required to customize these pages. Knowledge of Javascript and CGI will allow even more extensive customization.

The applet files are the java client applets that provide the client interface and host connection. The three applet types, Ultralite, Enhanced, and Power User, give the end user a choice of feature sets and Browser support. Extensive customization of the emulation client interface is available by using OpenVista. Knowledge of Java is needed to take the greatest advantage of the OpenVista tool.

Administrator's Interface

The four remaining sections are used by an OC://WebConnect Administrator to configure and administer the OC://WebConnect server and clients.

The User HTML and Client applet interfaces can be customized in many ways. It is not recommended that the Administrator's features be customized.

Customization Ideas

The main reason that OC://WebConnect has well defined pieces is to allow customization. Except for the OC://WebConnect Emulation server all the other pieces could be replaced. Although some things, like the configuration tools, we suggest not be replaced.

Some of the ideas suggested for customization are:

- make a link to the OC://WebConnect URL from a corporate web page.
- Make the initial OC://WebConnect HTML page a user page only. Split the End User Interface and the Administrative Interface. In other words remove End User access to the Administrative and Configuration options.
- Change the look and feel of the Session pages (index.html) to fit in with the existing corporate web pages.
- Add or remove the end user configuration options when starting an emulation session.
- identify the applet choices by browser requirements.
- set up the user interface by departments.
- deliver different applets with different security options to different user groups.
- allow the end user to select the LU to which they will connect.
- to nail down printer LUs.
- allow the end user to enter a userid and chose the session configuration, applet type and security by user id.
- add or remove emulation client menu option.
- replace the emulation client interface green screen with a custom graphical interface.
- replace the host application interface with a simpler graphical interface.
- replace the emulation client interface with a client interface that combines multiple data sources, by using OpenVista development tool.

Customization tools

A large amount of customization can be done with HTML. Combined with the cgiinfo utility provided with OC://WebConnect the product can be cleanly integrated into corporate web pages and be easily maintained by the dynamic HTML provided by the cgiinfo utility. Additional tools include javascript, CGI scripts, OpenVista, and other Java tools.

Customizing the HTML Interface

The default HTML interface provided with OC://WebConnect combines both the Administrator's and End User's interface. It is provided primarily for demo purposes but can be used, as is, in a production environment. The default HTML interface takes full advantage of the cgiinfo utility and macros to provide the current interface with the current server and session configuration information.

Default HTML files purpose and locations

The HTML **Sessions** page is the main index.html, which is organized into 4 frames made up of 4 different HTML files. In addition to the pages that makeup the **Sessions** page the tclient.html serves as a template for the dynamic HTML created to start an applet. All these files are located in the OC://WebConnect *html* directory.

- **index.html** - main interface for both end users and administrators. The index.html is the html that will be displayed when a user contacts OC://WebConnect through a browser using the OC://WebConnect host and port only and does not specify another html file.
- **header.html** - html which includes the OC://WebConnect logo, version number, and identifies the html page as the Sessions page.
- **sidebar.html** - html which provides links to other OC://WebConnect Administration, Configuration, and documentation functionality.
- **footer.html** - html which blends into the sidebar and contains the applet tag source when a session is started.
- **main.html** - html which uses the cgiinfo utility and macros to provide the end user a list of session configurations, provides user options for applet type and SSL security, and a start button. Gives the user choices when the start is chosen, cgiinfo queries the OC://WebConnect server and produces the applet tag which is returned to the browser combined with the tclient.html.
- **tclient.html** - html which is used as a template for the dynamic html used to start a session. The template includes a section in which the

applet tag generated for the session is specified and Javascript is included if Javascript printing.

Caution:

It is imperative that a backup copy be made of the html files prior to any modification. It is recommended that the default HTML files be used for Administrative and configuration purposes. The *cgiinfo.exe* uses the macros contained in these HTML templates to communicate with OC://WebConnect to retrieve the requested information through a third-party HTTP server.

Some Examples of HTML Changes

- ***Make a link to the OC://WebConnect URL from a corporate web page***

  Example:

  ```
  <P><A HREF="http://host1.oc.com:2080"><IMG SRC="hostaccess.gif"
  BORDER=0 HEIGHT=70 WIDTH=70></A></P>
  ```

- ***Make the initial OC://WebConnect HTML page a user page only. (restrict administrative and configuration access)***

  Split the End User Interface and the Administrative Interface. In other words remove End User access to the Administrative and Configuration options. Below are a few simple ways to remove the administrative options.

  Replace the index.html file with the main.html file
  or
  To remove the sidebar.html frame from the index.html delete the following line from the index.html file:

  ```
  <FRAME NAME="Sidebar"
  SRC="sidebar.html?host=sultry.oc.com&port=4273&httpport=2081"
  SCROLLING=NO NORESIZE>
  ```

- ***Simple look and feel (change to corporate look and feel)***
  - Modify the index.html, sidebar.html, and header.html files to replace the OC://WebConnect logos with corporate logos.
  - replace the background gif in the main.html file to match background used by other pages on the corporate website.
- ***Add or remove the end user configuration options when starting an emulation session***
  - modify the main.html file to remove the list box for the applet type (ultralite, enhanced and power user and add another input line to the cgiinfo call to hard code the applet type.

    Delete the following lines

    ```
    <SELECT NAME="type">
    ```

<OPTION VALUE="lite">Hi mom

<OPTION VALUE="enhance">Enhanced

<OPTION VALUE="power">Power User

</SELECT>

<P>:

Add the following line below the other cgiinfo input type lines:
<INPUT TYPE="hidden" NAME="type" VALUE="lite">

- ***set up the user interface by departments***
  - replace the main.html page with an HTML page that specifies departments or user groups
  - the new HTML page should have links to new HTML pages, similar to the main.html, for each department that specify the correct OC://WebConnect applet options for that department or group
  - add password protection to the department or group pages that require additional security. For a simple implementation of logons see the find.html in the OC://WebConnect *samples* directory.

Using a CGI script to nail down 3287 Lus for printing.

A set of sample cgi scripts is provided with the OC://WebConnect product for nailing down LUs to a specific user id. See the section “CGI scripts for choosing LUs” for more information

1. A copy of Perl installed must be available on the platform where OC://WebConnect is installed. Winperl.exe will not work on NT Platforms.
2. If you are using a third party web server it must reside on the same platform as OC://WebConnect.
3. Copy the files listed below from \wc\samples\3287resource to the directories indicated.
   a) Find.html. This is a sample HTML file that prompts the user for an ID and then initiates a CGI script. This file must be placed in the /doc directory of your HTTP server if you are using a third party web server. If you are using the OC://WebConnect server this file must be placed in the /wc/html directory.
   b) Vg.pl. This is a CGI script written in Perl. This file must to be placed in the same directory as your cgi executable. If you are using the OC://WebConnect server and have accepted the defaults at installation the directory would be C:\WC.
   c) Vginfo.txt. This is a flat file in which each line has a user ID with a host, port, and LU resource associated with it. This file must be placed in the \doc directory of your HTTP server if you are using a third party web server, an example is \InetPub\wwwroot if you are using the Microsoft IIS web server. If you are using the OC://WebConnect server this file must be placed in the \wc\html directory.
   d) Vgtemplate.html. This is a template used by "vg" to build a new HTML page. This file must be placed in the \doc directory of your HTTP server if you are using a third party web server,

an example is \InetPub\wwwroot if you are using the Microsoft IIS web server. If you are using the OC://WebConnect server this file must be placed in the \wc\html directory.

4. Edit vg.pl and make the following changes if necessary. These 4 variables should be the only variables you have to change:

a) #!/usr/local/bin/perl Change to the location where Perl is located and add the exe extension to the perl command.
b) $filename = "/Netscape/Server/cgi-bin/ocs/vginfo.txt"; Change to the location of the vginfo.txt file.
c) $htmlfile = "/Netscape/Server/cgi-bin/ocs/vgtemplate.html"; Change to the location of the vgtemplate.html file.
d) $delimiter = "::"; Change the delimenter to what you use to deliminate between the fields in your txt file

5. Edit the vginfo.txt file and modify the data as needed. The fields are user ID, host, port, and LU resource.
6. Establish a session and view the frame source with your browser. Copy the source and paste to a text document. Save the document as \wc\html\vgtemplate.html, which will overwrite the current file. NOTE: If you do not want the user to see the button on the html page change the applet values for height and width to 0.

How to create Static html to download and start an emulation applet

Applet tags with different parameters and parameter values are generated by OC://WebConnect depending upon the browser, browser version and user choices of session configuration, applet type and SSL. The macros and cgiinfo interface to the OC://WebConnect server provide dynamic applet tags based upon the current server and session configurations, the browser being used, and the user choices.

Not all applet tags will work with all browsers for example

- an applet tag generated to work with a JDK 1.1 java enabled browser will not work with JDK 1.0 java enabled browsers.
- an applet tag generated when using Netscape or Internet Explorer may not work with HotJava.

Capturing the OC://WebConnect dynamic applet tag for starting an emulation client

The applet tag source cannot be captured until it is generated when a session is started. Because the applet tag is generated the source does not exist until the start button is chosen.

1. Start OC://WebConnect and connect to the OC://WebConnect “Sessions”(*Index.html*).
2. Choose the session configuration, applet and security settings

3. Start a session
4. The applet will appear and a connection will be made.
5. The HTML **Sessions** page, which is now showing on the browser, is made up of 4 frames: header.html, main.html, sidebar.html, and footer.html. The applet tag is part of the footer.html frame in the lower left hand part of the Sessions page.
6. Using Netscape to capture the applet tag source button, click on the bottom left corner of the Sessions page with the left mouse and select the menu option **View Frame Source**.
7. Using Internet Explorer to capture the applet tag source click on the bottom left corner of the Sessions page with the right mouse button and a menu will appear. From that menu select the menu option **View Source**.
8. Copy the HTML source code needed to a new *html* file.
9. Save the file in the HTML sub-directory within the directory containing the wcd OC://WebConnect server.
10. The user can then access the new HTML file by including the html filename in the URL or a link to it can be included on another html page.

Emulation Applet Tag parameters

For a browser to download and execute an applet the browser needs to know what applet to download and where it is located. The applet needs information and applet parameters, to determine how it is to be used.

HTML is the method used to instruct a browser to download and execute an applet and specify the parameters required by the applet. The HTML specific to a java applet is called an “applet tag.” The applet tag is made up of the name of the applet files, where to find the applet on the HTTP server, and applet parameters.

The basic information is the applet filenames that should be downloaded and where the files are located on the server.

The OC://WebConnect emulation applets applet parameters include information such as the session configuration file name, SSL encryption and authentication information, data overriding what is in the session file, etc.

See section below for more detail on filenames and applet parameters

Applet tag syntax:

<applet archive=*xxxxx.xxx* code=*xxxxx.class* CODEBASE="*/path*" width=*nnn* height=*nnn*>

<param name = "cabbase" value="*xxxx.cab*">

<param name ="*parametername*" value="*paramxxxx*">

</applet>

Applet filename data

APPLET TAG INFO	DESCRIPTION	POSSIBLE VALUES
• <applet> </applet>	html keywords used as section delimiters specify applet information. All data between the <applet> and </applet> pertain to an applet	No value just keywords that delimit the applet tag section.
Archive •	Specify the filename of the java applet package. A java applet is a collection of java class files. Applets are packaged differently depending upon the browser and security option. Each browser supports a different format for packaging, compressing and signing files. Internet Explorer does not use the archive setting.	See Table 1 Applets Archive and cabbase values for applets without SSL or See Table 2 AppletsArchive and cabbase values for applets with SSL.
Code	The class file which should be executed first. Each applet package (jar,zip,cab) have many class files. One class file must be executed first. That class file will call and load the rest of the class files as needed.	See Table 3 Code Values.
CODEBASE	The subdirectory in which the java applet files located.	Default= "/" This is relative to the OC://WebConnect home directory and will change when using a 3rd party HTTP server. If using a 3rd party HTTP server is being used this parameter should be specified as a full URL path to the location of the OC://WebConnect emulation client packages.

Width	Specify the width of a button to be displayed on the HTML page, which will allow the user to start another session or the first session if autostart is set to 0.	Default = 0, the button does not display.
Height	Specify the height of a button to be displayed on the HTML page which will allow the user to start another session or the first session if autostart is set to 0.	Default = 0, the button does not display.
MAYSCRIPT	Keyword that instructs the browser and applet that Java scripting may be accessed by the applet. This setting is specifically for printing using the Javascript solution.	Do not include keyword if Javascripting is not an option.
Cabbase	Specifies the applet package which should be loaded by Internet Explorer. Other browsers will ignore this option.	See Table 1 Applets Archive and cabbase values for applets without SSL or See Table 2 AppletsArchive and cabbase values for applets with SSL.

Emulation Applet Parameters

APPLET PARAMETER	DESCRIPTION	POSSIBLE VALUES
host	The host name or IP address of the platform where the OC://WebConnect server is running. For security reasons, the Java 1.0 client software only connects to the server where the client originated, the machine that is the source of the download. This restriction is removed for signed Java 1.1 clients. (Required Parameter)	Any valid tcp/ip host name or ip address
titlehost	The host name or IP address which should be displayed in the title bar of the java emulation applet.	Any valid tcp/ip host name or ip address
port	The OC://WebConnect Webserver port that has been established to listen for emulation traffic. If SSL is not being used this is the Java or JCP port. If SSL is being used this is the Javas or Secure JCP port. (Required Parameter)	If a 3rd party server is specified this should be the path to the emulation client java and text files.
session	The filename of the session configuration file to be used. This filename is determined when an administrator creates the session configuration file.	Default session file names are as follows: def3270.ses def5250.ses def3287.ses defvt.ses. See the OC://WebConnect cfgdir/ses directory for a list of all session file names.
Beepfile	The audio file that will be used for the emulation bell. The audio file plays when the host sends a bell character to the emulation client.	Default=beep.au. A different ".au" file can be used, but must be stored in the OC://WebConnect html directory.

autostart	The number of sessions that should automatically be started.	Default = 1 Range = 1 - license key limit Specify a 0 to load but not execute the applet. The end user can start the applet by using the button, "Start Session", option.
Button	A button can be displayed to allow the user to start an initial session if autostart is set to 0. An additional session will be started if autostart is set to greater than 0. To eliminate the ability for an end user to start additional sessions do not include this applet parameter.	Default = "Start Session" The value for this option is the string of text to be displayed on the button.
Htmlport	The port number set up to server http traffic. This could be the wsd port or a 3rd party server port. This option is only required for non-US English support of the Ultralite applets.	Default – 2080
serverVersion	The version of the OC://WebConnect Emulation server to which the applet will connect. This version number can be determined by looking at the OC://WebConnect status page or an OC://WebConnect trace file.	WC plus the server version. Example: "WC3.2"
serverType	The type of platform on which the OC://WebConnect server is running.	UNIX NT
browserName	Required for ultralite applet running	MSIE,Netscape,HotJavas
browserVersion	Required for ultralite applets	The Version number
time	A time limit value to be used by the Max session and client token authentication features.	This value must be generated by the OC://WebConnect server.

Clickpad	To display the clickpad when the applet it initially displayed. The clickpad may be enabled or disabled via a client applet menu option.	ON OFF
emulation	Specify which emulation type is to be used.	3270 5250 VT 3287
langname	The code for the client and/or emulation language. This is the language which will be used for the client interface.	See Table 4 Client Language Applet Tag Values.
Printimpl	This instructs the applet to which print solution is to be used for Print Screen and/or 3287 print.	none - to disable Print Screen and 3287 print functionality JDKimpl - to use the JDK 1.1 print method embedded in the browser JSPrintImpl- to use Javascript included in the html file and supported by the browser. JprintImpl - to use the OC://Webprint solution installed on the browser.
Maxsess	The maximum number of sessions that may be spawned from the applet's File->New menu option.	0 = disabled 1 - server license key maximum.
autofit	Allow the print solutions to automatically compress 132 column print onto a portrait page.	OFF ON
autogui	To enable the AutoGUI menu option on the java emulation applet client. This option is only valid for Power User applets.	OFF ON

fontsize	Initial font size of host application text.	7 - 24
script	This parameter specifies a TCL script filename. The file is stored in the scripts sub-directory. This subdirectory appears in the directory that contains the wcd OC://WebConnect server. The file executes when the 3270 or 5250 Java client user enters a ctrl+R key combination at the keyboard.	Default = none. TCL script files are any valid file name with an extension of tcl that are stored in the OC://WebConnect scripts directory followed by any script parameters. The filename and script parameters should be space delimited. Example: “sample1.tcl param1 param2”.
Startup	This parameter specifies a TCL script that executes when the 3270 or 5250 Java client starts a new session. The script file is stored in the scripts sub-directory. This subdirectory appears in the directory that contains the wcd OC://WebConnect server.	Default = none. TCL script files are any valid file name with an extension of tcl that are stored in the OC://WebConnect scripts directory followed by any script parameters. The filename and script parameters should be space delimited. Example: “sample1.tcl param1 param2”.
cipher	The cipher suite to be used for SSL authentication and/or encryption.	See Table 5 SSL cipher suite values.
certfpsv	SLL certificate generated by the OC://WebConnect Server.	Is dymanically generated can’t be used in static html.
GatewayName	TCP/IP Host name or IP address for the TN server or gateway to which the session should connect. This value will override the session setting for Host.	Any valid tcp/ip host name or ip address.
gatewayPort	The TCP/IP port on which the TN server or gateway is listening for emulation traffic.	Default=23.
GatewayResource	The TN server or gateway device name. Possibly a LU name, Pool name, etc.	Any valid LUName, Device Name, or Pool Name.

Applet and applet package file names and purpose

Table 1 - Applet Archive and cabbase values for applets without SSL

Netscape

APPLETS WITHOUT SSL	NETSCAPE 3.X	NETSCAPE 4.X
Ultralite		
3270,5250,VT	Webconnect.zip	Webconnect.zip
3287	Webconnect3287.zip	Webconnect3287.zip
Enhanced		
3270	N/A	ns-E3270T.jar
5250	N/A	ns-E5250T.jar
VT	N/A	ns-EVTT.jar
3287	N/A	ns-E3287.jar
Power User		
3270,5250,VT	N/A	ns-Emus.jar
3287	N/A	ns-E3287.jar
GUI Configurator	N/A	ns-Config.jar

Internet Explorer and Hot Java

APPLETS WITHOUT SSL	INTERNET EXPLORER 3.X	INTERNET EXPLORER 4.X	HOTJAVA 1.X
Ultralite			
3270,5250,VT	Webconnect.cab	Webconnect.cab	Webconnect.jar
3287	Webconnect3287.cab	Webconnect3287.cab	Webconnect3287.jar
Enhanced			
3270	N/A	E3270T.cab	E3270T.jar
5250	N/A	E5250T.cab	E5250T.jar
VT	N/A	EVTT.cab	EVTT.jar
3287	N/A	E3287.cab	E3287.jar
Power User			
3270,5250,VT	N/A	Emus.cab	Emus.jar
3287	N/A	E3287.cab	E3287.jar
GUI Configurator	N/A	Config.cab	Config.jar

Table 2 - Applet Archive and cabbase values for Applets with SSL

Netscape:

APPLETS WITH SSL	NETSCAPE 3.X	NETSCAPE 4.X
Ultralite		
3270,5250,VT	N/A	N/A
3287	N/A	N/A
Enhanced		
3270	N/A	ns-ssl-E3270T.jar
5250	N/A	ns-ssl-E5250T.jar
VT	N/A	ns-ssl-EVTT.jar
3287	N/A	ns-ssl-E3287.jar
Power User		
3270,5250,VT	N/A	ns-ssl-Emus.jar
3287	N/A	ns-ssl-E3287.jar
GUI Configurator	N/A	ns-ssl-Config.jar

Internet Explorer and HotJava:

APPLETS WITH SSL	INTERNET EXPLORER 3.X	INTERNET EXPLORER 4.X	HOTJAVA 1.X
Ultralite			
3270,5250,VT	N/A	N/A	N/A
3287	N/A	N/A	N/A
Enhanced			
3270	N/A	ssl-E3270T.cab	ssl-E3270T.jar
5250	N/A	ssl-E5250T.cab	ssl-E5250T.jar
VT	N/A	ssl-EVTT.cab	ssl-EVTT.jar
3287	N/A	ssl-E3287.cab	ssl-E3287.jar
Power User			
3270,5250,VT	N/A	ssl-Emus.cab	ssl-Emus.jar
3287	N/A	ssl-E3287.cab	ssl-E3287.jar
GUI Configurator	N/A	ssl-Config.cab	ssl-Config.jar

Table 3 - Code Values

APPLETS	NETSCAPE 3.X OR INTERNET EXPLORER 3.X	NETSCAPE 4.X, INTERNET EXPLORER 4.X, HOTJAVA 1.X
Ultralite		
3270,5250,VT	WebConnect.class	WebConnect.class
3287	Webconnect3287.class	Webconnect3287.class
Enhanced		
3270	N/A	COM.oc.webconnect.client.WebConnect3270Thin.class
5250	N/A	COM.oc.webconnect.client.WebConnect5250Thin.class
VT	N/A	COM.oc.webconnect.client.WebConnectVTThin.class
3287	N/A	COM.oc.webconnect.client.WebConnect3287Thin.class
Power User		
3270,5250,VT	N/A	COM.oc.webconnect.client.WebConnectFat.class
3287	N/A	COM.oc.webconnect.client.WebConnect3287.class
GUI Configurator	N/A	COM.oc.webconnect.client.gui.config.ConfigFrame.class

Table 4 - Client Language Applet Tag Values

CLIENT LANGUAGE	HTML APPLET TAG VALUE
Swiss German	de_CH
German	de_DE
US English	en_US
British English	en_GB
Castillian Spanish	es_ES
French	fr_FR
Italian	it_IT
Japanese	ja_JP
Korean	ko_KR
Dutch	nl_NL
Norwegian	no_NO
Brazilian Portuguese	pt_BR
Turkish	tr_TR
Chinese(ROC)	zh_tw
Chinese(China)	zh_CN

Table 5 - SSL Cipher Suite Applet tag parameter values

CIPHER	APPLET TAG PARAMETER VALUE
NULL	NULL or 0000
40-bit DES w/SHA-1 Msg Authentication	0008
56-bit DES w/SHA-1 Msg Authentication	0009
Triple DES w/SHA-1 Msg Authentication	000A
40-bit RC4 w/MD5 Msg Authentication	0017
128-bit RC4 w/MD5 Msg Authentication	0018
128-bit RC4 w/SHA-1 Msg Authentication	0005
MD5 Msg Authentication (No Encryption)	0001
SHA-1 Msg Authentication (No Encryption)	0002

Using the cgiinfo interface to generate an OC://WebConnect Applet Tag

The OC://WebConnect's CGI-BIN program, *cgiinfo*, can be used to retrieve html applet tag data from the OC://WebConnect Emulation server and start a session.

Cgiinfo outputs all the applet parameters necessary to match the html applet tag data to the browser and browser version with the most up to date configuration data collected from the end user and OC://WebConnect HTML files. The use of cgiinfo is required for using OC://WebConnect's implementation of SSL and client token authorization.

The CGI-BIN parameters passed from the HTML, through the CGI-BIN, to OC://WebConnect use a name=value format. Following is a list of possible parameters passed from HTML templates to the CGI-BIN for communication with OC://WebConnect to retrieve requested information.

HTML Macros Passed to cgiinfo

To dynamically create HTML which include up to date applet parameter OC://WebConnect provides a CGI-BIN interface called cgiinfo. This allows an HTTP server to query the OC://WebConnect Emulation server for the latest configuration information. With this information the applet tag portion of the HTML is generated dynamically.

The macros/parameters are passed to the OC://WebConnect Emulation server through the Admin port. Cgiinfo is used by the default *Index.html* and other HTML templates to retrieve information to construct an applet tage for an HTTP server. The HTTP server may be the OC://WebConnect HTTP server "wsd" or a third party server.

The input parameters include data which is required to contact the OC://WebConnect Emulation server, specific session file which should be used to generate the applet tag, options that are not set in the session file and information for outputing the necessary HTML data.

Html example of Cgiinfo use:

<FORM ACTION="/cgi/cgiinfo" METHOD=POST TARGET="xxxxx">
<INPUT TYPE="hidden" NAME="parameter namet" VALUE="xxxx">
</FORM>

Input parameters for cgiinfo:

PARAMETER	DESCRIPTION	VALUES
hostname	Indicates the host name where OC://WebConnect Emulation server, "wcd", is running..	Any valid Domain Name Server hostname or tcp ip address.
portnum	Indicates the administration port for OC://WebConnect emulation server.	Any valid tcp port number. Default = 4270
command	Indicates the request command.	**Start** = Start a session
type	Indicates the session type to used.	**Lite** **enhance** **power**
ssl	Select ON or OFF to indicate if you are using Secure Socket Layer (SSL) protocol.	ON OFF
count	Indicates the number of sessions that will be started when the applet is loaded.	Default = 1 0 indicates that no sessions will autostart.
sfile	Indicates the session file to be used. The session file should correspond to the emulation **type**.	**Example:** def3270.ses Valid filenames are any .ses files stored in the OC://WebConnect cfgdir/ses directory.
http	Indicates the HTTP port to be used for Ultra Lite sessions.	Any valid tcp port.
mode	Indicates the mode of operation. Options are:	Cooked = Perform HTML macro expansion.
HTMLfile	Indicates the HTML file name to be used for macro substitution. The path to this file should be relative to the home directory of the "cgiinfo" process. The OC://WebConnect by default uses the tclient.html file. See the tclient.html file for example or modify the existing tclient.html file. The cgiinfo process inserts the applet tag data into the area of the html file specified for applet tag macro substitution by the following string |applpara| The combined html code is	Any valid html file stored on the same system as the cgiinfo process. Should have the following section for macro substitution: <!----macro begin------> |applpara| <!----macro begin----- >

PARAMETER	DESCRIPTION	VALUES
	then sent to the browser to start an applet. The file may contain any valid HTML commands but must contain the macro substitution section.	
Print	Indicates the print implementation chosen.	**JDK** = Use the JDK1.1 print solution, **JS** = Use the Javascript print solution, **OCWP** = Use the OC://WebPrint solution. **None** = Use no print solution and disable client print options. **Default** = Use the default print solution specified in the session file.
Outfile	Indicates the file name to be used for output.	Optional. Any valid filename.

Example HTML code that uses of cgiinfo:

```
<!-- WebConnect HTML -->
 <HTML>
<HEAD>
<TITLE>OC://WebConnect - Sessions</TITLE>
</HEAD>
<BODY BACKGROUND="/images/whiteroc.gif" LINK="#281860" VLINK="#281860">
 <FORM ACTION="/cgi/cgiinfo" METHOD=POST TARGET="Footer">
<INPUT TYPE="hidden" NAME="cmd" VALUE="start">
<INPUT TYPE="hidden" NAME="host" VALUE="host1.oc.com">
<INPUT TYPE="hidden" NAME="type" VALUE="lite">
<INPUT TYPE="hidden" NAME="sfile" VALUE="def3270">
<INPUT TYPE="hidden" NAME="port" VALUE="4270">
<INPUT TYPE="hidden" NAME="mode" VALUE="cooked">
<INPUT TYPE="hidden" NAME="html" VALUE="html/tclient.html">
<INPUT TYPE="hidden" NAME="http" VALUE="2080">
<INPUT TYPE="hidden" NAME="count" VALUE="1">
</FORM>
</BODY>
</HTML>
```

Output from use of cgiinfo:

Cgiinfo will output the HTML with the html file specified in the cgiinfo *html* parameter and the applet tag generated.

The applet tag will be inserted into the html in the *applpara* section of the original html file. The *applpara* should be specified as follows:

<!---macro begin---------->
|applpara|
<!---macro end------------>

Note:
If you are using the UltraLite applet, OC://WebConnect and the HTTP/web server must reside on the same machine.

Working with a Third Party HTTP Server

The OC://WebConnect HTTP server functionality can be provided by a 3rd party HTTP server. The ability for a 3rd party to process html pages for an end user's and deliver a java emulation session can be easily accomplished fairly easily. Once the java emulation session has been downloaded, the HTTP server is no longer being used. A direct persistent connection is used for the emulation traffic between the emulation clients and the OC://WebConnect emulation server.

There are two general ways that OC://WebConnect can work with a 3rd party HTTP server. The 3rd party server can use the OC://WebConnect cgiinfo to communicate with the OC://WebConnect server to obtain the latest configuration data and dynamic applet tags or static HTML with static applet tags can be written and maintained to provide the download of the emulation client applets to the browser.

Because there are many 3rd party HTTP Webservers which all work differently. This section contains general instructions for working with 3rd party HTTP Webservers.

Using static html code

Most commercial web servers provide a configuration utility that allows alternate document directories to be configurated.

1. Use the Webserver configuration utility to create an alternate document directory such as /wc. Point that alternate directory to the OC://WebConnect html directory (default is /wc/html).

   If this is configured correctly the 3rd party Webserver should be able to access the OC://WebConnect html directory by using the URL:

   http://host1.oc.com/wc

2. Modify the **CODEBASE** and **htmlport** values specified in the applet tag section of the HTML file used to start applets to the hostname plus the 3rd party server /wc directory.

   Example:

   <applet CODEBASE=http://host1.oc.com/wc archive-WebConnect.zip code=WebConnect.class. width=150 height=25>

   <param name=htmlport value=/wc>

2. Moving the java emulation client files to the 3rd party http server directory may be necessary.
3. Ultralite applets require the 3rd party server be on the same machine as the OC://WebConnect Emulation server. This is a limitation based upon the security model provided with JDK 1.0 java enabled browsers.
4. A text file called wcJStrings.txt provides the text for the applet menus and dialogs. This file is available for each client language supported. If using Ultralite applets this file will need to be copied to the 3rd party directory. The file can be located in the OC://WebConnect nls directory under a specific langauge directory.

Customizing the Client Interface

The ability to customize the emulation client applet interface can often be accomplished by choosing the right applet type. If the need to customize the applet is greater, OpenVista provides the ability to make extensive changes and even replace the applet user interface

Some of the need to customize the emulation client applet interface is solved with the availability of the different applet types and individual features like AutoGUI. Ultralite is available for the low end user, Enhanced applets are available for the user need more functionality and Power User is for those users that need file transfer and AutoGUI functionality.

OpenVista is a Java Integrated Development Environment with both visual and simple API access to the OC://WebConnect Java emulation applets. OpenVista provides both the ability to make minor changes to many of the user interface features of the standard applets and more importantly provides the capability to develop a completely different and custom emulation interface, by allowing the development of new Java applets using OC://WebConnect Java code to provide the emulation engine and connectivity to the OC://WebConnect server.

More Information

- For more information about the different emulation client applet types see *Chapter 11 Emulation Client Interface and Features* in the guide.
- For more information about OpenVista see the OpenVista User's Guide provided online with the OpenVista.

Chapter 13: TCL Scripting Extensions

Customizing Client Access to Host Applications

OC://WebConnect incorporates the TCL embeddable scripting language from Sun Microsystems. Using the standard TCL scripting language along with several extensions provided with OC://WebConnect, you can customize client access to 3270 and 5250 host applications. Arguments may be passed to the TCL script by supplying the arguments, following the script name as part of the applet parameters.

Example:

```
<param name="script" value="script.tcl arg1 arg2 arg3">
```

beep(3WC) Extension

Name

beep - play the "beep" audio file.

Synopsis

beep

Description

beep instructs the 3270 Java client software to play the default "beep" file. The default "beep" file is specified in the HTML file which downloads the 3270 Java Applet.

The audio file, "beep.au" is located in the html sub-directory of the directory where the OC://WebConnect server resides in the default "beep" file.

Return Value

Upon successful completion, **beep** returns **R_OKAY**. Otherwise, an error is returned which passes to **query error message** for more information.

Errors

Upon failure, **beep** returns one of the following error (s):

[ENOFILE] The default "beep" file does not exist.

Example:

```
set errno [beep]
if { $errno != "R_OKAY" } {
puts "\"beep\" failed, [query error message $errno]"
exit 1
}
```

copy(3WC) Extension

Name

copy - copy data to/from the screen buffer.

Synopsis

copy to [session *ID*] [position] *position* [text] *string*

copy from [session *ID*] [position] *position* [length] *length*

Description

copy copies data to or from the screen buffer.

ID specifies the session ID of a host session. If session *ID* is omitted, copy copies data to or from the screen buffer associated with the default session. Set the default session ID with the default command.

position specifies the screen position where the data is copied to or from. Specify position as an offset into the screen buffer or by row and column as follows:

row *row* column *column*

The row and column keywords are required if the *position* is specified by row and column instead of screen buffer offset value.

string specifies the data to be copied to the screen buffer.

length specifies the maximum length of the data copied from the screen buffer.

Return Value

Upon successful completion, **copy** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

In addition, **copy** from returns the length of the returned data, followed by the requested data. The **TCL** list returned by **copy from** has the following format:

Element #	Description
0	the return code.
1	the length of the requested data.
2	the requested data.

Retrieve the elements of the returned list individually with the **lindex** command.

Errors

Upon failure, copy returns one of the following error (s):

[EINVALID] The specified *position* is invalid.

[ENOSESSION] The specified session *ID* or the default session ID is invalid.

Example:

```
set data [copy from position 234 length 15]
set errno [lindex $data 0]
if { $errno != "R_OKAY" } {
puts "\"copy from\" failed, [query error message $errno]"
exit 1
```

```
}
puts "[lindex $data 1] bytes successfully copied ..."
puts "[lindex $data 2]"
```

default(3WC) Extension

Name

default - set OC://WebConnect defaults.

Synopsis

default [parameter [=] value] [...]

Description

default allows you to change OC://WebConnect defaults.

parameter specifies the OC://WebConnect parameter to change. User-modifiable parameters are described in the following table:

Parameter	Value	Description
keyboard	**locked** **unlocked**	lock the 3270 Java client's keyboard unlock the 3270 Java client's keyboard
session	*ID*	default session ID
timeout	*minutes* **minute[s]** *seconds* **second[s]**	default **wait** timeout specified in minutes and seconds

If *parameter* is omitted, **default** returns a list of user modifiable parameters along with the current value of each parameter.

Return Value

Upon successful completion, **default** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

In addition, **default** returns a list of parameter/value lists containing the previous value of each parameter which was modified. If no parameters were modified, the name and current value of all user modifiable parameters returns.

Each parameter/value list contains the following elements:

Element #	Description
0	parameter name.
1	parameter value.

The individual parameters return values as follows:

Parameter	Value
keyboard	**locked** or **unlocked**
session	The session ID.
Timeout	The timeout value in seconds

Retrieve the elements of the returned list individually with the **lindex** command.

Errors

Upon failure, default returns one of the following error (s):

[EINVALID] An invalid parameter was specified.

Example:

```
set data [default timeout 1 minute 30 seconds keyboard locked]
set errno [lindex $data 0]
if { $errno != "R_OKAY" } {
        puts "\"default\" failed, [query error message $errno]"
        exit 1
}
set value [lindex $data 1]
puts "The previous timeout was [lindex $value 1] seconds."
set value [lindex $data 2]
puts "The keyboard was [lindex $value 1]."
```

move(3WC) Extension

Name

move - moves the cursor to a new position.

Synopsis

move [cursor] [to] [session *ID*] [position] *position*

Description

move moves the cursor to the specified position.

ID specifies the session ID of a host session. If **session** *ID* is omitted, **move** moves the cursor associated with the default session. The default session ID may be set with the **default** command.

position specifies to which the screen position to move the cursor. Specify *position* as an offset into the screen buffer or by row and column as follows:

row *row* **column** *column*

The **row** and **column** keywords are required if the position is specified by row and column.

Return Value

Upon successful completion, **move** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

Errors

Upon failure, move returns one of the following error (s):

[EINVALID] The specified position is invalid.

[ENOSESSION] The specified session ID or the default session ID is invalid.

Example:

```
set result [move cursor to row 10 column 12]
if { $result != "R_OKAY" } {
        puts "\"move\" failed, [query error message $result]"
        exit 1
}
```

query(3WC) Extension

Name

query - return requested information

Synopsis

query cursor [position] [session *ID*]

query error [message] *errno*

query field [session *ID*] [[position] *position* [*modifier*]]

Description

query returns the requested information.

query cursor returns the current cursor position.

session *ID* specifies the session ID of a host session. If **session** *ID* is omitted, **query cursor** returns the current cursor position of the default session. The default session ID can be set with the **default** command.

query error returns a descriptive error message for the error specified by *errno. errno* may be specified as a numerical value or as a mnemonic which was returned by another OC://WebConnect TCL extension.

query field returns information about a specific field.

session *ID* specifies the session ID of a host session. If **session** *ID* is omitted, **query field** returns information for fields associated with the default session. The default session ID can be set with the **default** command.

position specifies the screen position of interest. *position* can be specified as an offset into the screen buffer or by row and column as follows:

row *row* **column** *column*

the **row** and **column** keywords are required if the *position* is specified by row and column. If *position* is omitted, field information is returned for all fields on the current screen.

modifier specifies the field for which the information is to be returned. *modifier* may be any one of the following:

this field
previous field
next field
next protected field
next unprotected field
previous protected field
previous unprotected field

If *modifier* is omitted, field information is returned for "this field" (the field including the screen position specified by *position*).

Return Value

Upon successful completion, **query** returns **R_OKAY**. Otherwise, an error is returned which may be passed to **query error message** for more information.

Note:

query error returns a descriptive error message for the specified error. No other values are returned. In addition, **query** returns the requested information.

The TCL list returned by query has the following format:

Element #	Description
0	the return code.
1	the requested information.

The requested information is returned in various formats depending on the information returned.

query cursor returns the current cursor position as an offset into the screen buffer.

query field returns a TCL list containing the field information for the specified field(s). If *position* is omitted, a field information list is returned for each field on the current screen.

Each field information list contains the following elements:

Element #	Description	Values
0	Field Position	*position*
1	Field Length	*length*
2	Field Type	**Protected** **Unprotected**
3	Data Type	**Numeric** **Alphanumeric**
4	Display Mode	**Normal intensity, pen not detectable** **Normal intensity, pen detectable High intensity, pen detectable** **Non-display, pen not detectable**
5	Field Status	**Modified** **NOT modified**

The elements of the returned list can be retrieved individually with the **lindex** command.

Errors

Upon failure, query returns one of the following error (s):

[EINVALID]	An invalid keyword was specified.
[ENOSESSION]	The specified session ID or the default session ID is invalid.

Example:

```
set data [query cursor position]
set errno [lindex $result 0]
if { $errno != "R_OKAY" } {
        puts "\"query\" failed, [query error message $errno]"
        exit 1
}
puts "Current Cursor Position: [lindex $data 1]"
```

search(3WC) Extension

Name

search - searches the screen buffer for specific data.

Synopsis

search [d*irection*] [for] [text] *string* [from] [session *ID*]

[[position] *position*]

Description

search searches the screen buffer for specific data.

direction specifies the direction in which the search is to proceed. *direction* must be one of the following:

forward
backward
all

If *direction* is omitted, **search** searches forward from the specified position.

string specifies the text string to be searched for.

session *ID* specifies the session ID of a host session. If **session** *ID* is omitted, **search** searches the screen buffer associated with the default session. Set the default session ID with the **default** command.

position specifies the screen position where the search begins. Specify *position* as an offset into the screen buffer or by row and column as follows:

row *row* **column** *column*

the **row** and **column** keywords are required if the *position* is specified by row and column. If *position* is omitted, the search starts at the beginning of the screen buffer (the default position is **1**).

Return Value

Upon successful completion, **search** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

In addition, **search** returns the screen position where *string* is found. The **TCL** list returned by **search** has the following format:

Element #	Description
0	the return code.
1	the screen position where *string* is found

Retrieve the elements of the returned list individually with the **lindex** command.

Errors

Upon failure, search returns one of the following error (s):

[ENOTFOUND] The specified string was not found in the screen buffer, or was not found at the location specified by position.

[ENOSESSION] The specified session ID or the default session ID is invalid.

Example:

```
set data [search forward for text "Hello, world!" from position 0]
set errno [lindex $data 0]
if { $errno != "R_OKAY" } {
   puts "\"search\" failed, [query error message $errno]"
   exit 1
}
puts "Text string found at screen position [lindex $data 1]"
```

sendfile(3WC) Extension

Name

sendfile - sends files to the 3270 Java client.

Synopsis

sendfile [[audio] [file] *audio_file*]

Description

sendfile sends files to the 3270 Java client.

audio_file is the name of an audio file sent to the 3270 Java client. The file plays immediately upon receipt. If the file name ends with the extension "au", the keyword, **audio** can be omitted.

Multiple files can be sent to the 3270 Java client with one **sendfile** command.

Return Value

Upon successful completion, **sendfile** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

Errors

Upon failure, **sendfile** returns one of the following error (s):

[ENOFILE] The specified file does not exist.

Example:

```
set errno [sendfile audio beep.au]
if { $errno != "R_OKAY" } {
        puts "\"sendfile\" failed, [query error message $errno]"
        exit 1
}
```

sendkey(3WC) Extension

Name

sendkey - sends function keys, data, and/or an AID key to the host.

Synopsis

sendkey [**session** *ID*] [**key** *key_code*] [[**text**] *string*] [**aidkey** *AID_Key*]

Description

sendkey sends keystrokes, including attention identifier (AID) keys, to the host.

session *ID* specifies the session ID of a host session. If **session** *ID* is omitted, sendkey sends the specified keystrokes to the host using the default session ID. Set the default session ID with the **default** command.

key_code specifies one or more of the key codes described in the following tables:

3270 Key Codes
5250 Key Codes

string specifies ASCII data to be sent to the host. ASCII mnemonics, representing special function keys described in the tables below, can be embedded directly in a text string. Otherwise, key codes must be preceded by the **key** keyword.

Multiple key codes and text may be interspersed on one **sendkey** command line.

AID_Key specifies one of the 3270 or 5250 AID keys described in the following tables:

3270 AID Keys
5250 AID Keys

> **Note:**
> Only one AID key can be specified on a **sendkey** command line. Everything after the first AID key is discarded. This restriction may be removed in the future.

Return Value

Upon successful completion, **sendkey** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

Errors

Upon failure, sendkey returns one of the following error (s):

[ETRUNCATED] Output was terminated after the first AID key, and all following data was truncated.

[ENOSESSION] The specified session ID or the default session ID is invalid.

Example:

```
set result [sendkey "Hello, world!" aidkey Enter]
if { $result != "R_OKAY" } {
puts "\"sendkey\" failed, [query error message $result]"
exit 1
}
```

3270 Attention Identifier (AID) Keys

Control Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Attn	@A@Q	System Request	@A@H	PA1	@x
Clear	@C	.	.	PA2	@y
Enter	@E	.	.	PA3	@z

Programmable Function Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
PF1	@1	PF7	@7	PF13	@d	PF19	@j
PF2	@2	PF8	@8	PF14	@e	PF20	@k
PF3	@3	PF9	@9	PF15	@f	PF21	@l
PF4	@4	PF10	@a	PF16	@g	PF22	@m
PF5	@5	PF11	@b	PF17	@h	PF23	@n
PF6	@6	PF12	@c	PF18	@i	PF24	@o

5250 Attention Identifier (AID) Keys

Control Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Clear	@C	Help	@H	RollUp	.
Enter	@E	Print	@P	RollDown	.

Function Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
F1	@1	F7	@7	F13	@d	F19	@j
F2	@2	F8	@8	F14	@e	F20	@k
F3	@3	F9	@9	F15	@f	F21	@l
F4	@4	F10	@a	F16	@g	F22	@m
F5	@5	F11	@b	F17	@h	F23	@n
F6	@6	F12	@c	F18	@i	F24	@o

3270 Key Codes

Control Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
CursorSelect	@A@J	EraseEOF	@F	Print	@P
FieldMark	@S@y	EraseInput	@A@F	Reset	@R

Cursor Movement Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
CursorUp	@U	NewLine	@N	Home	@0
CursorDown	@V	Tab	@T	WordLeft	@A@z
CursorLeft	@L	Backtab	@B	WordRight	@A@y
CursorRight	@Z	.	.	.	.

Cursor Attribute Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
AlternateCursor	@$	CursorGr?	.

Edit Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Backspace	@<	Delete	@D	Insert	@I
Dup	@S@x	DeleteWord	@A@D	.	.

Text Attribute Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
HighDefault	.	HighUnderscore	.
HighReverse	.	HighBlink	.

Text Color Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Red	@A@d	Yellow	@A@g	White	@A@j
Pink	@A@e	Blue	@A@h	ResetHostColors	@A@l
Green	@A@f	Turquoise	@A@i	.	.

5250 Key Codes

Control Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Attention	@A@Q	SystemRequest	@A@H
Reset	@R	Test	@A@C

Cursor Movement Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
CursorUp	@U	Return	@N	Home	@0
CursorDown	@V	Tab	@T	DoubleLeft	.
CursorLeft	@L	Backtab	@B	DoubleRight	.
CursorRight	@Z	.	.	.	.

Field Navigation Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
FieldPlus	@A@+	FieldMinus	@A@-	FieldExit	@A@E

Edit Keys:

Key Code	ASCII Mnemonic	Key Code	ASCII Mnemonic
Backspace	@<	Delete	@D
Dup	@S@x	Insert	@I

Text Assist Keys:

Key Code	Command Description	Key Code	Command Description
AltA	Insert Symbols	AltUp	Top of Page
AltB	Begin Bold	AltDown	Bottom of Page
AltC	Center Text	AltLeft	Beginning of Line
AltD	Next Text Column	AltRight	End of Line
AltH	Half Index Down	AltFieldPlus	Carrier Return
AltJ	End Bold/Underscore	AltFieldMinus	Carrier Return
AltN	Stop Code Advance	AltFieldExit	Carrier Return
AltP	Page End	AltSpace	Required Space
AltS	Stop Code Function	AltTab	Required Tab
AltU	Begin Underscore	AltBacktab	Shifted Tab
AltW	Word Underscore	.	.
AltY	Half Index Up	.	.

wait(3WC) Extension

Name

wait - suspends the TCL script.

Synopsis

wait [*timeout*] [for] keyboard [unlock] [session *ID*]

wait [*timeout*] [for] screen [update] [session *ID*]

wait [*timeout*] [for] [text] *string* [at] [screen *ID*]

[[position] *position*]

Description

wait suspends the TCL script until the specified event occurs. If the specified event has already occurred, **wait** returns immediately.

wait for keyboard returns when the host unlocks the keyboard after receiving an AID key.

wait for screen returns when the screen buffer is updated by the host.

wait for *string* returns when *string* is found at the specified position in the screen buffer. If position is omitted, **wait** returns when *string* is found anywhere in the screen buffer.

timeout specifies a length of time after which **wait** returns regardless of whether or not the specified event has occurred. If the specified event does not occur within the specified time, wait returns to the caller with an appropriate return code. Specify *timeout* as follows:

[**timeout**] *minutes* **minute[s]** *seconds* **second[s]**

string specifies the text string for which you are searching.

session *ID* specifies the session ID of a host session. If **session** *ID* is omitted, **wait** waits for the specified event to occur for the default session. Set the default session ID with the **default** command.

position specifies the screen position (for example, wait returns when *string* appears at the specified screen position or when the command times out.) Specify *position* as an offset into the screen buffer or by row and column as follows:

row *row* **column** *column*

The **row** and **column** keywords are required if the position is specified by row and column.

Return Value

Upon successful completion, **wait** returns **R_OKAY**. Otherwise, an error returns that passes to **query error message** for more information.

In addition, **wait** returns an offset into the screen buffer where the specified data begins. If no data was specified, an offset of "0" will be returned. The **TCL** list returned by **wait** has the following format:

Code	Description
0	the return code.
1	an offset into the screen buffer where *string* begins.

Retrieve the elements of the returned list individually with the **lindex** command.

Errors

Upon failure, **wait** returns one of the following error (s):

[ETIMEDOUT] The specified timeout period elapsed before the waited for event occurred.

[ENOSESSION] The specified session ID or the default session ID is invalid.

Example:

```
set data \
    [wait 1 minute 15 seconds for text "Hello, world!" at position
100]
set errno [lindex $data 0]
if { $errno == "ETIMEDOUT" } {
     puts "\"wait\" timed out!"
     exit 0
}
if { $errno != "R_OKAY" } {
     puts "\"wait\" failed, [query error message $errno]"
     exit 1
}
set offset [lindex $data 1]
puts "Text string found at screen position $offset."
```

Chapter 14: Transferring Data Files

Using IND$FILE Transfer

OC://WebConnect transfers files between a Java client and an SNA host application using the standard IND$FILE transfer protocol. A variety of networking needs including centralized data backups and data warehousing through an SNA host may make use of this functionality. Because SNA host files use different file formats than OC://WebConnect files and Java client files, use the appropriate options for converting files to the receiving host's file format during transfer. The format conversion allows the receiving host's applications to use the file.

The following table lists the SNA hosts and SNA applications used for transferring files (including the IBM program number and operating system for each application).

Application Program	Program Number	Operating system
3270 PC File Transfer for CICS/VS	5798-DQH	VS
3270 PC File Transfer for TSO	5665-311	MVS
3270 PC File Transfer for VM/CMS	5664-281	VM

Notes:

- OC://WebConnect supports only the DFT (Distributed Function Terminal) file transfer mode.
- IND$FILE transfer is available only when using the OC://WebConnect Power User Java applet.
- You must be familiar with the file transfer application program you want to use.

Sending and Receiving CICS/VS Files

OC://WebConnect provides file transfer between a Java client and the Customer Information Control System/Virtual Storage (CICS/VS) SNA application. You can use the Java client's **Transfers** menu for transferring files to suit your needs.

1. Make sure the Java client is connected to a desired SNA host and CICS application.
2. Select **File Transfer** from the **File** Menu.
3. Select **INDFile** from the **File Transfer** menu.
4. Select either **Send to Host** or **Receive from Host** from the **INDFile** submenu.
5. Select the **CICS** option.
6. The appropriate file transfer window displays with the hostname of the active session.
7. Click the **Local File** button to search for the peer's file. A file selection window displays. The procedures for searching for the peer file name vary with your system. After you select the peer file, the name displays in the text field to the right of the **Local File** button.
8. Click the **UNIX Format** button if you want to transfer files in UNIX format. This allows line separators to be converted to carriage return and line feed pairs during a Send operation. During a Receive operation, carriage return and line feed pairs are converted to line separators.
9. Type a host filename in the **CICS File Name** field in the **File Attributes** box.

Note:
The CICS filename can be a program name, a transaction identification, or identification selected by the CICS/VS application programmer. If the filename does not exist, the CICS/VS application automatically creates it. The filename can be entered as 1–8 characters in length. The character in position 1 must be entered as a letter; characters in positions 2–8 can be entered as letters or digits.

10. Type comments about the file being transferred in the **Comments** field in the **File Attributes** box. The comments are automatically installed in the first record of the CICS/VS host file.

11. Select a file type from the **Transfer Options** box to configure the way the file's contents are treated during the transfer process. Values are:

ASCII	This option instructs the SNA host to translate data between the EBCDIC and ASCII character formats. You can use this option for translating ASCII formatted files, such as text edit files or print files. You should not use the ASCII option for transferring binary data (such as output data from a database program) or object code files (such as C compiler object code).
Binary	This option instructs the SNA host to perform no character translation. The option can be used to transfer encrypted data, compiled programs, and other data that is unreadable.

Caution:

Do not click the **UNIX Format** checkbox when activating the **Binary** option or the binary data becomes corrupted.

Notes:

- If you do not specify the **CRLF** option in the Send mode, the SNA host disregards the local file's line separators.
- You should not use the CRLF option for transferring binary data (such as output data from a database program) or object code files (such as C compiler object code)
- You can click the **UNIX Format** checkbox for ASCII file transfers. This allows line separators to be converted to carriage return and line feed pairs during a Send operation. During a Receive operation, carriage return and line feed pairs are converted to line separators.
- Invoking the **No CRLF** option in the Receive dialog box instructs the CICS/VS host to copy the file unaltered to the appropriate TCP/IP host. The **No CRLF** option can be used to transfer encrypted data, compiled programs, and other data that is unreadable.

12. Select **CRLF** or **No CRLF**.
13. Click the **Cancel** button to disregard your settings.
    Click the **OK** button to begin file transfer.

Sending and Receiving TSO Files

OC://WebConnect's features allow you to transfer files between a Java client and the Time Sharing Option (TSO) SNA application. You can use the Java client's **Transfers** menu for transferring files to suit your needs.

To transfer files between your directory system and a TSO application

1. Make sure the Java client is connected to a desired SNA host and TSO application.
2. Select **File Transfer** from the **File** menu.
3. Select **INDFile** from the **File Transfer** menu.
4. Select either **Send to Host** or **Receive from Host** from the **INDFile** submenu.
5. Select the **TSO** option.
6. Click the **Local File** button to search for the peer's file. A file selection window displays. The procedures for searching for the peer file name vary with your system. After you select the peer file, the name displays in the text field to the right of the **Local File** button.
7. Click the **UNIX Format** button if you want to transfer files in UNIX format. This allows line separators to be converted to carriage return and line feed pairs during a Send operation. During a Receive operation, carriage return and line feed pairs are converted to line separators.
8. Type a dataset name in the **TSO Data Set Name** field.
9. Type a member name in the **Member Name** field.

Notes:

- The TSO host dataset name must conform to IBM naming conventions. You can enter an existing dataset name (stored in your library) or a new dataset name.
- No closing quote displays in the **Member Name** field.

Notes:

- The member name is optional. If entered, the member name should be a member in a partitioned dataset directory.
- When you use the **Send** dialog box to copy a file to a partitioned dataset and include a member name, the partitioned dataset must exist. OC://WebConnect does not create the partitioned dataset.
- The TSO application adds a user ID prefix to the combined dataset and member name. To eliminate the user ID prefix, enclose the dataset and member name in single right quotation marks, such as ’smith.pds2.file1’.

10. Type the appropriate password in the **Password** field. A password is required only if password-protection has been specified for the TSO dataset.
11. Select a transfer option from the **General** box in the **Transfer Options** area. The **Transfer Options** parameters allow you to configure the way the file’s contents are treated during the transfer process. Values are:

Append	This option allows you to append a local file to the end of an SNA host file; otherwise, you can append an SNA host file to a local file. The Append option overrides any other values specified by the LRECL parameter and RECFM options.
ASCII	This option instructs the SNA host to translate data between the EBCDIC and ASCII character formats. You can use this option for translating ASCII formatted files, such as text edit files or print files.
CRLF	This option (by using the Send dialog) instructs the SNA host to replace the local file line separators with SNA record separators. If you use the Receive dialog, the SNA host replaces the SNA record separators with local file line separators. If you do not specify the CRLF option in the Send mode, the SNA host disregards the local file’s line separators.

Note:

When you send a file to a TSO application, the local file’s line separators are replaced with record separators. When a Java client receives a file, the SNA host record separators are replaced with line separators. If you do not specify the CRLF option in the Send mode, the SNA host disregards the local file’s line separators.

Caution:

Do not use the ASCII or CRLF options for binary data (such as output data from a data base program) or object code files (such as C compiler object code).

12. Select a record format parameter for the SNA host file from the **Format** box in the **Transfer Options** area. The **Format** check boxes specify the record format of the SNA host file. This is only valid if you are sending a file. Values are:

Fixed	Indicates that the dataset's records are fixed-length
Variable	Indicates that the dataset's records are variable-length
Undefined	Indicates that the dataset contains undefined record lengths
None	Indicates no record format is to be used

13. Select the **Specify Space Parameters** checkbox in the **Allocation Units** area to set the amount of space to be allocated for a new dataset. The **Blocks**, **Tracks**, and **Cylinders** radio buttons are enabled. If the Space toggle button is not activated, the TSO application uses the Blocks parameter's default value. This is only valid if you are sending a file. Values are:

Blocks	This parameter represents the smallest storage entity to be used.
Tracks	This parameter represents the middle-sized storage entity to be used.
Cylinders	This parameter represents the largest storage entity to be used.
Primary	This parameter lets you specify the primary allocation for the **Blocks** parameter.
Increment	This parameter lets you specify the increment allocation for the Blocks parameter.

14. Type a size value in the **BLKSIZE** field in the **Transfer Options** area. You can enter the data block size of a TSO dataset. The variable you enter represents a data block's number of bytes. The default value is 80. This is only valid if you are sending a file.

Notes:

You might be replacing a file or appending one file to another file. If so, the TSO application uses the existing file's block size information—the BLKSIZE parameter is not used. In addition, the TSO application uses the file transfer operation's default record length if the BLKSIZE parameter is not activated.

15. Type a logical record length value of the SNA host file in the **LRECL** field in the **Transfer Options** area. The parameter value represents the number of characters for each record. If the parameter is not entered, the record length is determined by the file transfer operation. For new files, the parameter's default value is 80.

If you are replacing a file or appending information to a file, the characteristics of the existing file are used. If you are transferring variable length records, the parameter represents the maximum record size. If you do not send a record of the maximum operating system size, the parameter's value becomes the longest record sent. This is only valid if you are sending a file.

16. Click the **Cancel** button to disregard your settings.
    Click the **OK** button to begin file transfer.

Sending and Receiving CMS/VM Files

OC://WebConnect's features allow you transfer files between a Java client and the Virtual Machine/Conversational Monitor System (VM/CMS) SNA application. You can use the Java client's **Transfers** menu for transferring files to suit your needs.
To transfer files between your directory system and a VM/CMS application

1. Make sure OC://WebConnect is connected to a desired SNA host and VM/CMS application.
2. Select **File Transfer** from the **File** menu.
3. Select **INDFile** from the **File Transfer** menu.
4. Select either **Send to Host** or **Receive from Host** from the **INDFile** submenu.
5. Select the **CMS/VM** option.
6. Click the **Local File** button to search for the peer's file. A file selection window displays. The procedures for searching for the peer file name vary with your system. After you select the peer file, the name displays in the text field to the right of the **Local File** button.
7. Click the **UNIX Format** button if you want to transfer files in UNIX format. This allows line separators to be converted to carriage return and line feed pairs during a Send operation. During a Receive operation, carriage return and line feed pairs are converted to line separators.
8. Type a host filename in the **VM File Name** field in the **File Attributes** area. The VM filename can be from 1–8 characters in length.

Note:
The VM/CMS application automatically creates the receiving host's filename if a filename does not exist.

9. Type the appropriate file type in the **VM Filetype** field. The filetype parameter identifies the VM/CMS disk file type.
10. Type an appropriate value in the VM Filemode text box. The filemode parameter identifies the VM/CMS disk file mode. If you do not enter a filemode parameter, the VM/CMS application uses the A1 default value.

11. Select a transfer option from the **General** box in the **Transfer Options** area. The **Transfer Options** parameters allow you to configure the way the file's contents are treated during the transfer process. Values are:

APPEND	This option allows you to append a local file to the end of an SNA host file; otherwise, you can append an SNA host file to a local file. The Append option overrides any other values specified by the Logical Record Size parameter and Record Format options.
ASCII	This option instructs the SNA host to translate data between the EBCDIC and ASCII character formats. You can use this option for translating ASCII formatted files, such as text edit files or print files.
CRLF	This option (by using the Send dialog) instructs the SNA host to replace the local file line separators with SNA record separators. If you use the Receive dialog, the SNA host replaces the SNA record separators with local file line separators. If you do not specify the CRLF option in the Send mode, the SNA host disregards the local file's line separators.

Note:

If you do not activate the Append option in the Receive dialog box, the SNA host file replaces the Java client file. If you do not activate the Append option in the Send dialog box, the TCP/IP host file replaces the SNA host file.

Caution:

Do not use the ASCII or CRLF options for binary data (such as output data from a data base program) or object code files (such as C compiler object code). If the CRLF option is activated for a binary file transfer, unexpected results are produced when the file is used.

12. Select a record format parameter for the SNA host file from the **Format** box in the **Transfer Options** area. The **Format** check boxes specify the record format of the SNA host file. This is only valid if you are sending a file. Values are:

Fixed	Indicates that the dataset's records are fixed-length
Variable	Indicates that the dataset's records are variable-length
None	Indicates no record format is selected

13. Type a logical record length of the SNA host file in the **Logical Record Size** field in the **Transfer Options** area. The parameter value represents the number of characters for each record. If the parameter is not entered, the record length is determined by the file transfer operation. For new files, the parameter 's default value is 80. This is only valid if you are sending a file.

If you are replacing a file or appending information to a file, the characteristics of the existing file are used. If you are transferring variable length records, the parameter represents the maximum record size. If you do not send a record of the maximum operating system size, the parameter's value becomes the longest record sent.

14. Click the **Cancel** button to disregard your settings.
    Click the **OK** button to begin file transfer.

Chapter 15: Security Overview

Overview

OC://WebConnect provides a number of advanced features which can be utilized to securely deploy sessions on Intranets or the public Internet. These features can be utilized in various combinations to enable varying levels of security.

The main areas of concern dealing with security in OC://WebConnect are:

- data privacy (encryption)
- data integrity (message authentication)
- Firewalls and network topology
- Authentication of client to server
- Authentication of server to client

It's important to establish a desired overall network topology and security requirements criteria before starting to configure OC://WebConnect. For example, will a firewall be used and will sessions on the internal network require encryption. OC://WebConnect can be designed into a network topology with or without firewalls and also has the capability of running encrypted and non-encrypted sessions simultaneously. Knowing the security/topology requirements before you go on allows for a simpler OC://WebConnect installation and customization. Security administration for OC://WebConnect is done entirely at the server. No client administration is required.

If *server authentication* is a requirement, typically a concern when deploying over a public network, then the Secure Socket Layer (SSL) option should be used. Server authentication in SSL is provided via X.509 certificates. SSL also provides *Message Authentication* to prevent message tampering. There are many more benefits to using SSL which will be discussed later.
Client authentication is provided through a token passing mechanism. This mechanism relies on the customer's existing security between the browser and web server, providing a method to re-authenticate a client before granting sessions.

Two types of *message encryption* between the OC://WebConnect server and the client are provided. They may not be used simultaneously:

SSL	NETSCAPE COMMUNICATIONS, INC.	A CHOICE OF 6 CIPHER SUITES (ALSO INCLUDES THE RC4 ALGORITHM)
		Client/Server encryption algorithm negotiations
RC4	RSA Data Security, Inc.	40 bit encryption key*
		128 bit encryption key (128-bit encryption is Export Restricted, US Only)

Note: OC://WebConnect qualified for ECCN 5D002 general license exemption TSU (Technology and Software Unrestricted) per mass market notes. This ruling indicates that individual licenses are only required for Cuba, Iran, Iraq, Libya, North Korea, Sudan and Syria.

Firewalls and network topology

The multi-tier security approach in OC://WebConnect is designed to complement existing internet security, not replace firewalls and other security devices. Due to the many types of firewalls which can be deployed in various configurations, we cannot discuss all the possibilities here. In general, OC://Webconnect will work with Proxy Firewalls when a hole is punched in the firewall to allow the connection.

Limitations:

- OC://WebConnect will not work with Masquerade Firewalls.
- If OC://WebConnect is behind a firewall and the client is behind a different (Proxy) firewall, this configuration will not work properly.

The Admin port must be protected to operate the SSL and client token authentication features securely.

The ultimate protection is to have the Web server and OC://WebConnect on the same machine and to use the localhost for the IP address of the Admin port. In this way messages never leave the machine. The OC://WebConnect session is the focal point of security.

Protecting Host Resources

In OC://WebConnect, the point of access to host resources is through session definitions. For instance, host name, port and LU are all components of a session definition. By using the access protection of the web server you can restrict access to a given session. In addition, by using the *client authentication* feature the users security is propagated into OC://WebConnect to protect session access.

For example, the web server could authenticate the user, and then present different choices depending on who the user is. Individual users or groups could be confined to use certain session definitions mapping to specific host resources. By linking the token to the session definition, the authentication and mapping defined on the web server is extended to and enforced by the OC://WebConnect server. SSL cipher suites are set on a per session basis. This allows the configuration of different cipher suites on different sessions. The *SSL Required* option requires the user to use SSL (via secure Java port) when accessing a session with this option set. If the *SSL Required* option is not set, then SSL is optional for the session.

SSL vs. RC4

Two types of encryption are available between the OC://WebConnect Java client and the OC://WebConnect server. For a particular session, only one type of encryption can be used at a time. SSL-enabled applets take longer to download. This is due to the fact that SSL capabilities in today's browsers are not accessible to applets; therefore, SSL libraries must be included and downloaded with the applet. Once downloaded, a session will setup quicker using the SSL option than a session using RC4 with Diffie-Hellman. For keys less than 1024, SSL sessions will also use significantly less server CPU during session startup.

SSL may have varying levels of performance dependent on the particular cipher suite implemented. Cipher suites utilizing RC4 encryption and the MD5 hashing algorithm will yield the highest performance.

The RC4 encryption option allows for a quicker applet download time since it uses a smaller applet, however it takes longer to setup a session (due to the Diffie-Hellman algorithm used for key generation).

Also, the server CPU load for session startup is generally higher. At this time, there is no specific performance data on RC4 vs. SSL for large numbers of clients, however it is a logical assumption that RC4 will run quicker than SSL since SSL adds padding and performs message authentication.
On the Client side, SSL should not be perceptible. On the server, unless the server becomes CPU bound, SSL will not cause a degradation in performance.

Detailed discussions of the OC://WebConnect SSL and RC4 implementations can be found later in this chapter. For additional technical information on SSL and RC4, please refer to the following Web sites:

SSL	http://search.netscape.com/newsref/std/sslref.html
SSL	http://www.netscape.com/ à do a search on "SSL".
RC4	http://www.rsa.com/

SSL is recommended for deploying OC://WebConnect over the public internet. The RC4 w/Diffie-Hellman option is intended more for intranet use.

SSL in OC://WebConnect

OC://WebConnect uses SSL to secure connections between an OC://WebConnect Java client and the OC://WebConnect server without requiring any special configuration on the client machine. This is achieved by leveraging the SSL provided in the customer's web server and browser. That is, security parameters are passed to the OC://WebConnect Java client over the browser-to-web server connection.

When the browser requests an OC://WebConnect session from the web server, a process on the back end of the web server will connect to the OC://WebConnect server and obtain the configuration parameters for the client session. Included in these parameters are the SSL port on the OC://WebConnect server, the cipher suite to be used for the session, and a hash, or fingerprint, of the OC://WebConnect server certificate. This fingerprint is later used to verify the certificate received from the OC://WebConnect server during SSL negotiations, thereby authenticating the OC://WebConnect server.

OC://WebConnect uses an alternate port for SSL connections so that different security measures can be applied to the SSL and non-SSL ports. For instance, an installation of OC://WebConnect could choose to hide the unsecured port behind the corporate firewall but expose the SSL port to internet traffic.

A key pair must be generated for OC://WebConnect. The private key is password protected and used only by OC://WebConnect. An X.509 Certification Authority certifies the public key. The resulting certificate is used by clients to authenticate the server as part of the SSL protocol. Before enabling SSL in OC://Webconnect you should have previously installed a private key and certificate for the server. Please see information on Key Pairs and X.509 Certificates below.

Cipher Suites

SSL defines a Handshake Protocol for negotiating a "Cipher Suite" and allowing the client and server to authenticate each other. The cipher suite specifies the algorithms to be used for peer authentication, data encryption, and message authentication when normal session traffic begins. The actual algorithms defined by a cipher suite are independent of the SSL protocol.
In OC://WebConnect, several popular encryption algorithms, such as DES, Triple DES, and RC4, are supported. The RSA public-key algorithm is used for both key exchange and peer authentication. Secure Hash Algorithm (SHA-1) and MD5 are supported for message authentication.
A separate cipher suite can be selected for each configured session. Cipher suites are set from the security section of session configuration using the GUI or HTML configuration.

Key Pairs and X.509 Certificates

SSL utilizes public-key cryptography for peer authentication and key exchange. OC://WebConnect uses the RSA public-key algorithm for both of these functions. The server's public key is given to the client in a digital certificate (X.509 standard). The client generates a master secret to be used to derive a session key for data encryption. The client then encrypts the master secret with the server's public key and sends it back to the server. Now the server can decrypt the master secret and communicate with the client using the encryption algorithm specified in the negotiated cipher suite. This all requires that a key pair be generated for the server. The private key must be kept secret, only to be used by the server. The public key is given to an X.509 Certification Authority (CA) for certification.

The CA generates a certificate containing the server's name and public key, the CA's name, validity dates, and a serial number for the certificate. Finally, the CA "signs" the certificate with its own private key, so that its authenticity can be verified by anyone in possession of the CA's public key. An SSL client authenticates an SSL server by verifying the signature in the server certificate with the public key of the CA specified in the certificate. For this to work, the client must have ready access to the CA's certificate.

If configured to do so, an SSL server may request a certificate from the client so that the server also may authenticate the client. HTML extensions exist to trigger a browser to generate a key pair, request a certificate, and accept a certificate for installation. This allows a browser to operate with a Web-based CA. Netscape and Microsoft, both support this type of browser configuration under user control.

Whether you choose to use a trusted third party for your CA, or whether you establish a private CA within your company or organization will probably depend on who the targeted users of your system will be and the level of security you require.

If your users will typically be anonymous or outside of your administrative control, or if your security requirements are not stringent, you will probably want to use a certificate issued by a trusted third party. On the other hand, if you want the highest level of security possible, you will probably want to establish a private CA within your company or organization, using third party certification tools, such

as Netscape Certificate Server, Entrust Web CA, or XCERT Sentry CA. Then you can issue your own certificates for servers and clients, and configure them to honor your certificates and only your certificates.

To use SSL securely in OC://WebConnect, you will need to launch the OC://WebConnect Java applets from a secure web server using an SSL-enabled browser. The Java applet will then connect to the Secure Java Port of the OC://WebConnect Server, using the SSL protocol and authentication data passed in over the SSL-protected browser-to-web server connection.

The OC://WebConnect Server must have a key pair and certificate before you can use the SSL feature. These are normally generated during the server installation process, but can be generated later using the configuration utility. Since server authentication data is passed to a Java applet over the secure browser-to-web server connection, it is not necessary for a known CA to issue the OC://WebConnect server certificate. You can choose to allow OC://WebConnect to generate its own certificate, or if you prefer, you can have it generate only a PKCS #10 certificate request to be submitted to your CA. If you choose to use a CA to provide the certificate for the OC://WebConnect server, it will need to be manually installed.

The server certificate should be a base64-encoded, DER-formatted, X.509 certificate, stored in a file called cert.txt in the security subdirectory. This should be a concatenation of the server's certificate, the issuer's certificate, plus any others in the hierarchy if the issuer is not the root CA. They should be ordered server certificate first, root CA certificate last. You may need to cut and paste from two or more files to create cert.txt.

Limitations:

- Limited to Cipher suites provided.
- 128 bit encryption is Export Restricted.
- Cannot be used in conjunction with the RSA RC4 encryption option.

Dependencies:

- For complete security, an SSL enabled Web Browser and Web Server must be used.

Cipher Suites

Non-exportable Cipher Specifications Table

RSA_WITH_RC4_128_SHA

- RSA algorithm for key exchange and peer authentication.
- RC4 128-bit encryption.
- SHA (Secure Hash Algorithm) for message authentication.

RSA_WITH_RC4_128_MD5

- RSA algorithm for key exchange and peer authentication.
- RC4 128-bit encryption.
- MD5 algorithm for message authentication.

RSA_WITH_3DES_EDE_CBC_SHA

- RSA algorithm for key exchange and peer authentication.
- Triple DES encrypt-decrypt-encrypt (EDE) encryption, in cipher block chaining (CBC) mode.
- SHA (Secure Hash Algorithm) for message authentication.

RSA_WITH_DES_CBC_SHA

- RSA algorithm for key exchange and peer authentication.
- DES encryption in cipher block chaining (CBC) mode.
- SHA (Secure Hash Algorithm) for message authentication.

Exportable Cipher Specifications

RSA_EXPORT_WITH_RC4_40_MD5

- RSA algorithm for key exchange and peer authentication.
- RC4 40-bit encryption.
- MD5 algorithm for message authentication.

RSA_EXPORT_WITH_DES40_CBC_SHA

- RSA algorithm for key exchange and peer authentication.
- DES 40-bit encryption in cipher block chaining (CBC) mode.
- SHA (Secure Hash Algorithm) for message authentication.

SSL Protocol with no encryption

RSA_WITH_NULL_MD5

- RSA algorithm for peer authentication.
- MD5 algorithm for message authentication.

RSA_WITH_NULL_SHA

- RSA algorithm for peer authentication.
- SHA algorithm for message authentication.

RC4 Encryption Option

The RC4 option uses the Diffie-Hellman algorithm for key generation at the time a connection is made. The RC4 encryption option and key length is set on a per session basis. Since an inherent limitation of RC4 is that it is susceptible to Man-in-the-middle attacks, this option is best for Intranets.

Limitations:

- Only the 40 bit RC4 encryption is available in the UltraLight applet.
- Cannot be used in conjunction with the SSL encryption option.
- When RC4 is configured for a given session, then it is always required for that session.

Client Authentication (token)

The *token authentication* feature in OC://WebConnect is a user authentication mechanism that leverages the existing security infrastructure of the customer's web server/browser environment. The concept is that the OC://WebConnect server dynamically generates tokens, delivered to the browser along with the OC://WebConnect applet, to allow the applet to gain access back to the OC://WebConnect server. The result is that the OC://WebConnect installation is brought transparently under the umbrella of the customer's existing authentication scheme, incurring no additional administrative overhead. To be 100% effective, SSL must be running between the web server and browser, so that tokens are secure from outside snooping.

How it Works:

An OC://WebConnect user will use his browser to first connect to a home page or logon screen on the web server. The user's HTML will present a button or link to allow the user to request an OC://WebConnect session. OpenConnect provides functional HTML pages with the product that can be used outright or as an example for development of custom HTML.
As a result of clicking this button, the OC://WebConnect CGI-BIN will be invoked. The CGI-BIN will connect to the admin port of the OC://WebConnect server and request the applet parameters. The server will respond with HTML-formatted parameters intended to start up a java applet. Among other things, these parameters include the applet name/location, the name and port of the OC://WebConnect server and a session name used by the server to identify host information and session attributes. If the *token authentication* feature has been enabled in the server, an additional parameter will be supplied to the applet: a token.

This HTML, containing the applet parameters, is passed back through the web server to the browser. The browser requests the server to send down the specified applet, then invokes this applet under the control of the resident JVM, passing in the parameters originating from the OC://WebConnect server. Finally, the applet connects to the OC://WebConnect server, which in turn connects to the host, and the user session commences. If the applet has received a token, it must present this token to the server during the initial handshake (within a configured timeout period), or else the session will be rejected.

It is very important that the Admin port for OC://WebConnect remain behind the firewall, or on a private network, and not be exposed to the Internet or other unsecured network. Tokens would be accessible to anyone on these networks.

In OC://WebConnect, the point of access to host resources is through session definitions. For instance, host name, port and LU are all components of a session definition. Linking the token to a session effectively propagates any access protection present in the customer's HTML to the OC://WebConnect environment.

For example, the web server could authenticate the user, and then present different choices depending on who the user is. Individual users or groups could be confined to use certain session definitions mapping to specific host resources. By linking the token to the session definition, the authentication

and mapping defined on the web server is extended to and enforced by the OC://WebConnect server, thus protecting host resources.

Operation of the Token Authentication feature in OC://WebConnect Server

When the *token authentication* feature is enabled on the server, it is enabled for all sessions on the server. In addition to the on/off switch, the administrator is able to specify a time-to-live value in seconds. The default value is 90 seconds.

The *token authentication* feature uses a pseudo-random number generator powered by the MD5 hashing algorithm to generate the tokens (16 bytes long). A token is generated each time a request is received for applet parameters over the Admin port. The server keeps a copy of the token, which is given a time stamp and marked with the name of the session.

When the applet connects to request a session, it must present its token to the server. The server searches its list of active tokens for a match, discarding expired tokens along the way. If the token is found, the server verifies that the session matches, then discards the token.
If the server cannot find a token or if a session mismatch is detected, the client is disconnected and a descriptive error message is written to the system log, including the port and address of the offending client. The server also logs the occurrence of timed-out tokens.

For applets wishing to establish additional sessions, for instance, in response to selecting *New* from the file menu, protocol exists between the client and OC://WebConnect server to allow an existing authenticated session to request a new token.

Limitations:

- This feature will not work with a channel-type web architecture such as Marimba.
- This feature will not be immediately available for OpenVista.
- The Admin Port must be protected.

Dependencies:

In order for this feature to be effective, a secure link must be provided between the WebServer and the client browser.

SSL should be enabled between the Client Browser and OC://WebConnect in order to protect the token when requesting a session.

This feature will not work with any other OC://WebConnect implementation relying on static HTML pages for appletaunching. A live connection must be made from the web server to the OC://WebConnect server to fetch the token. A CGI-BIN is provided with the product for performing this function.

Security Questions

Where can I get more information on SSL and RC4?

Both Netscape and RSA have Web sites where more information is available.

For more information on SSL, please refer to:

http://search.netscape.com/newsref/std/sslref.html
http://www.netscape.com/ - and do a search on “SSL”.

More information on RC4 from RSA Data Security Inc. can be found at:

http://www.rsa.com/

What if I want to get my own certificate?

An alternative is to use a CA product, such as Netscape Certificate Server, Entrust Web CA or XCert Sentry CA. These products will all generate keys and certificates for their secure web servers, and can generate a certificate for the OC://WebConnect certificate request.
It is recommended that customers use the third party CA or CA product to generate the certificate for the secure web server only, and let OC://WebConnect generate its own certificate -- it is much less trouble and no less secure. OC://WebConnect always generates its own keys and certificate request, so there is no increase in security by having a third party certify the request.

Chapter 16: National Language Support

Overview

OC://WebConnect provides National Language Support (NLS) and other localization features for the administrator on the server side and for the end user. Server languages are totally independent from the individual client session languages allowing for true international use of this product. The localization features are divided into four categories; server language support, language for each client session, target host code page for the session, and keyboard support

OC://WebConnect Server Language Localization

OC://WebConnect can be configured to one of four possible server languages. The OC://WebConnect Server Language is used for the HTML configuration pages, the Graphical (GUI) Configuration client, the OC://WebConnect HTML session selection pages, and Online User's Guide. When the server language is changed the HTML files provided with OC://WebConnect will automatically be updated which will include any previous configured server host names or ports.
The server language will be used for all administrator interaction and associated help displays with this OC://WebConnect server.

There are four server languages available for system administration:

- US English
- French
- German
- Spanish

To select the server language as part of the install process, select function number seven, (7) *Configure Default Administration Language,* from the configure menu. Then select the appropriate number for the language you want to use for server administrative functions and press RETURN. Please refer to the *OC://WebConnect Installation Guide* for more detail instructions.

To change the server administration language after initial installation use the **OC://WebConnect Configuration Utility**. Select function number seven, (7) *Configure Default Administration Language,* from the utility menu. Then select the appropriate number for the language you want to use for server administrative functions and press RETURN. (For more information see *Chapter 5: Server Administration and Configuration*) .

OC://WebConnect Client Language Localization

The language for the client is chosen as part of the session setup. The language chosen will be used for all client generated messages, information displays, and menus at the client emulator window. The available languages are shown below along with the code used internally to represent that language. The chosen language is not related to the content or format of the data displayed in the emulator session.

LANGUAGE/COUNTRY	INTERNAL CODE
US English	en_US
UK English	en_GB
French	fr_FR
German	de_DE
Italian	it_IT
Swiss/German	de_CH
Swiss/French	fr_CH
Norwegian	no_NO
Dutch	nl_NL
Castilian Spanish	es_ES
Portuguese/Portugal	pt_PT
Portuguese/Brazil	pt_BR
Swedish	sv_SE
Turkish	tr_TR
Japanese	ja_JP
Chinese Traditional	zh_TW
Chinese Simplified	zh_CN
Korean	ko_KR

The client language is chosen from either the **HTML Administration and Configuration** or **GUI Configurator.** (See *Chapter 5: Server Administration and Configuration*) .

To Change the Client Language using HTML Configuration

Select the Configuration link on the main OC://WebConnect HTML page and enter the Administrator Password. Choose the session you wish to edit or create a new session. Selecting Edit, Copy or New will automatically link you to the selected Configuration page. From this page, link to the Display Page using the left hand buttons. Highlight the desired client language and either press return or use the left mouse click to select the highlighted language. This same page can be used to set other localized parameters as defined in this chapter.

To Change the client language using the GUI Configurator

Selecting the GUI Configurator link on the main OC://WebConnect HTML page and enter the Administrator Password. Select the sessions tab and then choose the session you wish to edit by selecting Properties or create a new session by selecting Create. Select Display Settings and highlight the desired client language and either press return or use the left mouse click to select the highlighted language. This same page can be used to set other localized parameters as defined in this chapter.

OC://WebConnect Target Host Code Page Support

Each target host has identified a code set and an associated code page. This code set and code page may be the system chosen default, but it does exist. For MVS systems in the United States the default code set is 697 and the code page is 37. The code set is the defined set of graphic characters supported. In the code set 697 example there is a graphic for the dollar sign, but no graphic for the pound-sterling-sign. The code page defines an encoding structure for each graphic in the code set. In code page 37 the graphic for the dollar sign is represented by the hexadecimal code of 5B. In code page 285 (UK English) the pound-sterling is represented by the same code point (5B). **OC://WebConnect** only requires knowledge of the code page.

Language support and code page support are not related in **OC://WebConnect** Version 3.1 and above. Language is used to identify the character strings displayed for operation of the client and configuration of the server. The code page is used to transform the target host data into Unicode equivalents. The client can function using the Italian language and the Swedish code page or the French language and the Chinese code page or any other combination.

When a session is defined for **OC://WebConnect** a target system code page number will be defined. For convenience, the most prevalent code page for the client session language is displayed for a default. The administrator may change this value.

The defaults displayed are shown below:

LANGUAGE/COUNTRY	EBCDIC DEFAULT CODE PAGE	VT DEFAULT CODE PAGE
US English	37	819
UK English	285	819
French	287	819
German	273	819
Italian	280	819
Swiss/German	500	819
Swiss/French	500	819
Norwegian	277	819
Dutch	37	819
Spanish	284	819
Portuguese/Portugal	37	819
Portuguese/Brazil	37	819
Swedish	278	819
Turkish	1026	819
Japanese	290,1027,300	33722
Chinese Traditional	937	964
Chinese Simplified	935	1383
Korean	833,834	no support

The client emulator window always uses code page 13488 (Unicode UCS-2 level 1).

The programming mechanism to convert from the host code page to the emulator code page, and back, is called the transform type. In **OC://WebConnect** we have defined six types of transformation. The purpose of having different transformation types is to reduce the size of translate tables, decrease the time to perform translations, or to reduce the amount of data sent between server and client.

The transform types are described below:

CONFIGURATION TITLE	INTERNAL CODE	DESCRIPTION
Single Byte to/from Single Byte	sbonly	Classic single byte translation table. The input code page is transformed to single byte ASCII Latin 1. The ASCII Latin 1 is mapped to the first 256 Unicode positions.
Single Byte to/from Unicode	sbcs	The single byte code page is transformed into its appropriate Unicode character. Supports all of the Latin XX and special characters. Unicode (2 bytes per character) is transmitted between server and client.
Single/Multi-byte to/from Unicode (Multiple Tables)	sbmb	Multiple tables are used to transform single byte EBCDIC characters into Unicode. Another single table is used to transform EBCDIC double byte into Unicode. This is used when Asian code page rotate is supported. Unicode is transmitted between server and client.
Single/Multi-byte to/from Unicode (Single Table)	mixed	A single table is used transform EBCDIC Asian single and double byte into Unicode. This is much more efficient than multiple tables, but cannot rotate. Unicode is transmitted between client and server.
Enhanced UNIX code to/from Unicode	euc	Used to transform from Asian UNIX EUC host representation into Unicode. Unicode is transmitted between client and server.
UNIX PC Code to/from Unicode	pc	Used when the Asian UNIX host is encoded with a PC encoding scheme. Unicode is transmitted between client and server.

The transform tables are very large, so we ship only selected tables to represent the languages and transform type that are supported. The shipped tables are shown below:

LANGUAGE	CODE PAGE	CONVERSION	COMMENTS
US English Netherlands Portugal Brazil	037 037 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII is mapped to Unicode at the client
UK	285 285 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
France	297 297 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Germany	273 273 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Italy	280 280 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Switzerland	500 500 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Norway	277 277 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode

LANGUAGE	CODE PAGE	CONVERSION	COMMENTS
Spain	284 284 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Sweden	278 278 819	Single Byte to/from Single Byte (sbonly) Single Byte to/from Unicode (sbcs) none	EBCDIC/ASCII EBCDIC/Unicode ASCII/Unicode
Japan	290 1027 300 942 33722	Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Multiple Tables) UNIX PC to/from Unicode Enhanced UNIX code to/from Unicode	EBCDIC Katakana single byte / Unicode EBCDIC Latin 1 single byte / Unicode EBCDIC Double byte / Unicode ASCII PC mixed (1041 + 301) / Unicode EUC (895 + 952 + 896 + 953) / Unicode
Korea	833 834	Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Multiple Tables)	EBCDIC single byte / Unicode EBCDIC double byte / Unicode (note this table is provided algorithmically) UNIX is not supported for Korean

LANGUAGE	CODE PAGE	CONVERSION	COMMENTS
Simplified Chinese	836 837 935 1381 1383	Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Single Table) UNIX PC to/from Unicode Enhanced UNIX code to/from Unicode	EBCDIC single byte / Unicode EBCDIC double byte / Unicode EBCDIC mixed (836 + 837) / Unicode ASCII PC mixed (1115 + 1380) / Unicode EUC (367 + 1382) / Unicode
Traditional Chinese	037 835 937 948 964	Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Multiple Tables) Single/Multi-byte to/from Unicode (Single Table) UNIX PC to/from Unicode Enhanced UNIX code to/from Unicode	EBCDIC single byte / Unicode EBCDIC double byte / Unicode EBCDIC mixed (037 + 835) / Unicode ASCII PC mixed (1043 + 927) / Unicode EUC (367 + 960 + 961) / Unicode
Turkey	1026 819	Single Byte to/from Unicode (sbcs) none	EBCDIC/Unicode ASCII/Unicode

Code page rotate is a capability of OC://WebConnect used mostly to support the rotation of code pages between Japanese Latin 1 EBCDIC and Japanese Katakana EBCDIC. The function is mapped to a keystroke using the function RotateCP, the default keystroke is Ctrl-F5. The keymap entry is " RotateCP=<Control+F5>".

The function is effective only for "Single Double Byte to/from Unicode (Multiple tables)" type transforms. It is automatically activated when more than one single byte code page is coded in the code page field. The rotation occurs between the single byte tables, the last entered code page number is the double byte table. When the RotateCP key is pressed the code page is changed to the next page in rotation, then, the entire screen data it re-transformed and transmitted.

The target host code page number and the transform type are chosen from either the **HTML Administration and Configuration** or **GUI Configurator.** (See *Chapter 5: Server Administration and Configuration*) .

To Change the Transform Type using the HTML Administration and Configuration

Select the Configuration link on the main OC://WebConnect HTML page and enter the Administrator Password. Choose the session you wish to edit or create a new session. Selecting Edit, Copy or New will automatically link you to the selected Configuration page. From this page, link to the Display Page using the left hand buttons. Highlight the desired host code page number and either press return or use the left mouse click to select the highlighted code page. Similarly, highlight the transform type required to support the desired language conversion as stated in the above tables.

To Change the Transform Type using the GUI Configurator

Selecting the GUI Configurator link on the main OC://WebConnect HTML page and enter the Administrator Password. Select the sessions tab and then choose the session you wish to edit by selecting Properties or create a new session by selecting Create. Select Display Settings and highlight the desired host code page number and either press return or use the left mouse click to select the highlighted code page number. Similarly, highlight the transform type required to support the desired language conversion as stated in the above tables.

OC://WebConnect Keyboard Considerations

OC://WebConnect provides a key mapping facility. This facility is provided to allow emulator specific keys to be mapped to a platform specific key. For example, the 3270 clear key can be mapped to the PC Ctrl plus A key. The key mapping function also allows the mapping of certain specific international keys to their proper codes. A sample international keymap is provided (natl3270.kbm) which will support keyboards in all supported locales.

If you must add a new or different keyboard layout the following information will aid you.
Many international keyboards have characters in different locations than their US counterparts. The different locations are mapped for **OC://WebConnect** by the Java virtual machine and are transparent to **OC://WebConnect**. In addition some keys, on international keyboards, are typed through the use of an alt-Gr key. Normally the alt-Gr key has replaced the right hand Ctrl key of the US keyboard. Java presents keys typed by the alt-Gr key to **OC://WebConnect** as the desired key with a Ctrl modifier.

If the requested key is represented by a Unicode value greater than 128 the Ctrl modifier is ignored by **OC://WebConnect**. This handles the great majority if international characters. There are a set of characters that are represented by a Unicode value less than 128 and are keyed on certain international keyboards using the alt-Gr key. Then an entry must be added to the keymap to represent

that key. If no keymap entry is made **OC://WebConnect** treat the key as a ctrl modified key with no map and ignore the key. The ctrl modified key is mapped to itself.

Example:

> The at-sign (@) is represented in Unicode as decimal 64 (X0040). On the French keyboard the at-sign is typed by holding the alt-Gr key and pressing the zero (0) key. In the **OC://WebConnect** Java code, if the at-sign plus Ctrl is not mapped, it will be ignored. The following line in the keymap file corrects the situation:
>
> 0040=<Control+at>

This line tells the Java client to replace the at "@" character with hexadecimal 40 (the at character) when the control key is also present.

OC://WebConnect includes sample international mappings for 3270 (natl3270.kbm), 5250 (natl5250.kbm) and VT (natlvt.kbm). The samples should be sufficient for most locales.
To assign a particular keyboard map to a specific client session number use either the **HTML Administration and Configuration** or **GUI Configurator.** (See *Chapter 5: Server Administration and Configuration*) .

To Change the Keyboard map using HTML Configuration

Select the Configuration link on the main OC://WebConnect HTML page and enter the Administrator Password. Choose the session you wish to edit or create a new session. Selecting Edit, Copy or New will automatically link you to the selected Configuration page. From this page, link to the Display Page using the left hand buttons. Highlight the desired keyboard map and either press return or use the left mouse click to select. This same page can be used to set other localized parameters as defined in this chapter.

To Change the Keyboard map using the GUI Configurator

Selecting the GUI Configurator link on the main OC://WebConnect HTML page and enter the Administrator Password. Select the sessions tab and then choose the session you wish to edit by selecting Properties or create a new session by selecting Create. Select Display Settings and enter the desired keyboard definition file name. This same page can be used to set other localized parameters as defined in this chapter.

OC://WebConnect File Transfer Localization

In the Asian locales file transfer to TSO, CICS and VM is accomplished using the host function APVUfile. APVUfile supports the capabilities of IND$file in addition to new capabilities for the Asian market. APVUfile allows optional inclusion of SI/SO in the transferred file. APVUfile supports optional trailing blank elimination. APVUfile most significantly supports transfer of double byte EBCDIC and translation into locale dependent data.

When files are transferred in ASCII mode using IND$file the conversion from/to ASCII is accomplished by IND$file host component. All translation is from/to Latin 1, therefore, the previously described additional translation is required. When files are transferred in ASCII (or JISCII) using APVUfile the conversion is accomplished by the off-host component (**OC://WebConnect** in our case). **OC://WebConnect** performs the conversion in two steps. First, the **OC://WebConnect** server transforms the data from double byte EBCDIC to Unicode using the parameters supplied for the session (code page number and transform type). Second, the Java client uses the Java virtual machine to transform from Unicode into the platform specific encoding of files. The double translation can result in data inconsistencies when a character does not exist in Unicode, but is unavoidable in a Java environment.

The host name of the file transfer program is customizable. The default is ind$file. **OC://WebConnect** automatically maps ind$file to apvufile for the supported Asian locales. If the customized name is not ind$file, the customized name will be used.

Chapter 17: OC://WebConnect Print Solutions

Selecting a Print Solution

As an administrator, you can allow users to print a session screen and enable 3287 printing by selecting the **Print Settings** option on the **Sessions** tab in the **GUI Configurator** window. Choose between four options that works best for your environment, allowing users to print from an open session window and enables 3287 printing from a browser window. Printing options are listed below.

OC://WebPrint Option

This option allows full control over font size and style and "auto fits" the document based on page orientation. This option requires you to install OC://WebPrint locally on the client, and a program for installation is provided. OC://WebPrint is dependent upon the runtime environment, so if you switch from an Internet Explorer to a Netscape browser, you must install the appropriate OC://WebPrint libraries.

Operating Systems Requirements	Java Environment Requirements	Installation Requirements
• Windows '95 • Windows NT	• Internet Explorer 3.0 or 4.0 • Netscape 3.0 or 4.0	Local installation required.

To Install OC://WebPrint

1. Download *OCWebPrint.exe* (428K) by accessing your OC://WebConnect server. To access your server, type in your browser's address field the path for the server with **OCWebPrint.html** at the end. An HTML page with the *OCWebPrint.exe* displays.
2. Close your browser.
3. Run *OCWebPrint.exe*. During the setup, you will be asked to provide the target directory (from your workstation's CLASSPATH) to install OC://WebPrint classes target directory (from your workstation's PATH) to install OC://WebPrint binary files.
4. Select one or more Java Virtual Machine (VM) you want OC://WebPrint to support. Sun SDK and your current browser will be selected by default.
5. Click the Install button on the window that displays.
6. Follow the online installation instructions provided.
7. Restart your browser.

JavaScript Print Option

Use the JavaScript Print option if you do not have access to the software required for the OC://WebPrint option. There is no software to install on the client. The JavaScript Print option provides quality printing because printing is performed by the browser.

Operating Systems Requirements	Java Environment Requirements	Installation Requirements
• Windows '95 • Windows NT • Solaris	• Netscape 3.x or 4.x • Internet Explorer 4.01	No local installation required.

JDK 1.1 Print option

The JDK 1.1 print solution using the Java solution built into all browsers which support JDK 1.1. No additional installation is required.

Operating Systems Requirements	Java Environment Requirements	Installation Requirements
• Windows '95 • Windows NT	• Netscape 4.x with JDK 1.1 patch • Internet Explorer 4.x • HotJava 1.x	No local installation required.

Printing a Screen

You can print a single session window after your administrator selects a print screen option.

1. From your open session window, select **Print Screen** from the **File** menu. The standard printer window for your system opens.
2. Follow the procedures for printing with your system.

Using 3287 Printing

You can print from your browser using the 3287 print session option after your administrator selects a print option. To print from your browser to a specific logical unit (LU) and gateway:

1. Select **3287 Print Session** from the **Select Sessions** window. A printer session window opens.
2. Make sure the correct gateway and LU displays in the printer session window. If not, reconfigure the **Host** and **Port Number** fields in the **Session Settings** on the **Session Properties** tab.
3. Send a mainframe print job to the selected LU. The print session window displays that the session is printing.

Note:

- To check the LU and gateway to which you are printing, select **Help Desk** from the **Help** menu on the printer session window that displays when you start a 3287 print session. A status window opens identifying printer information, such as the number jobs to print.
- To run print jobs unattended, disable the **Show Printer Dialog** from the **Settings** menu.

Appendix A: Glossary

3270 emulation

Imitation of an IBM 3270 computer terminal on a terminal connected to a TCP/IP computer so that the imitating system accepts the same data, executes the same computer programs, and achieves the same results as the imitated IBM terminal.

3270 session

The name given to a session when the TCP/IP computer is communicating with the host computer through the SNA3270 Presentation Services or 3270 TELNET Server.

3770 emulation

Imitation of an IBM RJE workstation on a terminal connected to a TCP/IP computer so that the imitating system accepts the same data, executes the same computer programs, and achieves the same results as the imitated IBM RJE workstation.

3770 session

The name given to a session when the TCP/IP computer is communicating with the host computer through the SNA3770 Presentation Services.

5250 emulation

Imitation of an IBM 5250 computer terminal on a terminal connected to a TCP/IP computer so that the imitating system accepts the same data, executes the same programs, and achieves the same results as the imitated IBM terminal.

API (Application Program Interface)

A language and message format used by an application program to communicate with the operating system or other system program such as a database management system (DBMS). APIs are implemented by writing function calls in the program, which provide the linkage to a specific subroutine for execution. Thus, an API implies that some program module or routine is either already in place or that must be linked in to perform the tasks requested by the function call.

Applet :

A small Java program that can be embedded in an HTML page. Applets differ from full-fledged Java applications in that they are not allowed to access certain resources on the local computer, such as files and serial devices (modems, printers, etc.), and are prohibited from communicating with most other computers across a network. The current rule is that an applet can only make an Internet connection to the computer from which the applet was sent.

ASCII

American Standard Code for Information Interchange. A standard coded character set, consisting of 7-bit coded characters (8 bits including a parity check bit), used for information exchange among most non-IBM data processing systems, data communication systems, and associated equipment. The basic-ASCII character set contains English language characters. See EBCDIC and extended ASCII.

Attribute byte

The byte used to establish the characteristics of the field that follows it. For example, a byte that indicates that the following field is blinking, highlighted, or unprotected.

Browser

The program that serves as your front end to the World Wide Web on the Internet. In order to view a site, you type its address (URL) into the browser's Location field; for example, www.computerlanguage.com, and the home page of that site is downloaded to you. The home page is an index to other pages on that site that you can jump to by clicking a "click here" message or an icon. Links on that site may take you to other related sites.

Byte

A sequence of eight adjacent binary digits that are operated upon as a unit and that constitute the smallest addressable unit in the system.

Certificate Authority

An organization that issues digital certificates (digital IDs) and makes its public key widely available to its intended audience.

CGI (Common Gateway Interface)

A set of rules that describe how a Web Server communicates with another piece of software on the same machine, and how the other piece of software (the “CGI program”) talks to the web server. Any piece of software can be a CGI program if it handles input and output according to the CGI standard. Usually a CGI program is a small program that takes data from a web server and does something with it, like putting the content of a form into an e-mail message, or turning the data into a database query. You can often see that a CGI program is being used by seeing “cgi-bin” in a URL, but not always.

CGI-bin

The most common name of a directory on a web server in which CGI programs are stored.

The “bin” part of “cgi-bin” is a shorthand version of “binary”, because once upon a time, most programs were refered to as “binaries”. In real life, most programs found in cgi-bin directories are text files -- scripts that are executed by binaries located elsewhere on the same machine.

Client

In the TCP/IP network environment, a process that employs (or consumes) resources provided by a server. Client is initiated by the user when issuing a networking command. The client process sends a request for service to a server process on the remote host. If the request is honored, a connection is established between the local client and the remote server processes. See server.

Code page

A table that defines a coded character set by assignment of a character meaning to each code point in the table for a language or a country.

Configurator

The OC://WebConnect automated, menu-driven utility used for customizing configuration files for the OC://WebConnect Server.

Configuration

(1) The arrangement of a computer system or network as defined by the nature, number, and the chief characteristics of its functional units. (2) The devices and programs that make up a system, subsystem, or network.

Daemon

A program running all the time on a UNIX system.

Digital Certificate

The digital equivalent to an ID card in the RSA public key encryption system. Also called digital IDs, digital certificates are issued by certification organizations after verifying that a public key belongs to a certain owner. The certification process varies depending on the certification authority (CA) that issues the certificates and the level of certification.

Domain Name

The unique name that identifies an Internet site. Domain Names always have 2 or more parts, separated by dots. The part on the left is the most specific, and the part on the right is the most general. A given machine may have more than one Domain Name but a given Domain Name points to only one machine.

E-mail (Electronic Mail)

Messages, usually text, sent from one person to another via computer. E-mail can also be sent automatically to a large number of addresses (Mailing List).

Emulation

The imitation of all or part of one system by another so that the imitating system accepts the same data, executes the same programs, and achieves the same results as the imitated computer system.

Extranet

Business to business communications. A network that allows an organization's partners and suppliers to interact with corporate information and applications. This communication is typically done via a public or private switched network or virtual private network, VPN.

FAQ (Frequently Asked Questions)

FAQs are documents that list and answer the most common questions on a particular subject. FAQs are usually written by people who have tired of answering the same question over and over.

Fire Wall

A combination of hardware and software that separates a LAN into two or more parts for security purposes.

FTP (File Transfer Protocol)

A very common method of moving files between two Internet sites. FTP is a special way to login to another Internet site for the purposes of retrieving and/or sending files. There are many Internet sites that have established publicly accessible repositories of material that can be obtained using FTP, by logging in using the account name anonymous, thus these sites are called anonymous ftp servers.

Gateway

(1) A functional unit that connects two computer networks or different network architectures. (2) A special purpose, dedicated computer that attaches to two or more networks and routes packets from one to the other.

Host

Any computer on a network that is a repository for services available to other computers on the network.

Host application subsystem

The host application subsystem is the program running on the host mainframe to and from which data is sent and received using the emulated station. Any VTAM application which supports 3270 display stations, 3770 RJEs, and printers (i.e., LU types 1, 2, and 3) can be accessed through the OC://WebConnect Server. For 3270 sessions, these host application programs include Customer Information Control System/Virtual Storage (CICS/VS), Information Management System (IMS), Time Sharing Option (TSO), and Virtual Machine/Conversational Monitor System (VM/CMS). For 3770 sessions, these host application programs include Job Entry Subsystem (JES) 2 and 3.

IBM channel. In the IBM System/370 and 370/XA architecture, the processor which does all of the actual input/output (I/O) processing.

HTML (HyperText Markup Language)

The coding language used to create Hypertext documents for use on the World Wide Web. HTML looks a lot like old-fashioned typesetting code, where you surround a block of text with codes that indicate how it should appear, additionally, in HTML you can specify that a block of text, or a word, is linked to another file on the Internet. HTML files are meant to be viewed using a World Wide Web Client Program, such as Netscape or Mosaic.

HTTP (HyperText Transport Protocol)

The protocol for moving hypertext files across the Internet. Requires a HTTP client program on one end, and an HTTP server program on the other end. HTTP is the most important protocol used in the World Wide Web (WWW).

Hypertext

Generally, any text that contains links to other documents - words or phrases in the document that can be chosen by a reader and which cause another document to be retrieved and displayed.

Internet (Upper case I)

The collection of independent and autonomous networks linked by gateways that use primarily the TCP/IP protocol suite and function as a single, cooperative virtual network.

Internet (Lower case i)

Any time you connect 2 or more networks together, you have an internet - as in inter-national or inter-state.

Internet address

The 32-bit address assigned to hosts on a TCP/IP internet.

Intranet

A private network inside a company or organization that uses the same kinds of software that you would find on the public Internet, but that is only for internal use. As the Internet has become more popular many of the tools used on the Internet are being used in private networks, for example, many companies have web servers that are available only to employees. Note that an Intranet may not actually be an internet -- it may simply be a network.

IP Address (Internet Protocol Address)

The physical address of a computer attached to a TCP/IP network. Every client and server station must have a unique IP address. Client workstations have either a permanent address or one that is dynamically assigned for each dial-up session (see DNS). IP addresses are written as four sets of numbers separated by periods; for example,

204.171.64.2.

IP. Internet Protocol

The TCP/IP standard protocol that defines the basic unit of information passed across an internet.

IP Routing. Internet

Protocol Routing provides a virtual connection from one TCP/IP-based LAN to another TCP/IP-based LAN through an SNA environment

Java

Java is a network-oriented programming language invented by Sun Microsystems that is specifically designed for writing programs that can be safely downloaded to your computer through the Internet and immediately run without fear of viruses or other harm to your computer or files.

JDK (Java Development Kit)

A software development package from Sun Microsystems that implements the basic set of tools needed to write, test and debug Java applications and applets.

JVM (Java Virtual Machine)

A Java interpreter from the JavaSoft division of Sun. It converts the Java intermediate language (byte code) into machine language one line at a time and then executes it. The Java Virtual Machine is licensed to software companies that incorporate it into their browsers and server software. Since it is used on all major platforms, Java programs run in "virtually" every computer. Microsoft also calls its Java interpreter a Java Virtual Machine.

Keyboard Mapping

The process whereby the Terminal Emulator maps the IBM 3270/3770 keys to the keyboard of the particular display station attached to the TCP/IP computer.

LU

Logical unit**.** In SNA, a port through which an end user accesses the SNA network in order to communicate with another end user and through which the end user accesses the functions provided by System Services Control Points (SSCPs).

LU 6.2

Provides a generalized facility for program-to-program communications. See APPC and LU type.

LU type

Shortened form for LU-LU session type. In SNA, the classification of an LU-LU session in terms of the specific subset of SNA protocols and options supported by the logical units (LUs) for that session. The SNA3270 Terminal Emulator supports LUs for display stations (LU type 2) and for printers (LU types 1 or 3). The SNA3770 Terminal Emulator supports LU type 1.

Plug-in

A piece of software that adds features to a larger piece of software. Common examples are plug-ins for the Netscape® browser and web server. The idea behind plug-in's is that a small piece of software is loaded into memory by the larger program, adding a new feature, and that users need only install the few plug-ins that they need, out of a much larger pool of possibilities. Plug-ins are usually created by people other than the publishers of the software the plug-in works with.

Port

A place where information goes into or out of a computer, or both. On the Internet port often refers to a number that is part of a URL, appearing after a colon (:) right after the domain name. Every service on an Internet server listens on a particular port number on that server. Most services have standard port numbers, e.g. Web servers normally listen on port 80.

Protocol

A set of procedures or conventions that are used to formalize data transfer between points.

PU

Physical unit. In SNA, the component that manages and monitors the resources (such as attached links and adjacent link stations) of a node, as requested by an SSCP via an SSCP-SSCP session.

Security Certificate

A chunk of information (often stored as a text file) that is used by the SSL protocol to establish a secure connection. Security Certificates contain information about who it belongs to, who it was issued by, a unique serial number or other unique identification, valid dates, and an encrypted "fingerprint" that can be used to verify the contents of the certificate. In order for an SSL connection to be created both sides must have a valid Security Certificate.

Server

In a TCP/IP network environment, a process that provides resources to a network. The server is the remote host process that services the request made by the client. The server is a background process that listens for incoming service requests. When a server receives a request, it establishes a connection with the requesting client, spawns a subprocess, and returns to listening for more incoming requests.

Session

A logical connection between two stations that allows them to communicate.

SNMP (Simple Network Management Protocol)

A set of standards for communication with devices connected to a TCP/IP network. Examples of these devices include routers, hubs, and switches. A device is said to be "SNMP compatible" if it can be monitored and/or controlled using SNMP messages. SNMP messages are known as "PDU's" - Protocol Data Units. Devices that are SNMP compatible contain SNMP "agent" software to receive, send, and act upon SNMP messages.

SNA (Systems Network Architecture)

IBM's description of the logical structure, formats, protocols, and operational sequences for transmitting information units through and controlling the configuration and operation of networks.

SSL (Secure Sockets Layer)

A protocol designed by Netscape Communications to enable encrypted, authenticated communications across the Internet. SSL used mostly (but not exclusively) in communications between web browsers and web servers. URL's that begin with "https" indicate that an SSL connection will be used. SSL provides 3 important things: Privacy, Authentication, and Message Integrity.

TCP/IP (Transmission Control Protocol/Internet Protocol)

(1) TCP provides a connection-oriented byte-stream service that is reliable and flow controlled. IP provides a connectionless datagram service that transparently forwards messages through the gateway. TCP is built on top of IP. TCP/IP protocols are defined by the Department of Defense Advanced Research Projects Agency (DARPA). (2) TCP/IP is also used synonymously for TCP/IP Application Suite. See TCP/IP Application Suite.

TCP/IP Application Suite

A collective term used for referring to DARPA-standard applications commonly distributed with the TCP/IP protocol. Two such applications are File Transfer Protocol (FTP) and Terminal Emulator Protocol (TELNET).

Telnet

(1) Acronym for teletype network. (2) A TCP/IP protocol used for remote login between hosts.

Terminal

A display station, RJE workstation, or printer.

Terminal emulator

In the OpenConnect Server, refers to either the SNA3270 Terminal Emulator or the SNA3770 Terminal Emulator. The OpenConnect Server's SNA3270 Terminal Emulator provides IBM 3270 Information Display System emulation of IBM 3278 Display Stations, IBM 3278 Color Display Stations, and IBM 3287 Printers. The SNA3770 Terminal Emulator provides IBM 3770 Data Communication System emulation of the IBM 3776 Communication Terminals and IBM 3777 Communication Terminals. The 5250 TELNET Server terminal emulation emulates IBM 5250 midrange terminal types.

URL (Uniform Resource Locator)

The standard way to give the address of any resource on the Internet that is part of the World Wide Web (WWW).