Administrator's Guide

Citrix ICA Java Client

Version 4.11

Citrix Systems, Inc.

Document Code cad.java1.8

Contents

Before you Begin

This manual is for system administrators responsible for installing, configuring, and maintaining the Citrix ICA Client for Java (also called the Citrix ICA Java Client). This manual assumes knowledge of:

- Citrix MetaFrame Application Server for Windows or Citrix *WINFRAME*
- The operating system on the client computer
- Java Virtual Machines (JVMs) or Java-enabled Web browsers
- Installation, operation, and maintenance of network and printing hardware

How to Use this Guide

To get the most out of the *Citrix ICA Java Client Administrator's Guide*, review the table of contents to familiarize yourself with the topics discussed.

This guide contains the following sections:

Chapter	Contents
Chapter 1, "Introduction to the Citrix ICA Java Client"	Gives a list of features and system requirements.
Chapter 2, "Installing the Citrix ICA Java Client"	Describes how to install the Citrix ICA Java Client.
Chapter 3, "Configuring the Citrix ICA Java Client"	Describes how to start and configure connections to Citrix servers.

Conventions

The following conventional terms, text formats, and symbols are used throughout the printed documentation:

Convention	Meaning
Bold	Indicates boxes and buttons, column headings, command-line commands and options, icons, dialog box titles, lists, menu names, tabs, and menu commands.
Italic	Indicates a placeholder for information or parameters that you must provide. For example, if the procedure asks you to type *filename*, you must type the actual name of a file. Italic also indicates new terms and the titles of other books.
ALL UPPERCASE	Represents keyboard keys (for example, CTRL, ENTER, F2).
[brackets]	Encloses optional items in syntax statements. For example, [*password*] indicates that you can choose to type a *password* with the command. Type only the information within the brackets, not the brackets themselves.
...(ellipsis)	Indicates a command element can be repeated.
`Monospace`	Represents examples of screen text or entries that you type at the command line or initialization files.
▶	Indicates a procedure with sequential steps.
•	Indicates a procedure with only one step.
▪	Indicates a list of related information, not procedural steps.

The Citrix ICA Clients allow users to connect to MetaFrame servers and *WINFRAME* servers. When describing a feature or procedure common to MetaFrame and *WINFRAME* servers, this manual uses the term *Citrix server*. When describing a feature unique to a MetaFrame or *WINFRAME* server, this manual specifies either a MetaFrame or *WINFRAME* server.

Finding More Information

This manual contains conceptual information and installation and configuration steps for the Citrix ICA Java Client. For additional information, consult the following sources:

- The *Citrix ICA Client Administrator's Guides* for the ICA DOS, Win16, Win32, Web, Macintosh, and UNIX Clients.
- For instructions on installing, configuring, and maintaining your Citrix servers, see the documentation included in your MetaFrame or *WINFRAME* package.

This manual and other Citrix documentation is available in Adobe Portable Document Format (PDF) in the documentation directory of your MetaFrame or *WINFRAME* CD-ROM. Using the Adobe Acrobat Reader, you can view and search the documentation electronically or print it for easy reference. To download the free Adobe Acrobat Reader, please visit Adobe's Web site at http://www.adobe.com.

Important Consult the Readme.txt files for MetaFrame, *WINFRAME*, and the Citrix ICA Clients for any last-minute updates, installation instructions, and corrections to the documentation.

Citrix on the World Wide Web

Citrix offers online Technical Support Services at http://www.citrix.com that include the following:

- Downloadable Citrix ICA Clients, available at http://download.citrix.com
- A Frequently Asked Questions page with answers to the most common technical issues
- An FTP server containing the latest service packs and hotfixes for download
- An Online Knowledge Base containing an extensive collection of technical articles, troubleshooting tips, and white papers
- Interactive online support forums

Year 2000 Readiness

For a detailed description of the Year 2000 Readiness of Citrix products, see our Web site at http://www.citrix.com/misc/y2000.htm.

CHAPTER 1

Introduction to the Citrix ICA Java Client

The Citrix ICA Java Client lets you access a Citrix *WINFRAME* or MetaFrame server from a client running a Java virtual machine (JVM). The JVM can be a PC application running a JVM, a Java-based device, or a Web browser that supports Java applets.

Application and Applet Modes

The Citrix ICA Java Client can be used in one of two modes: applet mode or application mode. Each mode can be used with or without Program Neighborhood.

In *applet mode*, the ICA Java Client resides on a Web server. The Web server contains an HTML page with the ICA Java Client applet tag. Users run the ICA Java Client by using a browser that has Java support.

When run in applet mode, the ICA Java Client is also called the Java Web Client.

In *application mode*, the ICA Java Client resides on a client system that has an installed Java virtual machine. Connections to a Citrix server are initiated by executing the ICA Java Client from the command line (with the exception of Macintosh client systems, which have no command line).

In general, the ICA Java Client mode used depends upon the type of access users need. From the user's point of view, application mode is the more flexible method of launching remote control sessions. Application mode sessions run in their own window on the local desktop. Users can specify command line parameters when invoking sessions to control such ICA Java Client features as event logging, client printing, hotkeys, window resolutions, initial programs, and server addresses.

In applet mode, users launch sessions that are embedded in an HTML page on a Web browser. Not all of the command line parameters available to application mode users are available to the applet mode user; usually, only those users with write access to the Web server can edit the HTML page that contains the ICA Java Client applet tag and its parameters. For this reason, applet mode sessions offer

more control to the network administrator than the user and may be preferable when administrators want to relieve users of the mechanics of launching sessions. For details on which parameters are available for each mode, see "ICA Java Client Parameters" in Chapter 3.

Although both applet and application modes use the same set of class files and support the same functionality, this manual discusses them separately when procedures are specific to the mode used.

Citrix ICA Java Client Features

This release of the ICA Java Client supports the following features:

- Video resolutions up to 1280 x 1024
- Client printer mapping
- Event logging
- Hotkeys
- Client audio
- COM port mapping
- Business recovery
- Data compression
- Auto-reconnect
- Clipboard data exchange between local and remote desktops (raw textual data only, such as Unicode and ASCII text)
- Shadowing
- International keyboard support
- Multiple locale support
- Euro currency symbol support
- Program Neighborhood native GUI

This release of the ICA Java Client does not support the following features commonly supported by other Citrix ICA Clients:

- Persistent caching
- Encryption levels higher than Basic
- Color depths other than 256 colors
- Transport protocols other than TCP/IP
- Client drive mapping
- Seamless windows

Citrix ICA Java Client Requirements

To run the ICA Java Client as an applet, the client system must have the following:

- A Web browser with Java Development Kit (JDK) 1.1 Java support
- TCP/IP (Dial-up or LAN)
- VGA or equivalent display
- Mouse
- Keyboard
- Access to a Web server that will store the ICA Java Client files

To run the ICA Java Client as an application, the client system must have the following:

- A Java virtual machine (JVM) with JDK version 1.1 or greater support
- TCP/IP (Dial-up or LAN)
- VGA or equivalent display
- Mouse
- Keyboard

Java 1.1 Compliance

The Citrix ICA Java Client Version 4.11 is Java Development Kit (JDK) 1.1-compliant. To run the ICA Java Client you must have a Java Virtual Machine (JVM) or Java-enabled Web browser that fully supports JDK 1.1.

Java Environments

A large number of Java enabled environments are available, and their functionality varies from platform to platform. To validate proper functionality of the Citrix ICA Java Client, Citrix selects a representative group of platforms for testing. The combinations of browsers and platforms (for applet mode), and platforms and JDKs (for application mode) listed in the sections below have been tested and approved for use with the ICA Java Client. The absence of a platform in either list does not necessarily indicate that the ICA client for Java will not work; however, Citrix does not support any platform not included below.

Note To support client systems using JDK 1.0-compliant JVMs or browsers (such as Internet Explorer 3.x or below or Netscape Navigator 3.x or below), you must use the JDK-1.0-compliant version of the Citrix ICA Java Client. The JDK-1.0-compliant Citrix ICA Java Client is available from the Citrix Web site at http://www.citrix.com

Browser Compatibility

The Citrix ICA Java Client has been tested with various browsers and client systems. The following list describes some browser and platform combinations that have been tested and approved for use with the ICA Java Client as an applet:

Browser	Platforms
Microsoft Internet Explorer 4.0, Microsoft Internet Explorer 5.0	Windows 95, Windows 98,Windows NT 4.0, and Solaris 2.6
Microsoft Internet Explorer 4.0x with Apple MRJ 2.1.2	Macintosh
Netscape Navigator 4.04 for OS/2	OS/2 Warp 4.0
Netscape Communicator 4.6	Windows 95, Windows 98, Windows NT 4.0, and Solaris 2.6
HotJava Browser 1.1, HotJava Browser 3.0	Windows 95, Windows 98, Windows NT 4.0, Solaris 2.6, and JavaOS for Business
HotJava Views for JavaOS	Sun JavaOS

Tested JVMs

The Citrix ICA Java Client has been tested with several JDKs and client systems. The following list describes some JDK and platform combinations that have been tested and approved for use with the ICA Java Client as an application:

- Solaris 2.6 - Sun JDK 1.1.7B
- Mac OS 8 - Apple MRJ 2.1.2
- Windows 98 and Windows NT 4.0 - Sun JDK 1.1.7B; Microsoft JVM release 5.0.0.3165
- OS/2 Warp 4.0 - IBM JDK 1.1.7
- JavaOS for Business 2.1 - IBM JDK 1.1.6
- Sun JavaStation Server - Sun JDK 1.1.4

CHAPTER 2

Installing the Citrix ICA Java Client

This chapter explains how to install the Citrix ICA Java Client as an applet and as an application on various platforms.

This chapter also explains how to uninstall the ICA Java Client.

Tasks to Complete

In this chapter you will complete the following tasks.

Applet mode installations:

- Install the ICA Java Client on your Web server—(5 minutes)
- Create a Web page users can visit to launch the ICA Java Client—(10 minutes)

Application mode installations:

- Install the ICA Java Client on each client machine—(5 minutes per machine)
- Macintosh users must create a Macintosh launching script—(5 minutes)
- JavaStation Server users must configure the JavaStation server to serve the ICA Java Client to its client systems—(20 minutes)
- JavaOS for Business Server users must configure the JavaOS for Business server to serve the ICA Java Client to its client systems—(20 minutes)

Installation

Due to the ICA Java Client's support for multiple platforms and its ability to function in both application and applet modes, the installation procedures for the ICA Java Client must account for a wide range of operating systems, hardware, and intended usage. With the exception of a common first step, applet and application mode installations differ and are described separately. The following procedure describes the first step of both applet and application installations.

Copying Setup.class to Your Web Server

The \Icaclnt\Icajava directory on your CD-ROM contains the following installation file:

- **Setup.class**: This setup program is a Java application that you can run to install the ICA Java Client 4.11 on any client system or Web server that has an installed Java virtual machine (JVM).

In addition to Setup.class, the \Icaclnt\Icajava directory contains Install.htm. This HTML document contains a hyperlink to Setup.class in the \Icaclnt\Icajava directory. If you are installing the ICA Java Client as an application, multiple client users can visit this page to download Setup.class for individual application mode installations.

▶ **To copy Setup.class to your Web server**

- Both applet and application mode installations require Setup.class be placed in a location accessible to all client systems. Copy the contents of the \Icaclnt\Icajava directory to a suitable location on your Web server. For your convenience, you can create a separate directory in your Web server's Web root directory for the files. For example, create a directory called \JICA in the Web root directory and copy the files into \JICA. If you are installing the ICA Java Client as an application, make sure you copy both Install.htm and Setup.class to the same directory on the Web server. If you are installing the ICA Java Client as an applet only, you do not need to copy Install.htm to the Web server.

Note The application mode installation procedure described in this chapter facilitates installing the ICA Java Client on multiple client systems. When installing the ICA Java Client on a limited number of client systems, it may be more convenient to simply copy the installation archive by some method of file transfer to your client systems and then proceed with installation. See "Running Setup.class" later in this chapter for more information.

Applet Installation

Unlike ActiveX controls and Netscape plug-ins, which are downloaded once and then saved for future use by client systems, Java applets are not stored by client systems and must be downloaded each time they are used. To execute the ICA Java Client in applet mode, the client system must be able to download the ICA Java Client class files from the Web server.

Note See "Copying Setup.class to Your Web Server" above before performing the following procedure.

Use the following procedure to make the class files accessible to client systems:

▶ **To install the ICA Java Client as an applet**

1. Create a directory in a suitable place on the Web server. For example, create a directory named \APPLET.
2. Use the JVM installed on your Web server to run Setup.class. See "Running Setup.class" for more information about executing this Java setup program on various platforms.
3. Create an HTML page that users can access to initiate sessions on the Citrix server. Place this HTML page in the directory you created in Step 1. See "Creating a Web Page for Applet Mode Installations" later in this chapter for more information.

Application Installation

Note See "Copying Setup.class to Your Web Server" above before performing the following procedure.

▶ **To download and install the ICA Java Client**

1. Make sure the client system is properly configured and connected to the network.
2. Start a Web browser. In the browser's address field, enter the IP address or DNS name of the Web server containing the ICA Java Client files, followed by the name of the ICA Java Client setup HTML file (Install.htm). For example, if you copied the ICA Java Client installation files into a directory named /JICA on a Web server with IP address 126.4.2.7, enter **http://126.4.2.7/JICA/install.htm**. The ICA Java Client setup HTML page appears.
3. Download Setup.class save it to the client system's hard drive. You can create a directory named "Citrix" on the client computer and save the file to this directory. Then use the Java Virtual Machine installed on the client system to run Setup.class. See "Running Setup.class" below for more information on executing this Java setup program on various platforms.

Note Some browsers may try to execute the class file instead of displaying a **Save as** dialog box. In this case, click the right mouse button on the link to Setup.class and select the **Save as** option.

4. After you run Setup.class, installation is complete for Windows, UNIX, and OS/2 client systems. For instructions about how to start the ICA Java Client in application mode, see "Using the ICA Java Client as an Application" in Chapter 3. Macintosh users may see an error message warning that some batch

files were not created during installation. The batch file that cannot be created is a launching script used to automate starting the ICA Java Client in application mode. Macintosh users can create this batch file manually using the procedure in "Creating a Macintosh Launching Script" later in this chapter.

Application mode installations on JavaStation Servers and JavaOS for Business Servers require additional configuration, covered in "Configuring the JavaOS for Business Server 2.1" and "Configuring the JavaStation Server" later in this chapter.

Running Setup.class

The following procedures describe how to run Setup.class on different platforms:

To run Setup.class on a Windows PC

1. Start a command prompt. On Windows 95 or Windows 98, click **Start** and then click **Run**. In the **Run** dialog box, type **command** and press ENTER. On Windows NT 4.0, click **Start** and then click **Run**. In the **Run** dialog box, type **cmd** and press ENTER.
2. If you downloaded the ICA Java Client, change to the directory where you saved the Setup.class file. For example, if you saved Setup.class in \Downloads, type **cd \downloads**.

   Note To install the ICA Java Client from CD-ROM, insert the CD-ROM in the CD-ROM drive. At the command prompt, type cd *x*:, where *x* is the letter of your CD-ROM drive. Press ENTER. Type **cd icaclnt\icajava** and press ENTER.

3. Type ***x**jvmpgm* setup**, where *x* is the full path of your JVM program and *jvmpgm* is the JVM program's name. Press ENTER. For example, if you use the Microsoft JVM program Jview and it is located in a directory called C:\Windows\, type **c:\windows\jview setup**. The ICA Java Client installation wizard appears.

   Notes To find your Java virtual machine, click **Start** on the Windows Taskbar. Select **Find** and then click **Files or Folders**. The **Find** dialog box appears. In the **Named** field, type **jview.exe**. Click **Find Now**. Note the location of jview.exe

   To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, type **c:\windows\jview /d:user.language=fr setup**.

4. The ICA Java Client installation wizard appears. Follow the instructions on your screen. Application mode installations are complete once you have finished using the installation wizard. Applet mode installations require addional configuration, covered in "Creating a Web Page for Applet Mode Installations" later in this chapter.

Note If you install the ICA Java Client on a system running Microsoft Windows NT and do not have administrator privileges, the install process will complete but an error message will appear. You can ignore this error message; the ICA Java Client has been correctly installed. However, uninstall information has not been added to Add/Remove Programs in Control Panel. To uninstall the ICA Java Client, see "Uninstalling the ICA Java Client" later in this chapter.

▶ To run Setup.class on a UNIX system

1. Log in as root.
2. Start a shell.
3. If you downloaded the ICA Java Client, change to the directory where you saved the Setup.class file. For example, if you saved Setup.class in /downloads, type **cd /downloads**.

   Note To install the ICA Java Client from CD-ROM, insert the CD-ROM in the CD-ROM drive and mount the CD. Change to the /ICACLNT/ICAJAVA directory on the CD-ROM.

   ---

4. Type ***x/jvmpgm* setup**, where *x* is the full path of your JVM program and *jvmpgm* is the JVM program's name. Press ENTER. For example, if you use Sun's JVM program Java and it is located in a directory called /usr/bin, type **/usr/bin/java setup**.

   Note To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, type **/usr/bin/java -duser.language=fr setup**.

   ---

5. The ICA Java Client installation wizard appears. Follow the instructions on your screen. Application mode installations are complete once you have finished using the installation wizard. Applet mode installations require additional configuration, covered in "Creating a Web Page for Applet Mode Installations" later in this chapter.

▶ To run Setup.class on an OS/2 system

1. Start an OS/2 command prompt. From the **Start** menu, point to **OS/2 System** and then point to **Command Prompts**. Click **OS/2 Window**.

2. If you downloaded the ICA Java Client, change to the directory where you saved the Setup.class file. For example, if you saved Setup.class in \downloads, type **cd \downloads**.

Note To install the ICA JAVA Client from CD-ROM, insert the CD-ROM in the CD-ROM drive. At the command prompt, type **cd *x*:**, where *x* is the letter of your CD-ROM drive, and press ENTER. Type **cd icaclnt\icajava** and press ENTER.

---

3. Type ***x jvmpgm* setup**, where *x* is the location of your JVM program and *jvmpgm* is the JVM program's name. Press ENTER. For example, if you use IBM's JVM program Java and it is located in a directory called C:\Java11, type **c:\java11\java setup**.

Note To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, type **c:\java11\java /d:user.language=fr setup**.

---

4. The ICA Java Client installation wizard appears. Follow the instructions on your screen. Application mode installations are complete once you have finished using the installation wizard. Applet mode installations require additional configuration, covered in "Creating a Web Page for Applet Mode Installations" later in this chapter.

▶ To run Setup.class on the JavaOS for Business Server

1. Download or copy the ICA Java Client installation file Setup.class from the installation media to some directory on your JavaOS for Business Server.
2. Start a command prompt. On the JavaOS for Business Server, click **Start**, **Programs**, and then click **Command Prompt**.
3. Change to the directory where you saved Setup.class. At the command prompt, type **cd *x*:**, where *x* is the drive and path to Setup.class
4. Type ***x**jvmpgm* setup**, where *x* is the full path of your JVM program and *jvmpgm* is the JVM program's name. Press ENTER. For example, if you use IBM's JVM program Java and it is located in a directory called C:\Java11, type **c:\java11\java setup**.

Note To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, type **c:\java11\java /d:user.language=fr setup**.

---

5. The ICA Java Client installation wizard appears. Follow the instructions on your screen. Application mode installations, in which you intend to configure JavaOS for Business client systems to load and use the ICA Java Client, require configuration of the server. See "Configuring the JavaOS for Business Server 2.1" later in this chapter. Applet mode installations require additional configuration, covered in "Creating a Web Page for Applet Mode Installations" later in this chapter.

▶ To run Setup.class on a JavaStation Server

1. Log in as root.
2. Start a shell.
3. If you downloaded the ICA Java Client, change to the directory where you saved the Setup.class file. For example, if you saved Setup.class in \downloads, type **cd \downloads**.

Note To install the ICA JAVA Client from CD-ROM, insert the CD-ROM in the CD-ROM drive and mount the CD. At the command prompt, type **cd *x*:**, where *x* is the letter of your CD-ROM drive, and press ENTER. Type **cd icaclnt/icajava** and press ENTER.

---

4. Type ***x/jvmpgm* setup**, where *x* is the full path of your JVM program and *jvmpgm* is the JVM program's name. Press ENTER. For example, if you use Sun's JVM program Java and it is located in a directory called /usr/bin, type **/usr/bin/java setup**.

Note To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, type **/usr/bin/java -duser.language=fr setup**.

---

5. The ICA Java Client installation wizard appears. Follow the instructions on your screen. Application mode installations, in which you intend to configure JavaStation client systems to load and use the ICA Java Client, require configuration of the NetraJ 2.0 server. See "Configuring the JavaStation Server." Applet mode installations require additional configuration, covered in "Creating a Web Page for Applet Mode Installations."

To run Setup.class on a Macintosh system

1. Download (or copy from the CD-ROM) Setup.class to your Macintosh hard drive.

   Note To locate Setup.class on the CD-ROM, insert the CD-ROM in the CD-ROM drive, then double-click the CD-ROM icon on your desktop. Double-click the **ICACLNT** folder and then double-click the **ICAJAVA** folder.

2. Drag **Setup.class** onto the JVM application on your desktop. For example, if you use the Macintosh JVM JBindery 2.0, drag **Setup.class** onto **JBindery 2.0** on the desktop.

   Note To force installation messages to display in a language other than that used by the system's default local, you must specify the system property user.language with the desired local. For example, to display French installation messages, click **Properties** in Jbindery 2.0. In the text box to the left of the equal sign type **user.language**. In the text box to the right of the equal sign, enter **fr**.

3. Execute **Setup.class**. For example, in JBindery 2.0, click **Run**.
4. The ICA Java Client installation wizard appears. Follow the instructions on your screen. In the installation wizard's **Choose Destination Location** screen, you must change the destination folder to some location on the Macintosh system's hard drive; for example, /Hard%20Disk/Citrix/JavaClient.
5. You may see an error message warning that some batch files were not created during installation. The batch file that cannot be created is a launching script used to automate starting the ICA Java Client in application mode. Macintosh users can create this batch file manually using the procedure described in "Creating a Macintosh Launching Script" later in this chapter.
6. Applet mode installations require additional configuration, covered in "Creating a Web Page for Applet Mode Installations" later in this chapter.

Creating a Macintosh Launching Script

Macintosh users can create their own client launching file for a Program Neighborhood session with the following procedure:

Note To create a launching file for a non-Program Neighborhood session, type **com.citrix.JICA** instead of **com.citrix.pn** in Step 4. Also, name your launching file jicasession instead of pnsession in Step 7.

▶ **To create a Macintosh launching script**

1. Open the folder where you installed the Citrix ICA Java Client.
2. Open the JVM program, JBindery 2.0.
3. From the list of icons on the left of the JBindery dialog box, click **Command**.
4. In the **Class Name** field, type **com.citrix.pn**.
5. In the list of icons on the left, click **Classpath**.
6. Drag **JICAEngJ.jar** from the Citrix ICA Java Client installation folder onto the **Additions to class path** field. The file name appears in the field. Click **Save Settings**. You can create a folder on your desktop called Citrix Files and save the icon there. Enter **pnsession** as the name for the launching file in the **Save Settings** field. Make sure **Save as Application** is checked. Click **Save**.
7. Click **Quit**. To run the ICA Java Client, double-click the icon you created.

Note You may want to run the ICA Java Client using the GUI Command Line feature (not available with Program Neighborhood). To use the GUI Command Line feature, create a script using the procedure above but instead of adding the **-address:*CitrixServer*** parameter and value, add the parameter **-guicmdline** in the **Optional Parameters** field. When you run this script, a dialog box appears that allows you to enter parameters to pass to the ICA Java Client. Type the address parameter and any other optional parameters in the dialog box and click **Run**. By doing this you avoid having to edit the launching file whenever you want to specify different command line parameters.

Configuring the JavaOS for Business Server 2.1

The following procedure describes how to set up the JavaOS for Business Server 2.1 to serve the ICA Java Client to its client systems. This procedure involves creating a specially formatted .zip file that contains the ICA Java Client and then using the JCT tool to configure this file as the application client systems load.

▶ **To configure the JavaOS for Business Server 2.1**

1. Create the application .zip file:
   A. On the JavaOS for Business Server, start a command prompt.
   B. Change to the directory where you installed the ICA Java Client. Make sure this directory contains the file JICAEngJ.jar.
   C. Type **mkdir classes** to create a directory called “classes” in the ICA Java Client installation directory.
   D. Type **copy JICAEngJ.jar classes** to copy JICAEngJ.jar to the classes directory.

E. Use a compression utility, such as WinZip or PKZIP, to create a zip file named "jica.zip." Jica.zip must contain at its root the directory "classes" and the classes directory must contain JICAEngJ.jar. JavaOS for Business requires jica.zip be created with this specific configuration in order to load the ICA Java Client using the Application Load Settings file.

F. Place jica.zip in the directory where you store the other application zip files (such as hotjava.zip) used by client systems.

2. Start the JSD on the JavaOS for Business Server. Use Startjsd.bat to start the JSD if it is not already running.
3. Start the JavaOS Configuration Tool (JCT) on the JavaOS for Business Server. Use Startjct.bat to start the JCT if it is not already running.
4. In the JCT, select **Users** or **Groups** and then click **Edit**.
5. In the **Available** list, select **Application Load Settings**. Click **Add**. The **Application Load Settings** dialog box appears. Enter the following values in the fields:

   A. In **Service URL**, type **file:/*SERVICES*/AppLoad.jar**, where *SERVICES* is the NFS mount point where the loadable services are located. You can change the path *SERVICES* as necessary, but not the service name (AppLoad.jar).

   B. In **Main class**, type **com.citrix.pn**.

   C. In **Base Classpath URL**, type **file:/*APPS*/jica.zip**, where *APPS* is the NFS mount point where jica.zip is located.

   Note If you have not defined an NFS mount point for this directory, please follow the directions in *JavaOS for Business Installation and Planning*, provided online with your JavaOS for Business Server.

   D. In **Package Names**, leave the field blank (delete any value that appears in the field).

6. Click **Save** to exit the **Application Load Settings** dialog box.
7. Reboot the client system and log on as the modified user to load the Citrix ICA Java Client.

For more information, see *JavaOS for Business Network Operations*, provided online with your JavaOS for Business Server.

Configuring the JavaStation Server

The following procedure describes how to set up the JavaStation server to serve the ICA Java Client to its client systems. This procedure involves creating a specially formatted .jar file that contains the ICA Java Client and then using the NetraJ 2.0 server software to configure this file as the application client systems load.

To configure the NetraJ 2.0 Server software

1. Create the application .jar file:
   A. On the NetraJ 2.0 Server, start a shell.
   B. Change to the directory where you installed the ICA Java Client. Make sure this directory contains the file JICAEngJ.jar.
   C. Type **mkdir classes** to create a directory called “classes” in the ICA Java Client installation directory.
   D. Change to the classes directory, (type **cd classes**). Type **jar -xvf ..\JICAEngJ.jar** to extract the file JICAEngJ.jar.
   E. Change to the directory above classes (type **cd ..**).
   F. Create a .jar file named “jica.jar” by typing **jar -cvf jica.jar classes**. Jica.jar must contain at its root the directory “classes” and the classes directory must contain ICA Java Client class files. JavaOS requires jica.jar be created with this specific configuration in order to load the ICA Java Client as an application.
   G. Place jica.jar in the directory that is accessible to the HTTP server where you store the other application .jar files (such as Views.jar) used by client systems.
2. In NetraJ 2.0 Network Computer Server Administration, select **Client Application Administration** under the Advanced NC Server Administration section and then click **Add**.
3. Enter the following values in the fields:
   A. In **Application Name**, type a title such as CitrixJICA to appear in an application menu item.
   B. In **Main Class of the Application**, type **com.citrix.pn**.
   C. In **Application Archive (ZIP) Path**, type **http://*server*/*appsdir*/jica.jar**, where *server* is the server name and *appsdir* is the path where jica.jar is located. For example, if jica.jar is located on an HTTP server named \\Webserver in the /netra/hjv/classes directory off the Web server’s root directory, type **http://Webserver/netra/hjv/classes/jica.jar**.
   D. In **Home Property of the Application**, enter **null**.
   E. Click **OK** to accept the changes and exit the **Client Application Administration** page.
4. In NetraJ 2.0 Network Computer Server Administration, select **Network Computer Administration** under the Network Computers Administration section and then click **Modify** for the respective network computer.
   A. In **Default Application**, select the name you entered previously in **Application Name** (in this example, CitrixJICA).
   B. Click **OK** to accept the changes and exit.

5. Reboot the client system and log on as the modified user to load the Citrix ICA Java Client.

   For more information, see the *NetraJ 2.0 Administrator's Guide*.

Creating a Web Page for Applet Mode Installations

After running Setup.class to install the ICA Java Client on your Web server, you must create an HTML page that client system users can visit to initiate ICA sessions. Use the following procedure:

▶ To create an HTML page that users can visit to launch the ICA Java Client in applet mode

- Create an HTML page like the example below and place the file in the Web server's Web root directory with the ICA Java Client program files:

```
<html>
<body>
<applet code=com.citrix.JICA.class
codebase=http://webserver/directory
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

  The parameters that need to be modified for your environment are:

  - **Codebase**. The codebase parameter is a URL that specifies the location of the ICA Java Client class files. This parameter is necessary only if the class files and the HTML page are located in different directories or on different Web servers. You do not need to specify this parameter if the class files and the HTML page are in the same directory on the Web server. For example, if the HTML page is in the \Page directory on the Web server and the ICA Java Client class files are in the \Applet subdirectory of the Web server Web root directory, enter `codebase=/Applet`. If the class files and the HTML page used to access the ICA Java Client are located on different Web servers, enter the full URL of the class files; for example, `codebase=http://icasrv/jclient`.
  - **Archive**. The archive parameter specifies the name of the archive containing the ICA Java Client class files to use. Different browsers support different archives, and more than one archive can be specified (separate multiple archives with a comma and no spaces). See "Using the ICA Java Client as an Applet" in Chapter 3, "Configuring the ICA Java Client" for information on which archives work with several popular Web browsers.

See “Using the ICA Java Client as an Applet” in Chapter 3 for more information about configuring the ICA Java Client for applet mode use. See “ICA Java Client Parameters” in Chapter 3, “Configuring the ICA Java Client” for a list of the parameters that can be used with the ICA Java Client.

Uninstalling the ICA Java Client

Your installation includes an uninstall program that you can use to remove the ICA Java Client application from individual client computers. The uninstall program does not apply to client computers that use the ICA Java Client in applet mode. The uninstall program removes all ICA Java Client files from the client system.

Note These instructions apply to ICA Java Client installations on both client and server systems.

On Windows 95/98/NT systems, the installation process adds an entry to Add/Remove Programs in Control Panel. Use this to uninstall the ICA Java Client or use the procedure below.

- **To uninstall the ICA Java Client application from a Windows, UNIX, OS/2, or JavaOS for Business client system**

  1. Start a command prompt. (For JavaOS for Business installations, this procedure is done on the JavaOS for Business server.)
  2. Change to the directory where the ICA Java Client is installed.
  3. Type ***x**jvmpgm* uninstall** at the command prompt, where *x* is the full path of your JVM program and *jvmpgm* is the JVM program’s name. For example, if you use the Microsoft JVM program Jview and it is located in a directory called C:\Windows, type **c:\windows\jview uninstall**. The ICA Java Client uninstall wizard appears. Follow the directions on your screen.

- **To uninstall the ICA Java Client from a Macintosh client system**

  1. Locate the directory where the ICA Java Client is installed.
  2. Drag **Uninstall.class** onto the JVM application on your desktop. For example, if you use the Macintosh JVM program JBindery 2.0, drag **Uninstall.class** onto **JBindery 2.0** on the desktop.
  3. Execute **Uninstall.class**. For example, in JBindery 2.0, click **Run**. The ICA Java Client uninstall wizard appears. Follow the directions on your screen.

Removing .Ini Files and Program Neighborhood Cache Files

When you remove the ICA Java Client, user-specific and program-specific .ini files, and Program Neighborhood cache files are not removed. Repeat the procedure below for each user to remove these files.

To remove the .ini files and Program Neighborhood cache files, first locate the files (see "Specifying .Ini Files on the Command Line" in Chapter 3 to ascertain the location of the .ini files; the Program Neighborhood files are located in the same directory). Next, delete all the .ini and .ser files. The commands you use depend on which platform you're running. For example, on a UNIX system you might perform the following steps:

1. At a command prompt, type **cd ~user/.citrix** where *user* is the login name of the user for whom you want to remove the files.
2. Type **rm -f *.ini**.
3. Type **rm -f *.ser**.

CHAPTER 3

Configuring the Citrix ICA Java Client

This chapter describes how to configure and start the ICA Java Client for applet and application mode installations on various platforms.

Tasks to Complete

In this chapter you will complete the following tasks.

Applet mode installations:

- Choose an appropriate archive for applet mode use—(5 minutes)

Application mode installations:

- Start the ICA Java Client in application mode—(5 minutes)

This chapter also describes some optional session customization tasks:

- Specifying parameters
- Configuring the ICA Java Client for use with Program Neighborhood
- Configuring application sets
- Configuring the ICA Java Client for use in a specific national language
- Configuring COM port mapping
- Configuring client audio
- Configuring the ICA Java Client to connect to a server across a firewall
- Configuring auto-reconnect and session termination
- Configuring printers
- Configuring event logging
- Configuring Business Recovery
- Configuring keyboard support
- Configuring hotkeys
- Specifying .ini files on the command line

Using the ICA Java Client as an Applet

The last step of ICA Java Client setup is to configure your HTML pages to accommodate the requirements of the client systems that will download and run the ICA Java Client. When tailoring the HTML pages, there are two major considerations: which Java class file archives are supported by users' browsers and how to handle Java network security restrictions. Both issues are addressed by selecting the proper Java class file archive for your particular network setup.

Once the ICA Java Client is properly installed and configured, using it involves simply opening a browser and clicking the link in the HTML page to start an ICA session.

ICA Java Client Components

Because different implementations of the Java virtual machine (JVM) support different ways of storing class files, the ICA Java Client presents the class files in three different archive formats. Using these archives allows users to connect to a Citrix server that is not the same machine as the Web server, write log files to the client's drive, and print to the client's printers.

Note Using signed archives allows applet mode users to work within the Java network security restriction that prevents users from connecting to other computers on the network or writing files to the client's hard disk.

When running the ICA Java Client in either application or applet mode, use an archive format supported by your JVM or the JVM of your Web browser. If your JVM does not support any of these archive formats, the ICA Java Client can also be run in application and applet mode from the extracted class files. Use the appropriate utility to extract the class files from an archive. A detailed description of the files follows:

- **JICAEngM.cab**. A signed cabinet (cab) file containing the ICA Java Client class files. Use this file with Internet Explorer on Windows platforms (you must set Internet Explorer's safety level to Medium).
- **JICAEngN.jar**. A signed jar file containing the ICA Java Client class files. Use this file only with Netscape Navigator; other browsers will fail to load the ICA Java Client from this archive.

- **JICAEngJ.jar**. A signed jar file containing the ICA Java Client class files. Use this file with browsers other than Internet Explorer and Netscape Navigator.

The following files are also extracted during ICA Java Client installation:

- **launching files**. Platform-specific script files. If you use setup.class to install the ICA Java Client, the setup program detects the operating system and installed JVM of the client machine during ICA Java Client installation and creates script files used to run the ICA Java Client in application mode (jicasession and pnsession for UNIX computers; jicasession.bat and pnsession.bat for Windows NT, Windows 98, and Windows 95; and jicasession.cmd and pnsession.cmd for OS/2 computers). If you use JavaEnt.zip, JavaEnt.sea.hqx, or JavaEnt.tar.gz, the archive contains script files (jicasession and pnsession for UNIX computers; jicasession.bat and pnsession.bat for Windows NT, Windows 98, and Windows 95; and jicasession.cmd and pnsession.cmd for OS/2 computers) that may have to be edited before you can use them to launch the ICA Java Client. In either case, Macintosh users must create their own script file. See “Creating a Macintosh Launching Script” in the “Application Installation” section for more information on creating a Macintosh launching file.
- **Help**. A directory containing HTML help files.
- **Install.idb**. A file used by InstallShield during setup. This file exists if you used the installation archive setup.class to install the ICA Java Client.
- **Uninstall.class**. Uninstall program for the ICA Java Client. This file exists if you used the installation archive setup.class to install the ICA Java Client.

Supported Archives

Which archive you use to run the ICA Java Client in applet mode depends upon the types of archive formats supported by the client system's browser. The following list specifies which archives are supported by several popular browsers.

Browser	Supported archives
Netscape 4.04 for OS/2	jar
Netscape Communicator 4.05 with AWT 1.1 support or Netscape Communicator 4.06+ for Windows 95, Windows 98, Windows NT 4.0, and Solaris 2.6	jar
Internet Explorer 4.0x for Windows 95, Windows 98, Windows NT 4.0	cab
Internet Explorer 4.0x for non-Windows platforms	jar
HotJava Browser 1.1 for Windows 95, Windows 98, Windows NT 4.0, JavaOS for Business, and Solaris 2.6	jar
HotJava Views for Sun JavaOS	jar

For example, to use the ICA Java Client with Netscape 4.0 to connect to a Citrix server named CitrixServer, specify JICAEngN.jar in the applet tag. Your HTML page might look like this:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

To use the ICA Java Client with Internet Explorer 4.0 to connect to a machine named CitrixServer, specify JICAEngM.cab as the archive. When using cab files, the parameter to add is not archive but cabbase. Your HTML page might look like this:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=cabbase value=JICAEngM.cab>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

Note Internet Explorer users must set their safety levels for active content to **Medium** to use the cab archive.

Applet Security

Due to network security restrictions imposed by Java, many Web browsers do not permit users to connect to other computers on the network when using Java applets. When the ICA Java Client attempts to make a connection to the Citrix server specified in the HTML page, the Java security manager detects the attempt to connect to another computer and cancels the operation. The result of this security restriction is that under normal conditions, a client system can connect to a Citrix server only if it is also the same machine as the Web server that contains the ICA Java Client class files.

One way to work around this restriction is to use signed archives. The signature confirms that the files being downloaded came from Citrix and have not been altered since the signature was applied. Many Web browsers allow connections to a Citrix server that is not the same machine as the Web server if a signed archive is used.

Tip As a second Java security restriction workaround, a Web server can be installed on the Citrix server. This option might be preferable in the case where client systems have browsers that do not support signed archives. The HTML pages that users visit to initiate ICA sessions can be kept on another Web server (for example, your Intranet Web server) and point to the class files on the Citrix server/Web server machine by using the codebase parameter in the HTML file. See “Creating a Web Page for Applet Mode Installations” in Chapter 2 for a description of the codebase parameter.

To connect to Citrix servers that are not the same machine as the Web server supplying the client system with the applet code, the client system’s browser must support signed archives (jar or cab files). The list below specifies which ICA Java Client signed archives can be used with some popular browsers:

Browser and platform	Signed archive
Netscape Communicator 4.05 with AWT 1.1 support or Netscape Communicator 4.06+ for Windows 95, Windows 98, Windows NT 4.0, and Solaris 2.6	JICAEngN.jar
Internet Explorer 4.0x for Windows 95, Windows 98, and Windows NT 4.0	JICAEngM.cab
Internet Explorer 4.0x for non-Windows platforms	JICAEngJ.jar
HotJava Browser 1.1 for Windows 95, Windows 98, Windows NT, JavaOS for Business, and Solaris 2.6	JICAEngJ.jar
HotJava Views for Sun JavaOS	JICAEngJ.jar

Notes Internet Explorer users must set their safety levels for active content to **Medium** to use the signed cab file.

Although Netscape 4.04 for OS/2 supports running Java applets from .jar files, it does not support .jar file signatures. Netscape 4.04 for OS/2 users can connect to a Citrix server only if it is also the same machine as the Web server that contains the ICA Java Client class files.

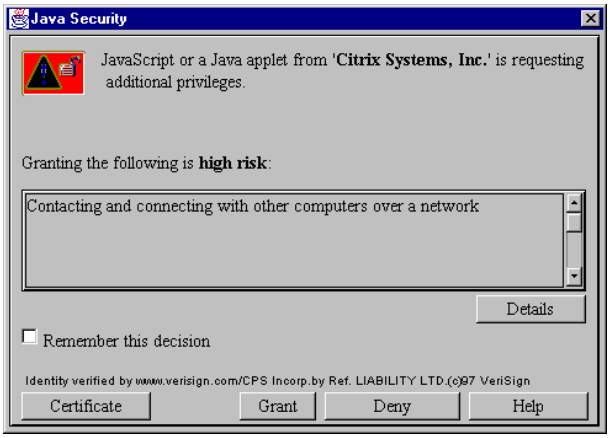
Although Internet Explorer 4.0x for Solaris 2.6 and Macintosh support running Java applets from cabinet files, they do not support cabinet file signatures. Users of these browsers can connect to a Citrix server only if it is also the same machine as the Web server that contains the ICA Java Client class files.

In the HTML page, specify which signed archive you want to use. When attempting a connection to the Citrix server, the user is prompted with the Citrix signed certificate. Once the user accepts the signature, the connection is permitted.

For example, to use the ICA Java Client with Netscape 4.0 to connect to a Citrix server named CitrixServer that is not the same machine as the Web server, specify JICAEngN.jar in the applet tag. Your HTML page might look like this:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

When visiting an HTML page like the example above for the first time, the **Java Security** dialog box appears:

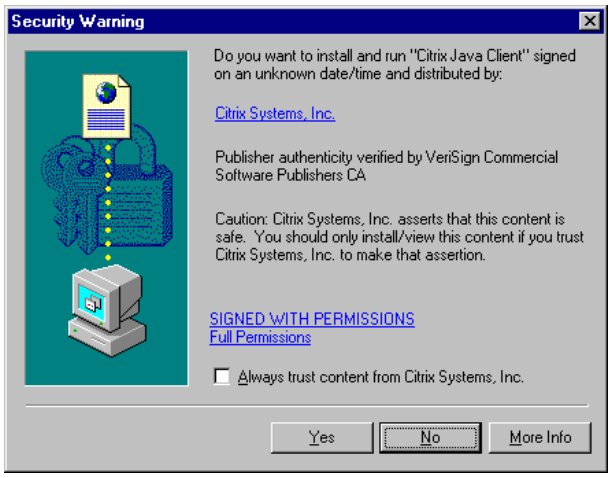


Click **Remember this decision** and then click **Grant** to begin the ICA session.

To use a signed cab file, include the statement `<param name=cabbase value=JICAEngM.cab>` in the HTML file and remove the archive parameter. For example:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=cabbase value=JICAEngM.cab>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

When users visit this HTML page for the first time, the following dialog box appears:



Click **Always trust content from Citrix Systems, Inc.** and then click **Yes** to download and run the ICA Java Client.

You can configure a single HTML page so that users with Microsoft Internet Explorer 4.x and Netscape Communicator 4.x can connect and run the ICA Java Client:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=cabbase value=JICAEngM.cab>
<param name=address value=CitrixServer>
</applet>
</body>
</html>
```

In the example above, the Netscape users download and run JICAEngN.jar while Internet Explorer users use JICAEngM.cab.

When using the signed .jar file for HotJava Browser (JICAEngJ.jar), client system users must manually configure their browsers. The following procedure describes how to use the signed .jar file with HotJava Browser:

To configure HotJava Browser to use a signed .jar file

1. Start HotJava Browser. Enter the URL of the ICA Java Client applet.
2. From the HotJava Browser **Edit** menu, click **Preferences** and then click **Applet Security**.
3. On the **Applet Security** page, click **Advanced**. Select the Citrix certificate in the list and then click **Details**.

4. In the **Certificate Details** dialog box, select **Fingerprint Verified** and then click **OK**.
5. Using the browser's browse button, return to the URL of the applet and click **Reload**.
6. Click on **Click Here to Connect**.
7. When the **Security Violation** dialog box appears, click **Allow All Connects Without Warning** and then click **OK**.
8. Repeat Step 7 as necessary.

Accepting the Citrix Java Certificate on Macintosh Systems with MRJ 2.1.2

To allow the ICA Java Client running in applet mode to write preferences and configuration files to the hard disk in your machine, you must accept the signed Citrix Java certificate as trusted. Use the Javakey utility to accept the Citrix Java Certificate. The Javakey utility is located in the Java/MRJ SDK 2.1/Tools/JDK Tools folder. You can also click **Find** on the **File** menu to locate this utility by searching for **javakey**.

▶ **To use Javakey to accept the Citrix Java certificate**

1. Start Javakey.
2. Click **Create Identity** from the popup menu.
3. In **Entity,** type **ICA**. Click **Trusted**.
4. Click **Do Javakey**. This creates an identity for the certificate.
5. Select **Import Certificate** from the popup menu. The **Identity or Signer** textbox should contain **ICA**.
6. In the **Certificate Source File** list, click **Select File**. Use the file selector to select **CTXHotJavaCertificate.x509**. This file is in your ICA Java Client installation directory, together with the ICA Java Client's .jar and .cab files.

   Note If you are downloading the ICA Java Client from the Web, then the Web site provider should have provided a link to the certificate file for you.

7. Once you have selected the certificate file, it will appear in the **Certificate Source File** list. Click **Do Javakey** to im`port the certificate.

The Citrix Java certificate for the ICA Java Client is now accepted, and the client applet is trusted by your computer.

Using the ICA Java Client as an Application

Once you have installed the ICA Java Client, use the included launching script to start an ICA session. You can also use optional parameters with the launching script to configure ICA session properties.

Starting the ICA Java Client Application

The Citrix ICA Java Client includes script files that can be used to automate starting sessions in application mode. This section describes their use. See "Starting the ICA Java Client Without Using Script Files" for information on running the ICA Java Client in application mode without using script files.

If you used JavaEnt.zip, JavaEnt.sea.hqx, or JavaEnt.tar.gz to install the ICA Java Client, your installation directory contains one of the following script files: jicasession for UNIX computers, jicasession.bat for Windows computers, or jicasession.cmd for OS/2 computers. Each of these files contains a command line that specifies a certain JVM to use when running the ICA Java Client. Edit the name of the Java program (and any other JVM specific parameters) listed in the script file to match the Java program used by your client system. By default, the script files specify SUN's JVM program java. If you use another JVM, for example Microsoft's jview, edit the script to specify **jview** instead of **java**. Additionally, to use the script files, your JVM must be in your path.

If you used setup.class to install the ICA Java Client, the setup program detected the client machine's operating system and JVM during installation and created a script file usable with that platform and JVM. This file should not require any editing.

Macintosh users must create their own launching file. See "Creating a Macintosh Launching Script" in Chapter 2, "Installing the ICA Java Client" for information on creating a launching script for Macintosh client computers.

The following procedures describe how to run the ICA Java Client on different platforms using script files:

▶ To start the ICA Java Client in application mode on a Windows client PC

1. Start a command prompt. On Windows 95 and Windows 98, click **Start** and then click **Run**. In the **Run** dialog box, type **command** and press ENTER. On Windows NT 4.0, click **Start** and then click **Run**. In the **Run** dialog box, type **cmd** and press ENTER.
2. Change to the directory where the ICA Java Client is installed. For example, if the ICA Client was installed in a directory called C:\Citrix\ and your command prompt displays **D:>**, type **C:** and press ENTER, and then type **cd \Citrix** and press ENTER.

3. Type **jicasession -address:*CitrixServer*** at the command prompt, where *CitrixServer* is the name of the Citrix server to which you are connecting.

To start the ICA Java Client in application mode on a UNIX client system

1. Start a shell. Change to the directory where the ICA Java Client was installed.
2. Type **jicasession -address:*CitrixServer*** at the command prompt, where *CitrixServer* is the name of the Citrix server to which you are connecting.

To start the ICA Java Client as an application on an OS/2 client system

1. Start an OS/2 command prompt. From the **Start** menu, point to **OS/2 System** and then point to **Command Prompts**. Click **OS/2 Window**.
2. Change to the directory where the ICA Java Client is installed. For example, if the ICA Client was installed in a directory called C:\Citrix\ and your command prompt displays **D:>**, type **C:** and press ENTER, and then type **cd \Citrix** and press ENTER.
3. Type **jicasession -address:*CitrixServer*** at the command prompt, where *CitrixServer* is the name of the Citrix server to which you are connecting.

To start the ICA Java Client as an application on a Macintosh client system

Macintosh users must create their own launching file as described in "Creating a Macintosh Launching Script" in Chapter 2, "Installing the ICA Java Client."

1. Open the folder containing the Macintosh launching script.
2. Double-click the launching script.

To start the Citrix ICA Java Client as an application on a JavaOS for Business client system

The following procedure assumes the JavaOS for Business server has been configured to load the ICA Java Client when an individual client system boots. See "Configuring the JavaOS for Business Server 2.0" or "Configuring the JavaOS for Business Server 2.1" in Chapter 2, "Installing the ICA Java Client" for further information.

Note The procedure for starting the ICA Java Client on a JavaOS for Business client system does not use an ICA Java Client script file.

---

1. Boot the JavaOS for Business client system.
2. Log on as a user who has been configured to run the ICA Java Client.

To start the Citrix ICA Java Client as an application on a JavaStation client system

The following procedure assumes the NetraJ 2.0 Server software has been configured to load the ICA Java Client when an individual client system boots. See "Configuring the JavaStation Server" for further information.

Note The procedure for starting the ICA Java Client on a JavaStation client system does not use an ICA Java Client script file.

---

1. Boot the JavaStation client system.
2. Log on as a user who has been configured to run the ICA Java Client.

Specifying Parameters

Use the following syntax to specify multiple parameters in application mode:

```
jicasession [-parametername:value] [-parametername:value]...
```

where jicasession is the launching file, *parametername* is the name of a parameter, and *value* is the value for the parameter. You can specify multiple parameters in any order. The address parameter is the only required parameter. To pass parameters to the ICA Java Client, use a dash (-), followed by the parameter name, followed by a colon (:), followed by the parameter value.

For example, to connect to the Citrix server named CitrixServer as user Barry from a command prompt using the script file jicasession, type:

```
jicasession -address:CitrixServer -username:Barry
```

You can also connect to a Citrix server using an ICA file. Specify the name of the ICA file on the command line without a dash, followed by any optional parameters. For example, to start a session using an ICA file named Connection.ica, type:

```
jicasession Connection.ica -username:Barry
```

Important If you specify both an ICA file and command line parameters, the command line parameters override the settings in the ICA file. For example, in the command above, the username Barry overrides any username specified in the ICA file.

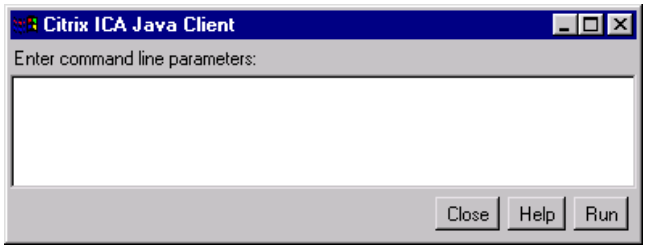
For more information about parameters that can be used with the ICA Java Client, see "ICA Java Client Parameters" later in this chapter.

GUI Command Line

The ICA Java Client includes a feature called the GUI Command Line (not available when running Program Neighborhood). The *GUI Command Line* is a graphical tool that simulates the command line for users of client computers that lack a command line (for example, Macintosh systems) or for users who find the command line cumbersome to use. Use the GUI Command Line dialog box to pass parameters to the ICA Java Client. The GUI Command Line requires the same syntax as the command line. The GUI Command Line can be invoked in two ways.

▶ **To start the GUI Command Line**

- Specify the parameter -guicmdline on the command line (without a value). For example, type: **jicasession -address:CitrixServer -guicmdline**. The **GUI Command Line** dialog box appears:



  Type any additional parameters in the **Enter command line parameters** field. Use the same syntax used on the command line.

- You can also start the GUI Command Line by specifying no parameters on the command line. For example if you type **jicasession** without any parameters, the **GUI Command Line** dialog box appears.

Starting the ICA Java Client Without Using Script Files

You can also run the ICA Java Client in application mode without using a script file. The class file to run is com.citrix.JICA. When starting the ICA Java Client without a script file, use the following syntax:

```
jvmpgm com.citrix.JICA [-parametername:value][-parametername:value]...
```

where *jvmpgm* is the name of your JVM program, *parametername* is the name of a parameter, and *value* is the value for the parameter. You can specify multiple parameters in any order. The address parameter is the only required parameter. To pass parameters to the ICA Java Client, use a dash (-), followed by the parameter name, followed by a colon (:), followed by the parameter value.

Important When starting the ICA Java Client without a script file, you must have the ICA Java Client class files in your classpath and your JVM in your path. For example, if the ICA Java Client class file archive JICAEngJ.jar is stored in a directory called \Application on drive C, set your classpath to C:\Application\JICAEngJ.jar. If the class files have been extracted from an archive, specify the location only; for example, C:\Application. To place the JVM in the path, set the path to the location of the JVM.

For example, to connect to a Citrix server named CitrixServer from a command prompt using the Sun Microsystems JVM (named java), type:

```
java com.citrix.JICA -address:CitrixServer
```

ICA Java Client Parameters

The following parameters can be specified in the HTML file or on the command line to provide additional features and customization. The Address parameter is the only required parameter; other parameters in this list are optional.

Parameter	Description
Address	The address of the Citrix server or the name of the published application. (If a published application is entered as an address value, TCPBrowserAddress and InitialProgram parameters must also be specified. TCPBrowserAddress must be specified if your client is on a different subnet than your server.)
TCPBrowserAddress	The address of a Citrix server. This parameter is used when your client is on a different subnet than the Citrix server. This parameter is also used to designate groups of primary and backup servers. See "Business Recovery" later in this chapter for more information.
Username	The user name to use during login.
Domain	The name of the domain for the user name.
Hotkey2 Hotkey4 Hotkey5 Hotkey6 Hotkey7 Hotkey8	These parameters set hotkeys that can be used to control various client functions. See "Hotkeys" later in this chapter for more information.
KeyboardLayout and KeyboardType	These parameters are used to specify the type of keyboard.
LocalClientPrinters	This parameter is used for passing information to the server about client printers.

Parameter	Description
LogFile LogAppend LogConnect LogErrors LogTransmit LogReceive LogKeyboard	These parameters are used to implement Event Logging. See "Event Logging" later in this chapter for more information.
Password	This is the password of the user. The Password parameter cannot be used to specify an encrypted password. To specify an encrypted password, use an ICA file or .ini file that contains an encrypted password.
InitialProgram	The name of the initial program to run after connecting to the Citrix server.
WorkDirectory	The path of the working directory where the initial program is run after the user connects to the Citrix server.
ICAPortNumber	The default ICA port number is 1494. A different port number can be specified using this parameter or by appending the port number to the address value; for example, "CitrixServer:1495."
UseAlternateAddress	Use an alternate server address across a firewall. This parameter is always preceded by "user.wfclient." (If you use this parameter, the parameter TCPBrowserAddress must also be specified.)
ClientAudio	Use this parameter to enable client audio. The values are **on** and **off**. The default is **off**.
Width	The width of the ICA session window. The default value in application mode is 640. There is no default in applet mode. This parameter is used differently in applet and application modes. See the examples below.
Height	The height of the ICA session window. The default value in application mode is 480. There is no default in applet mode. This parameter is used differently in applet and application modes. See the examples below.

The following parameters are used only in applet mode:

Parameter	Description
EndSessionTimeout	By default, when users leave a page containing an open ICA session for more than 300 seconds (five minutes), the session is disconnected. Use this parameter to specify a different value in seconds.
Border	This parameter turns the border around the ICA session in the browser window on or off. Values for this parameter are **on** or **off**. The default value is **off**.

Parameter	Description
BorderWidth	This parameter allows the user to specify a border width in pixels. The default value is **6**.
Cabbase	This parameter specifies the .cab file to be used when using Internet Explorer 4.0. Its value is JICAEngM.cab. See "Using the ICA Java Client as an Applet" earlier in this chapter for more information.
Start	This parameter controls the ICA Java Client's behavior when you start a session. The values are **manual** (default) and **auto**. See "Auto-reconnect and Session Termination" later in this chapter for more information.
End	This parameter controls the ICA Java Client's behavior when you terminate a session. The values are **manual** (default), **auto**, **terminate**, and *url*. See "Auto-reconnect and Session Termination" later in this chapter for more information.
Language Country	These two parameters are used together to cause the ICA Java Client's user interface components to appear in a language other than the language of the client machine; for example, to display ICA Java Client components in French on a German machine. By default, the ICA Java Client displays the components in the language of the client machine. See "Starting the ICA Java Client in a Specific National Language" for more information.
Icafile	This parameter is used to specify an ICA file for the ICA Java Client to use. The value entered must be a valid URL.

The following parameters are used only in application mode:

Parameter	Description
Clientsn	Use the client serial number parameter when running the ICA Java Client in application mode to connect to a MetaFrame for Terminals or *WINFRAME* for Terminals server. (A client serial number cannot be used when running the ICA Java Client as an applet.)
Iniappsrv	The URL of an appsrv.ini file to use with the ICA Java Client.
Inimodule	The URL of a module.ini file to use with the ICA Java Client.
Iniwfclient	The URL of a wfclient.ini file to use with the ICA Java Client.
IniDir	The URL of a directory containing Ini files to use with the ICA Java Client.
Guicmdline	Use this parameter to display the GUI Command Line dialog box. The GUI Command Line dialog box is intended for use as a convenient tool for those platforms that lack a command line or for users who find the command line cumbersome to use. This parameter is specified without a value. For example, type **jicasession -guicmdline**.

See “Specifying .Ini Files on the Command Line” later in this chapter for more information on specifying .ini files. See “GUI Command Line” earlier in this chapter for more information on the GUI Command Line dialog box.

Examples

The following examples illustrate a sample connection using several parameters from the parameter lists above. The first example is in application mode and the second is in applet mode.

To connect to a Citrix server named CitrixServer using a launching file named jicasession as user tonycarp from the test domain running Microsoft Word as an initial program with a window resolution of 1024 x 768, type (all on one line):

```
jicasession -address:CitrixServer -username:tonycarp -domain:test
"-initialprogram:C:\Program Files\Word\Winword.exe"
-height:768 -width:1024
```

Note Parameters can be designated in any order. When entering a parameter value that contains a space, place the entire parameter (parameter name and value) in quotation marks.

To make the same connection in applet mode with the additional parameters of a 600 second end session timeout, a border of 10 pixels, and with a specified ICA file named “Connection.ica” that is stored in the C:\Citrix directory, create an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=1024 height=768>
<param name=address value=CitrixServer>
<param name=username value=tonycarp>
<param name=domain value=test>
<param name=endsessiontimeout value=600>
<param name=border value=on>
<param name=borderwidth value=10>
<param name=initialprogram
value="C:\Program Files\Word\Winword.exe">
<param name=icafile value=c:\citrix\connection.ica>
</applet>
</body>
</html>
```

Published Applications

To connect to a published application, use the following command syntax:

```
jicasession -address:Publishedappname -tcpbrowseraddress:ServerName
-InitialProgram:#Publishedappname
```

where *Publishedappname* is the name of the published application, *ServerName* is the name of a Citrix server used for TCP browsing, and *#Publishedappname* is the name of the published application preceded by a pound sign (#).

In applet mode, specify:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=Publishedappname>
<param name=tcpbrowseraddress value=CitrixServer>
<param name=initialprogram value=#Publishedappname>
</applet>
</body>
</html>
```

Using the ICA Java Client with Program Neighborhood

Program Neighborhood is your view of the published applications you are authorized to access. When you start the ICA Java Client with the Program Neighborhood interface, you see groups of applications called application sets. This section of the ICA Java Client Help describes how to start and use the ICA Java Client with Program Neighborhood.

Starting the ICA Java Client in Program Neighborhood Mode as an Application

When starting the ICA Java Client with Program Neighborhood as an application (from the command line), use the class file com.citrix.pn; for example, jvmpgm com.citrix.pn [-parametername:value] [-parametername:value]...

where [-parametername:value] are additional, optional parameters.

If this is the first time you are making a connection with Program Neighborhood, then the Select Application Set window appears. Select an application set from the drop-down menu and click **OK**. The Citrix Program Neighborhood Login box appears (unless your login information has been previously saved). Enter your user name, password, and domain and click **OK**. The Program Neighborhood window appears with a view of all the applications you can run.

Starting the ICA Java Client in Program Neighborhood Mode as an Applet

To start an ICA Java Client in Program Neighborhood mode as an applet, use the class file com.citrix.pn.class instead of com.citrix.JICA.class in your HTML file.

For example, to use the ICA Java Client with Internet Explorer 4.0 to start an ICA Client in Program Neighborhood mode connected to a machine named Server1, create an HTML file similar to this:

```
<html>
<body>
<applet code=com.citrix.pn.class
width=640 height=480>
<param name=cabbase value=JICAEngM.cab>
<param name=address value=Server1>
</applet>
</body>
</html>
```

Note In applet mode, the Java Client's Program Neighborhood interface is slightly different from the Win32 version. The Java version uses only a tool bar while the Win32 version has both a menu bar and a tool bar. The buttons on the Java client's tool bar work the same as the similarly named menu items in the Win32 version.

For additional examples on creating HTML files for starting an ICA Java Client as an applet, see "Using the ICA Java Client as an Applet" earlier in this chapter. For instructions on how to configure connections to server and published applications, see "Using Application Sets" below.

Using Application Sets

An application set is a user's view of the applications published on a given server farm, which that user is authorized to access. Applications published in an application set are pre-configured for such session properties as window size and colors and supported level of encryption, audio, and video. If these settings are not required to run the published application (such as a required level of encryption), they can be changed on the client machine at the application set level. The following procedures describe how to configure the properties and settings of an application set.

Configuring Connection Properties for Application Sets

To configure the connection properties for application sets

1. Start Program Neighborhood. Click **Up** in the toolbar or click on Application Sets until you see the available application sets.
2. Select the application set you want to configure.
3. In the Program Neighborhood toolbar, click **Properties** to display the **Properties** dialog box.
4. Click the **Connection** tab to display the **Connection** page. From the **Connection** page, you can configure the following:
   - **Connection Type**: Choose a connection type for the application set. Select **Local Area Network** to connect to the Citrix server over a local network that covers a confined geographic area (such as an office building or complex). Select **Wide Area Network** to connect to the Citrix server over a network that covers a wide geographical area.
   - **Server Location**: Server location (also called the ICA Browser or server browsing) provides a method for a user at a network-connected Citrix ICA Client computer to view a list of all Citrix servers on the network that have ICA connections configured for the network protocol, and a list of all published applications.

Tip Set a specific server address for the Citrix server that functions as the master ICA Browser when your network configuration uses routers and gateways, or to eliminate broadcasts on your network.

   - **Network Protocol**: The **Network Protocol** field instructs the Citrix ICA Client what low-level protocol to use to connect to the Citrix server. The Citrix ICA Java Client supports only the TCP/IP protocol.
   - **Server Group**: Use the **Server Group** and **Address List** fields to create lists of specific servers that you want to designate as primary and backup servers for connecting to application sets. Backup server groups provide business recovery for your client in the event that you cannot contact any servers in your primary group.

     You can define up to three groups of Citrix servers to which you want to connect: a primary and two backups. Each group can contain from one to five servers. When you specify a server group for your client, the client attempts to contact all the servers within that group simultaneously; the first server to respond is used to get the address of the Master Browser.

     Use this field to designate whether the servers entered in the Address List field belong to your Primary, first backup (Backup 1), or second backup (Backup 2) group.

- **Address List**: Use this field to view and change the list of Citrix servers used in the selected server group. If you have not selected any servers, [Auto-Locate] is selected by default.
  - Click **Add** to add a Citrix server to your server group's address list.
  - Click **Delete** to remove the selected Citrix server from the list.
  - If **[Auto-Locate]** is selected, the first server is located automatically.

Configuring Default Options for Application Sets

To configure the default options for application sets

1. Start Program Neighborhood.
2. Select the application set you want to configure.
3. In the Program Neighborhood toolbar, click **Properties** to display the **Properties** dialog box.
4. Click the **Default Options** tab to display the **Default Options** page. From the **Default Options** page, you can configure the following:

Note If an option has a **Use Server Default** checkbox, then the server can specify the value for that option. To override the server default, click the checkbox to remove the check. You can now select a new value from the available choices.

- **Use data compression**: Data compression reduces the amount of data that needs to be transferred but requires additional processor resources to compress and decompress the data. If your connection is bandwidth-limited, enabling data compression increases performance.
- **Enable sound**: Check this box to enable sound support. The client computer must have a sound card installed. Published applications can then play sounds on the client. The Citrix ICA Java Client supports only **Medium** sound quality.
- **Encryption level**: Select the level of encryption for the ICA connection. The Citrix ICA Java Client does not support encryption levels higher than **Basic**.
- **Window Size**: This field specifies the window size in which the published application runs.
- **Window Colors**: This field specifies the number of colors displayed (256).

If the server-configured default settings are being used by any of the properties, the **Use Server Default** box is checked. To change the settings, de-select this check box and choose new settings. Click **OK** to save these settings.

Configuring Login Properties for Application Sets

To configure login properties for application sets

1. Start Program Neighborhood.
2. Select the application set you want to configure.
3. In the Program Neighborhood toolbar, click **Properties** to display the **Properties** dialog box.
4. Click the **Login Information** tab to display the **Login Information** page.
5. Enter a valid username, domain, and password for this application set.
6. To save this password after you exit Program Neighborhood and close all connections, select **Save password**. If you leave this box unchecked, this password is retained only as long as Program Neighborhood and all current connections are open.
7. Click **OK**.

Configuring General Settings for Application Sets

To configure the general settings for application sets

1. Start Program Neighborhood.
2. Select the application set you want to configure.
3. From the **Tools** menu, click **Settings** to display the **Settings** dialog box.
4. Click the **General** tab to display the **General** page. From the **General** page, you can configure the following:
   - **Client Name**: This field allows you to change the name of the client computer. The Citrix server uses the client name to uniquely identify resources (such as mapped printers and disk drives) associated with a given client PC. The client name must be unique for each computer running the Citrix ICA Client.
   - **Serial Number**: This is the serial number of the ICA Client software. This field is necessary only when you are using the Citrix ICA Client with a product such as *WINFRAME* Host/Terminal, which requires each client to have a Citrix PC Client Pack serial number in order to connect to the server. If a serial number is required, you must enter it exactly as it appears on the Serial Number card. The **Serial Number** field is not used by MetaFrame servers.
   - **Keyboard Layout**: Allows you to specify the keyboard layout of your client computer. The Citrix server uses the keyboard layout information to configure your user session for your keyboard layout. The default value, which is **(User Profile)**, uses the keyboard layout specified in your user profile.

- **Keyboard Type**: Allows you to specify the keyboard type of your client computer. The Citrix server uses the keyboard type information to configure your user session for your keyboard type. Use the default value of **Default** for most English and European keyboards. When used with a Japanese keyboard, **Default** auto-detects the keyboard type.

Configuring Hotkeys for Application Sets

▶ **To configure hotkeys for application sets**

1. Start Program Neighborhood.
2. Select the application set you want to configure.
3. From the **Tools** menu, click **Settings** to display the **Settings** dialog box.
4. Click the **Hotkeys** tab to display the **Hotkeys** page.
5. For each hotkey in the list, select a shift state and a key.
6. You can disable the hotkey by selecting **(none)** for the key. The fields on the **Hotkeys** page are:
   - **Close Remote Application**: The Close Remote Application hotkey disconnects the published application from the Citrix server and closes the Citrix ICA Client window. The behavior of this hotkey is the same as choosing Close from the system menu of the ICA Client window.

     Closing the published application in this manner either leaves the associated application in a disconnected state on the Citrix server, or exits the application on the Citrix server, depending on how the server is configured.
   - **CTRL-ALT-DEL**: This hotkey causes the CTRL-ALT-DEL key sequence to be sent to the server that is running the published application. In Windows NT, the CTRL-ALT-DEL key sequence causes a Windows NT session to switch to the Windows NT Security desktop.
   - **CTRL-ESC**: This hotkey causes the CTRL-ESC key sequence to be sent to the server that is running the published application. CTRL-ESC is a standard Windows hotkey. Refer to your Windows documentation for more information on the CTRL-ESC hotkey.
   - **ALT-ESC**: This hotkey causes the ALT-ESC key sequence to be sent to the server that is running the published application. ALT-ESC is a standard Windows hotkey. Refer to your Windows documentation for more information on the ALT-ESC hotkey.
   - **ALT-TAB**: This hotkey causes the ALT-TAB key sequence to be sent to the server that is running the published application. ALT-TAB is a standard Windows hotkey. Refer to your Windows documentation for more information on the ALT-TAB hotkey.

- **ALT-BACKTAB**: This hotkey causes the ALT-SHIFT-TAB key sequence to be sent to the server that is running the published application. ALT-SHIFT-TAB is a standard Windows hotkey. Refer to your Windows documentation for more information on the ALT-SHIFT-TAB hotkey.
- **CTRL-SHIFT-ESC**: This hotkey causes the CTRL-SHIFT-ESC key sequence to be sent to the server that is running the published application. CTRL-SHIFT-ESC is a standard Windows NT 4.0 hotkey. Refer to your Windows NT 4.0 documentation for more information on the CTRL-SHIFT-ESC hotkey.

Configuring Event Logging for Application Sets

The **Event Logging** page allows you to instruct the Citrix ICA Client whether or not to keep a log of various events that occur while running published applications.

To configure event logging for application sets

1. Start Program Neighborhood.
2. Select the application set you want to configure.
3. From the **Tools** menu, click **Settings** to display the **Settings** dialog box.
4. Click the **Event Logging** tab to display the **Event Logging** page. From the **Event Logging** page, you can configure the following settings:
   - **Event Log File**: In the **Name** field, enter the name of the file in which to log Citrix ICA Client events.
     - Select the **Overwrite existing event log** button to cause the event log file to be overwritten with new events when a published application is run.
     - Select the **Append to existing event log** button to keep old events and add new ones to the end of the file.
   - **Log Events**: Use these buttons to select the event categories that you want to log. If no events are selected, no logging takes place. Five event categories can be selected for logging:
     - **Connections and Disconnections**: Logs an event whenever the Citrix ICA Client connects and disconnects from a Citrix server. This category is selected by default.
     - **Errors**: Logs an event whenever an error is encountered by the Citrix ICA Client. This category is selected by default.
     - **Data Transmitted**: Logs an event for each packet of information sent by the Citrix ICA Client to the Citrix server. This is intended primarily for technical support purposes.

- **Data Received**: Logs an event for each packet of information received by the Citrix ICA Client from the Citrix server. This category is intended primarily for technical support purposes.
- **Keyboard and Mouse Data**: Logs an event whenever you press a key on the keyboard or move the mouse. This category is intended for technical support purposes.

Starting the ICA Java Client in a Specific National Language

The ICA Java Client allows users to specify what locale to use to display the ICA Java Client's user interface (including various messages and dialog boxes). By default, if you do not specify a locale, your session uses the locale of your client machine to display the ICA Java Client.

The following locales are supported by the Citrix ICA Java Client:

- en (English)
- de (German)
- fr (French)
- es (Spanish)
- ja (Japanese)

For example, to use Japanese as your locale on a non-Japanese machine when connecting to the Citrix server named CitrixServer in applet mode, create an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=1024 height=768>
<param name=address value=CitrixServer>
<param name=Language value=ja>
<param name=Country value=ja>
</applet>
</body>
</html>
```

Both Language and Country parameters must be specified and must be set to the same value.

When forcing a locale in application mode, you must define a system property to the JVM using the command line argument appropriate for your JVM. You must add this command line argument to the command line in your launching script.

For example, to force Japanese as your locale when using Microsoft's JVM, add **/d:user.language=ja** to the command line in your launching script.

To create the same connection using Sun's JVM, add **-Duser.language=ja** to the command line in your launching script.

To create this connection from a Macintosh client machine, you must use JBindery to add **user.language** as a system property with the appropriate language (en, de, fr, es, or ja) as its value to the launching script for Macintosh.

OS/2 systems require that you specify the command **set LANG=*x*** and press ENTER before using jicasession.cmd to start the ICA Java Client, where *x* is en, de, fr, es, or ja.

Important In the application mode examples, note that you precede values with an equals sign (=) instead of a colon. These parameters are virtual machine parameters, not ICA Java Client parameters (which *are* preceded by a colon).

COM Port Mapping

The ICA Java Client supports COM port mapping on the following platforms:

- Windows 95, Windows 98, Windows NT 4.0
- Solaris Sparc, Solaris x86
- JavaOs for Business

Note To enable COM port mapping you must have the javax.comm standard extension installed on the client. For Windows 95/98/NT and Solaris platforms, you can download the Java Communications API from www.java.sun.com/products/javacomm. The JavaOS for Business platform includes javax.comm support.

Client COM port mapping allows devices attached to the client computer's COM ports to be used during an ICA session on the Citrix server. These mappings can be used by your applications just like any other network mappings.

Mapping a Client COM Port

▶ **To map a client COM port**

1. Start the ICA Java Client and log on to the Citrix server.
2. Start a DOS command prompt: on *WINFRAME*, double-click **Command Prompt** in the Main program group. On MetaFrame, click **Start** and then click **Programs**, then click **Command Prompt**.

3. At the prompt, type **net use com*x*: \\client\com*z***: where *x* is the number of the COM port on the server and *z* is the number of the COM port you want to map. Press ENTER.
4. To confirm the operation, type **net use** at the prompt. The list that appears contains mapped drives, LPT ports, and mapped COM ports.

Note COM port mapping is not Microsoft Windows TAPI-compatible. TAPI devices cannot be mapped to client COM ports.

Mapping Client Audio

Client audio mapping enables applications running on the Citrix server to play sounds through a sound device installed on the client computer. You can control the amount of bandwidth used by client audio mapping.

▶ **To configure ICA Client audio on a MetaFrame server**

1. Click **ICA Settings** in Terminal Server Connection Configuration.
2. Select an option from the **Client Audio Quality** drop-down list.

▶ **To configure ICA Client audio on a *WINFRAME* server**

1. Click **ICA Settings** in Citrix Connection Configuration.
2. Select an option from the **Client Audio Quality** drop-down list.

Client Audio Mapping can cause excessive load on the Citrix servers and the network. The higher the audio quality, the more bandwidth is required to transfer the audio data. Higher quality audio also uses more server CPU to process. Three different audio quality settings are available, or client audio mapping can be disabled completely.

Important You can set audio quality on a per connection basis, but users can also set it on the client computer. If the client and server audio quality settings are different, the lower of the two qualities is used.

The **Client Audio Quality** options are:

- **High**. This setting is recommended only for connections where bandwidth is plentiful and sound quality is important. This setting allows clients to play a sound file at its native data rate. Sounds at the highest quality level require about 1.3Mbps of bandwidth to play clearly. Transmitting this amount of data can result in increased CPU utilization and network congestion.
- **Medium**. This setting is recommended for most LAN-based connections. This setting causes any sounds sent to the client to be compressed to a maximum of 64Kbps. This compression results in a moderate decrease in the quality of the

sound played on the client computer. The host CPU utilization will decrease compared with the uncompressed version due to the reduction in the amount of data being sent across the wire.

- **Low**. This setting is recommended for low-bandwidth connections, including most modem connections. This setting causes any sounds sent to the client to be compressed to a maximum of 16Kbps. This compression results in a significant decrease in the quality of the sound. The CPU requirements and benefits of this setting are similar to those of the Moderate setting; however, the lower data rate allows reasonable performance for a low-bandwidth connection.

Note The ICA Java Client supports only Medium audio quality. You must set the server to either High or Medium when configuring audio mapping for use with the ICA Java Client.

Connecting to a Server Across a Firewall

Network firewalls can allow or block packets based on the destination address and port. If you are using the ICA Java Client through a network firewall that employs IP address translation, you must specify the following parameters:

- **UseAlternateAddress**: Use an alternate address across a firewall (specified by the parameter TcpBrowserAddress). This parameter is always preceded by user.wfclient. The values are **0** (default; actual address is used) and **1** (alternate address is used). If this parameter is set to **1**, the parameter TcpBrowserAddress must also be specified. Setting this parameter to **0** is the same as not using the parameter.
- **TcpBrowserAddress**: The external Internet address of a Citrix server that is configured to map the internal network address of Citrix servers to external Internet addresses.

Note All Citrix servers in the farm must be configured with their alternate (external) address. See the *MetaFrame Administrator's Guide* for more information.

For example, to connect to a server across a firewall in applet mode and use an alternate address for the server Fountain, make an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=address value=Fountain>
<param name=TcpBrowserAddress value=177.17.1.7>
<param name=user.wfclient.UseAlternateAddress value=1>
</applet>
</body>
</html>
```

To make the same connection as above in application mode, type:

```
jicasession -address:Fountain -TcpBrowserAddress:177.17.1.7
-user.wfclient.UseAlternateAddress:1
```

Auto-reconnect and Session Termination

The ICA Java Client supports new parameters that allow you to control the client's behavior when starting or ending a session. The following parameters can be specified.

- **Start**: This parameter controls an ICA Java Client's behavior when you start a session. The values are **manual** and **auto**. If you set this parameter to **manual** (default), you must click **Click to connect** to connect to a Citrix server. If set to **auto**, the message “Connecting to server” is displayed and you are automatically connected to the server.
- **End**: This parameter controls the ICA Java Client’s behavior when you terminate a session. The values are **manual** (default), **auto**, **terminate**, and *url*. If you set this parameter to **manual**, you can reconnect your session by clicking **Click to connect**. When set to **auto**, the message “Connection Terminated - Reconnecting in 8 seconds” is displayed. The number of seconds counts down to 0 and the server is reconnected. When set to **terminate**, either the message “Connection Terminated” or “Connection Error” is displayed and you cannot reconnect to the server. When set to *url*, where *url* is the address of any Web page, either the message “Connection Terminated - Moving to *url* in 8 seconds” or “Connection Error - Moving to *url* in 8 seconds” is displayed. The number of seconds counts down to 0 and you are connected to the indicated Web page.

For example, to use the Start parameter to automatically connect to a session and the End parameter to prevent you from reconnecting to the server, make an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=Start value=auto>
<param name=End value=terminate>
</applet>
</body>
</html>
```

Printing

The ICA Java Client can be configured to map client printers to an ICA session so that client system users can print to their local or network printers. Printers can be created with ICA Client Printer Configuration, auto-created, or manually created.

ICA Client Printer Configuration

After starting your ICA session, click **Start** and point to **Programs**. Next point to **MetaFrame Tools (Common)** and click **ICA Client Printer Configuration**. From the **Printer** menu, click **New**. The **Add ICA Client Printer** Wizard appears. In the Wizard, first select your printer from the list of printer drivers and click **Next**. Now select the port on your client system to which your printer is connected and click **Next**. Finally, type a name for this printer and click **Next**. The next screen shows a summary of your printer configuration. If any of the entries are incorrect, click **Back** to make corrections. When finished, click **Finish**.

Notes If the Add ICA Client Printer Wizard does not display a list of printers, a printer must be added to the server. Check with your system administrator.

An error message may appear indicating that the ICA Client printer could not be created. Ignore this message. Your printer was correctly created.

Tip If you start your ICA session from the command line, you can use the -inidir parameter to create a file so your printer configuration is saved between sessions. For example, to start your ICA session using the c:\config directory to store the configuration file, type the following (along with any other required parameters): **jicasession -inidir:c:\config**.

When you configure a printer, the printer.ini file is created in the specified directory. Use the same command line to start future ICA sessions so the configuration file is found.

Auto-Created Printers

Auto-created printers are configured in the HTML page or from the command line. When users log on to a Citrix server, their printers appear in Print Manager or in the Printers folder and are ready for use. When users log off, their local printers are deleted from the Citrix server.

To auto-create printers, the ICA Java Client must be passed three values:

- **Printer name**. Can be any name by which you would like to identify the printer.
- **Port name**. Specify a file name, port name, or printer IP address (or network name) and print queue.

  When printing to a file, the output file is composed of printer machine code. This file can be sent to a printer using a platform-specific utility. For example, use a command prompt on Windows platforms to send the file to a printer by copying the file to a printer port.

  When printing to a port, specify the port (for example, LPT1:).

  When printing to a network printer, specify the printer's IP address or network name and print queue (for example, ipaddress:printqueuename).

  The ICA Java Client also allows users to print to standard out or standard error. Specify **stdout** or **stderr** to print to standard out or standard error, respectively.
- **Driver**. Specify the printer driver. This name must exactly match the driver name on the Citrix server. Check the driver list on the server. For MetaFrame servers, double-click My Computer, double-click Printers, then double-click Add Printer. In the Add Printer Wizard, be sure My Computer is selected. Click Next. Select a port, such as LPT1:. Click Next. The list under Printers contains the printer drivers. For *WINFRAME* servers, double-click Print Manager in the Main Program group. On the Printer menu, click Create Printer. Use the arrow across from the Driver box to see a list of printer drivers.

Note To auto-create a printer, the Citrix server must have the printer driver installed. If the driver is not installed, the printer is not auto-created. In this case, the user must manually add the printer during the ICA session.

The parameter name to use when passing the printer name to the ICA Java Client is user.localclientprinters. When passing the port name, use the parameter user.*printername*.port, where *printername* is the name you specify in user.localclientprinters. When passing the driver, use the parameter user.*printername*.driver, where *printername* is also the name specified in user.localclientprinters.

The following example demonstrates auto-creating a printer by passing these values to the ICA Java Client in applet mode on a Windows client system. In this example, the printer's name is Laser. It is connected to the client's LPT1 port and has a driver named HP LaserJet 4/4M Plus PS.

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=CitrixServer>
<param name=user.localclientprinters value=Laser>
<param name=user.Laser.port value=lpt1:>
<param name=user.Laser.driver value="HP LaserJet
4/4M Plus PS">
</applet>
</body>
</html>
```

In this example, the printer's name is also Laser and has a driver named HP LaserJet 4/4M Plus PS. The printer is a network printer that exists on a network print server with an IP address of 127.0.0.0 and a print queue named LPT_PASSTHRU.

```
<html>
<body>
<applet code=com.citrix.JICA.class
archive=JICAEngN.jar
width=640 height=480>
<param name=address value=CitrixServer>
<param name=user.localclientprinters value=Laser>
<param name=user.Laser.port value=127.0.0.0:LPT_PASSTHRU>
<param name=user.Laser.driver value="HP LaserJet
4/4M Plus PS">
</applet>
</body>
</html>
```

To create the printer described in the first applet mode example above, type the following on the command line (all on one line):

```
jicasession -address:CitrixServer -user.localclientprinters:Laser
-user.Laser.port:lpt1: "-user.Laser.driver:HP LaserJet 4/4M Plus PS"
```

To create the network printer described in the second applet mode example above, type the following on the command line (all on one line):

```
jicasession -address:CitrixServer -user.localclientprinters:Laser
-user.Laser.port:127.0.0.0:LPT_PASSTHRU "-user.Laser.driver:HP LaserJet
4/4M Plus PS"
```

The following example illustrates configuring multiple printers, one of which is a network printer. The port name of a network printer may be a UNC name or the name of a local port that has a network printer mapped to it. In this example, the printer NetPrinter1 is a network printer mapped to the local LPT2 port and has a driver named HP Laser Jet 400 Series PS.

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name="address" value=CitrixServer>
<param name=user.localclientprinters
value=Laser,NetPrinter1>
<param name=user.Laser.port value=lpt1:>
<param name=user.Laser.driver value="HP LaserJet
4/4M Plus PS">
<param name=user.NetPrinter1.port value=lpt2:>
<param name=user.NetPrinter1.driver value="HP LaserJet
400 Series PS">
</applet>
</body>
</html>
```

In application mode, type the following on the command line (all on one line):

```
jicasession -address:CitrixServer
-user.localclientprinters:Laser,NetPrinter1 -user.Laser.port:lpt1:
"-user.Laser.driver:HP LaserJet 4/4M Plus PS"
-user.NetPrinter1.port:lpt2: "-user.NetPrinter1.driver:HP LaserJet 400
Series PS"
```

Manually Created Printers

Manually created printers are added by users, using Print Manager, while connected to a Citrix server with the ICA Java Client. The procedures for manually adding a printer when connected to a *WINFRAME* or a MetaFrame server differ.

Notes You must specify the printer parameters as explained in Auto-Created Printers, above, in order to manually create printers.

It is only necessary to manually create a printer when either the printer driver is not installed on the server, or the server is configured to not allow auto-created printers.

Manually created printers are not deleted from Print Manager at log off.

▶ **To manually add a printer during a *WINFRAME* server session**

1. Make an applet or application mode connection to a *WINFRAME* server.

2. In the remote session window, double-click **Main** and then double-click **Print Manager**.
3. From the **Printer** menu, click **Create Printer**. The **Create Printer** dialog box appears.
4. Enter the name of your printer in the **Printer Name** field. The name should be in the format clientname#LPTx, where clientname is the name of your ICA client computer and x is the client computer LPT port to which the printer is attached. In the **Driver** field, select the printer's driver. In the **Print to** field, select the client computer's LPT port; for example, CLIENT\LPT1:. Click **OK**
5. Depending upon the type of printer, a series of dialog boxes will appear in which you can configure the printer's settings. After you enter the information, the printer appears as an entry in Print Manager.

▶ **To manually add a printer during a MetaFrame server session**

1. Make an applet or application mode connection to a MetaFrame server.
2. In the remote session window, double-click **My Computer** and then double-click **Printers**.
3. Double-click **Add Printer**. Select **Network printer server** and click **Next**.
4. In the **Shared Printers** field, double-click **Client Network** and then double-click **Client**.
5. Double-click the LPT port to which the local printer is attached.
6. If the server does not have a suitable printer driver installed, you are prompted to install the driver on the client computer. Click **OK**.
7. In the **Manufacturer** field, select your printer's manufacturer. In the **Printers** field, select the model of your printer. Click **OK**.
8. If prompted for the location of the printer driver, click **OK** and enter the location of the files in the **Copy files from** field. Click **OK**.
9. Click **Finish**.

Event Logging

Event Logging creates a log file of various events that occur while running the ICA Java Client. The following parameters can be specified:

Parameter	Description
LogFile	The name of the log file to be kept. Can be any name. Log data can alternately be sent to standard out or standard error by specifying **stdout** or **stderr** instead of a file name.
LogAppend	The event log can be overwritten by new events or new events can be added to the end of the existing file. Specify **on** to save old events and add new events to the end of the file. Specify **off** to overwrite old events.
LogConnect	Logs an event when the ICA Java Client connects to or disconnects from the Citrix server. Values are **on** or **off**.
LogErrors	Logs ICA Java Client errors. Values are **on** or **off**.
LogTransmit	Logs data sent from the ICA Java Client to the Citrix server. Values are **on** or **off**.
LogReceive	Logs data sent from the Citrix server to the ICA Java Client. Values are **on** or **off**.
LogKeyboard	Logs keyboard data. Values are **on** or **off**.

All parameters except Logfile are **off** by default.

When specifying parameters for event logging, the parameter names are preceded by the words user.wfclient. For example, the parameter LogConnect becomes user.wfclient.Logconnect.

The following examples demonstrate how to create an event log named Events.log that contains error and connection data. This log is not overwritten by new events.

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=address value=CitrixServer>
<param name=user.wfclient.logfile value=Events.log>
<param name=user.wfclient.logerrors value=on>
<param name=user.wfclient.logconnect value=on>
<param name=user.wfclient.logappend value=on>
</applet>
</body>
</html>
```

From the command line, type:

```
jicasession -address:CitrixServer -user.wfclient.logfile:Events.log
-user.wfclient.logerrors:on -user.wfclient.logconnect:on
-user.wfclient.logappend:on
```

Business Recovery

This feature provides consistent connections to published applications in the event of a primary server disruption. You can define up to three groups of Citrix servers to which you want to connect: a primary and two backups. Each group can contain from one to five servers. When you specify a server group for your client, the client attempts to contact all the servers within that group simultaneously (broadcasting); the first server to respond is the one to which you connect. You can specify up to three groups containing up to five servers each. Use the following parameters to specify server groups:

- **TcpBrowserAddress to TcpBrowserAddress5**: Specify the primary group of servers.
- **TcpBrowserAddress6 to TcpBrowserAddress10**: Specify the first backup group of servers.
- **TcpBrowserAddress11 to TcpBrowserAddress15**: Specify the second backup group of servers.

Any unused addresses are filled in with five dashes (-----). These dashes are required to fill in any gaps in the list but are not required at the end of the list.

In the following examples, the primary group of servers contains "Groundhog," "Woodchuck," and "Hedgehog." The first backup group contains the servers "Steel" and "Bronze." There is no secondary backup group.

For example, to specify primary and backup servers applet mode, make an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=TcpBrowserAddress value=Groundhog>
<param name=TcpBrowserAddress2 value=Woodchuck>
<param name=TcpBrowserAddress3 value=Hedgehog>
<param name=TcpBrowserAddress4 value=----->
<param name=TcpBrowserAddress6 value=Steel>
<param name=TcpBrowserAddress7 value=Bronze>
</applet>
</body>
</html>
```

To specify the same primary and backup servers as above, but using the command line, type:

```
jicasession -TcpBrowserAddress:Groundhog
-TcpBrowserAddress2:Woodchuck -TcpBrowserAddress3:Hedgehog
-TcpBrowserAddress4:----- -TcpBrowserAddress5:-----
-TcpBrowserAddress6:Steel -TcpBrowserAddress7:Bronze
```

Keyboards

The ICA Java Client lets users specify what type of keyboard to use in their ICA sessions. By default, if you do not specify a keyboard preference, your session uses the keyboard that is specified as the default keyboard for the Citrix server.

Note Different JVMs interpret key detection differently.

The ICA Java Client supports using any keyboard supported by the Citrix server to which it is connecting.

To specify a keyboard other than the server's default for use in an ICA session, specify the parameter system.wfclient.keyboardlayout with a value from the server's list of supported keyboards. Values that contain a space (for example, Swiss German) should be enclosed in quotation marks.

The following locations contain a list of supported keyboards:

- On MetaFrame servers: open Control Panel, double-click **Keyboard**, and then click the **Input Locales** tab. The **Input Locales** list in this tab includes all installed locales and layouts. You must use the value specified in the **Layout** column for your keyboard when specifying a keyboard other than the server's default.
- On *WINFRAME* servers: open Control Panel, double-click **International**. The **Keyboard Layout** field includes a list of all keyboards you can specify. You must use the value specified in this list for your keyboard when specifying a keyboard other than the server's default.

For example, to specify a Danish keyboard in applet mode, create an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=address value=CitrixServer>
<param name=system.wfclient.keyboardlayout value=Danish>
</applet>
</body>
</html>
```

From the command line, type:

```
jicasession -address:CitrixServer -system.wfclient.keyboardlayout:Danish
```

The ICA Java Client supports the following keyboard types to distinguish between subtypes of the Japanese keyboard layout:

- 101 Keyboard (Japanese)
- 106 Keyboard (Japanese)

To specify a Japanese keyboard type, specify the system.wfclient.keyboardtype parameter.

For example, to specify a Japanese type 101 keyboard in applet mode, create an HTML page like the following:

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=address value=CitrixServer>
<param name=system.wfclient.keyboardlayout value="Japanese MS-IME98">
<param name=system.wfclient.keyboardtype value="101 Keyboard
(Japanese)">
</applet>
</body>
</html>
```

From the command line, type:

```
jicasession -address:CitrixServer
"-system.wfclient.keyboardlayout:Japanese MS-IME98"
"-system.wfclient.keyboardtype:101 Keyboard (Japanese)"
```

Notes As described in "Specifying .Ini Files on the Command Line" later in this chapter, application mode users can use the .ini files of other Citrix ICA Clients by referencing the .ini files in command line parameters. Keyboard layout information is saved in wfclient.ini; users can specify this file on the command line to make use of its keyboard configuration information.

Hotkeys

The Citrix ICA Java Client provides hotkeys that can be used to control various Java Client functions. When the ICA session has the focus on the client system's desktop, pressing these hotkeys causes the associated action to occur.

The following hotkeys are available to client system users:

Parameter	Description
Hotkey2	Close Remote Application. The Close Remote Application hotkey closes programs opened in an ICA session. If no programs are open, this hotkey prompts the user that the ICA session will be shut down and then, if given permission, disconnects the ICA session. Hotkey2 is mapped by default to SHIFT+F3.
Hotkey4	Substitute for the standard Windows hotkey CTRL+ALT+DEL. The CTRL+ALT+DEL hotkey displays the Windows NT Security desktop in the ICA session. Hotkey4 is mapped by default to CTRL+F1.
Hotkey5	Substitute for the standard Windows hotkey CTRL+ESC. On *WINFRAME* servers, this hotkey causes the remote Task List to appear. On MetaFrame servers, the remote Windows NT Start menu appears. Hotkey5 is mapped by default to CTRL+F2.
Hotkey6	Substitute for the standard Windows hotkey ALT+ESC. This hotkey is used to bring the focus to maximized and minimized windows of programs that have been opened in an ICA session. Hotkey6 is mapped by default to ALT+F2.
Hotkey7	Substitute for the standard Windows hotkey ALT+TAB. Use this hotkey to cycle through applications that are open in the ICA session. A popup box appears and displays the programs as you cycle through them. The chosen application receives keyboard and mouse focus. Hotkey7 is mapped by default to ALT+PLUS.
Hotkey8	Substitute for the standard Windows hotkey ALT+SHIFT+TAB. Like the ALT+TAB hotkey, this key sequence cycles through applications that are open in the ICA session but in the opposite direction. The chosen application receives keyboard and mouse focus. Hotkey8 is mapped by default to ALT+MINUS.

ICA Java Client hotkeys use a two key sequence. The first key defines the shift state and the second defines the character. To set the ICA Java Client hotkeys to key combinations other than their defaults, select a shift state value and a character value from the following lists:

- Shift state: SHIFT, CTRL, ALT, any combination of these keys, or none
- Character: ESCAPE, MINUS, PLUS, STAR, TAB, any of the F keys (F1 through F12), or none

These values are passed to the ICA Client as parameters in the HTML page or from the command line. For each hotkey, two parameters are specified: one for the shift state and a second for the character state.

When specifying parameters for hotkeys, the hotkey name (Hotkey2, for example) is preceded by the words user.wfclient. and followed by shift or char, depending upon whether the parameter describes the character or shift state key of the key sequence.

The following example describes how to create Close Remote Application and ALT+TAB hotkeys for use in an applet mode ICA session. The Close Remote Application hotkey will have the key sequence CTRL+F1 and the ALT+TAB hotkey key sequence will be SHIFT+TAB.

```
<html>
<body>
<applet code=com.citrix.JICA.class
width=640 height=480>
<param name=address value=CitrixServer>
<param name=user.wfclient.hotkey2shift value=ctrl>
<param name=user.wfclient.hotkey2char value=f1>
<param name=user.wfclient.hotkey7shift value=shift>
<param name=user.wfclient.hotkey7char value=tab>
</applet>
</body>
</html>
```

In application mode, type the following on the command line:

```
jicasession -address:CitrixServer -user.wfclient.hotkey2shift:ctrl
-user.wfclient.hotkey2char:f1 -user.wfclient.hotkey7shift:shift
-user.wfclient.hotkey7char:tab
```

To disable a hotkey, enter **none** as a value for one of the parameters.

Euro Currency Symbol Support

The Citrix ICA Java Client (Version 4.02 and higher) supports creation of the Euro currency symbol provided the following requirements are met:

- JDK 1.1.7B or later Java environment
- Keyboard and operating system that have a Euro symbol key

Additionally, with Version 4.10 or later of the Java client, it is possible to cut a Euro symbol from a native desktop document and paste it into the client window.

ICA Java Client Advanced Syntax

Note This section provides an explanation of the command line and HTML syntax used to configure the ICA Java Client. This explanation gives you an understanding of the logic behind the examples presented in this chapter (such as client printing, event logging, and hotkeys) and allows users to apply this logic to further customization of the ICA Java Client and ICA sessions.

To initiate an ICA session, the ICA Java Client engine must build an *information base*. This information base contains the settings that govern the client's general operation and settings that apply to the current connection (such as address, username, and password). In the case of other Citrix ICA Clients, such as the ICA Win32 Client or the ICA DOS Client, this information base is maintained by either Remote Application Manager or Program Neighborhood. Users can customize the behavior of their ICA Client and individual ICA sessions by using Remote Application Manager or Program Neighborhood to set various options and preferences. These settings are stored in three .ini files:

- appsrv.ini: A read/write file that contains settings pertaining to the user.
- module.ini: A read/write file that contains settings pertaining to the workstation. It is shared by all users on a particular workstation.
- wfclient.ini: A read-only file that contains settings pertaining to a particular network. Established by a system administrator.

When the client system user initiates a connection, the ICA Client engine references the .ini files and configures the connection according to the settings in the .ini files. If a setting is not present in an .ini file, the system default is used.

An example of an appsrv.ini file is excerpted below:

```
;**********************************************************************
;**
;** APPSRV.INI - Application Server Configuration File.
;**
;** This file contains user settings which define application
;** servers and other preferences.
;**
;**********************************************************************
[WFClient]
Version=2
LogFile=error.log
LogAppend=Off
LogConnect=On
LogErrors=On
LogTransmit=Off
LogReceive=Off
```

```
LogKeyboard=Off
Hotkey2Char=F3
Hotkey2Shift=Shift
Hotkey3Char=F2
Hotkey3Shift=Shift
Hotkey4Char=F1
Hotkey4Shift=Ctrl
Hotkey5Char=F2
Hotkey5Shift=Ctrl
Hotkey6Char=F2
Hotkey6Shift=Alt
Hotkey7Char=plus
Hotkey7Shift=Alt
Hotkey8Char=minus
Hotkey8Shift=Alt
DisableSound=Off
MouseTimer=100
KeyboardTimer=50
TcpBrowserAddress=
ServerToClientBuffPowerOf2=15
ClientToServerBuffPowerOf2=13
DesiredColor=2
ScreenPercent=75
DesiredHRES=640
DesiredVRES=480
KeyboardLayout=(User Profile)
KeyboardType=(DEFAULT)
ApplicationSetManagerIconOff=Off
CustomConnectionsIconOff=Off
FindNewApplicationSetIconOff=Off
AddICAIconOff=Off
PNDefault=<none>

[Common Default Information]
ProgramGroup=Citrix ICA Client
WinStationDriver=ICA 3.0
Password=000100

[Default Network Connection]
Address=LLAMA4M
TransportDriver=TCP/IP
Published Application=No

[LocalClientPrinters]
MyPrinter=

[MyPrinter]
Port=LPT1
Driver=HP LaserJet 4/4M Plus PS
```

```
[ApplicationServers]
llama4m=

[llama4m]
TransportDriver=TCP/IP
Address=LLAMA4M
Compress=Off
PersistentCacheEnabled=Off
ClientAudio=Off
MouseTimer=0
KeyboardTimer=0
EncryptionLevelSession=Basic
AudioBandwidthLimit=-1
WinStationDriver=ICA 3.0
Username=jerry
Domain=accounting
Password=000100
;*********************************************************************
```

The settings other ICA Clients store in appsrv.ini can be reproduced on the command line or in the HTML page for the ICA Java Client by using the proper syntax. The ICA Java Client engine accepts parameters taken from the appsrv.ini section of the information base if they are presented with the following syntax:

-metasection.section.key:val

In HTML, the syntax is:

<param name=*metasection.section.key* value=*val*>

Use the following four parameters for both cases:

- *Metasection* is the name of one of the sections of the information base. In the case of parameters taken from AppSrv.ini, the metasection name is “user.”
- *Section* is the name of a section in an Ini file. Sections are denoted by [] symbols in the Ini file. For example, the excerpted AppSrv.ini file above contains four sections: [WFClient], [Common Default Information], [Default Network Connection], and [ApplicationServers].

  Notes Other sections may be present in appsrv.ini. These sections contain connection-specific information, such as [Llama4m] and [MyPrinter] in the sample appsrv.ini file above.

  The [Common Default Information] section is not used by the ICA Java Client.

- *Key* is the basic unit of sections. For example, the [WFClient] section of the AppSrv.ini file above contains the keys LogFile, LogFileWin16, LogFileWin32, etc.

- *Val* is a variable to which the key is set. For example, the key LogFile is set to the value error.log.

The file above contains the section [WFClient], which contains the key LogFile. LogFile is set to the value error.log.

To recreate this parameter on the command line, specify:

```
-user.wfclient.logfile:error.log
```

In HTML, specify:

```
<param name=user.wfclient.logfile value=error.log>
```

Parameters set in module.ini and wfclient.ini can be set in the same manner. The metasection name for parameters based on module.ini settings is "system." Parameters based on wfclient.ini settings have the metasection name "global."

The information base also contains parameters that describe the current connection the user is about to initiate. In the example appsrv.ini file above, the section [llama4m] contains current connection parameters. Llama4m is a hypothetical Citrix server name. The section [llama4m] contains parameters that describe such settings as username, domain, password, address, and initial program. Because the ICA Java Client uses the information base to immediately make a connection, the metasection and section parameters are not specified.

The command-line syntax for current connection settings is:

```
-key:value
```

In HTML, specify:

```
<param name=key value=val>
```

as in `-address:CitrixServer` or `<param name=address value=CitrixServer>`

Important Use only parameters for features of the ICA Java Client. For example, any parameters in the .ini files that pertain to modems or protocols other than TCP/IP cannot be used with the ICA Java Client.

Specifying .Ini Files on the Command Line

The ICA Java Client allows you to specify .ini files of other Citrix ICA Clients. Specifying the .ini files of another ICA Client causes the ICA Java Client to use the settings within those files. For example, if you have an appsrv.ini file from another ICA Client that contains useful default settings, you might find it convenient to specify this file when using the ICA Java Client.

To specify an .ini file, you must pass the name (including the path, if necessary) of the file to the ICA Java Client on the command line.

.Ini file locations are passed using the following parameters:

Parameter	Description
Iniappsrv	Specifies the location and name of an appsrv.ini file.
Inimodule	Specifies the location and name of a module.ini file.
Iniwfclient	Specifies the location and name of a wfclient.ini file.
IniDir	Specifies the location of a group of .ini files.

For example, to use an appsrv.ini file named appsrv.ini, specify:

```
jicasession -iniappsrv:C:\Citrix\appsrv.ini
```

You can also specify a connection entry in an appsrv.ini file. For example, suppose you have an appsrv.ini file that, like the excerpted appsrv.ini file shown earlier in this chapter, contains a section for a connection called [llama4m]. This section contains configuration information such as the address of the server, user name for logon, compression values, etc. To reference this information, you can add this section name to the parameters you specify on the command line. Assume that, like the example above, the file is located in a directory called C:\Citrix and is named appsrv.ini. To specify this entry, type:

```
jicasession llama4m -iniappsrv:C:\Citrix\appsrv.ini
```

Notes The address parameter is the only parameter that must be specified for a non-Program Neighborhood session; however, it can be anywhere in the search path. For example, it can be located in the appsrv.ini file and therefore, not be specified on the command line. This parameter is searched for in these locations: command line, ICA file, appsrv.ini.

In general, parameters on the command line take precedence over the same parameters in ICA or .ini files. Parameters in ICA files take precedence over the same parameters in .ini files. When a parameter is specified more than once in the same place (same ICA file, for example), the last occurrence of the parameter has precedence.

Use the IniDir parameter to specify the location of a set of .ini files (appsrv.ini, module.ini, wfclient.ini, or any combination of these files) to the ICA Java Client. .Ini file use is a three step process:

▶ To pass Ini files to the ICA Java Client

1. By default, the ICA Java Client checks if any (or any combination) of the following three Ini parameters (Iniappsrv, Inimodule, Iniwfclient) is specified on the command line.

2. If a parameter is set, the ICA Java Client uses the .ini file located in the specified URL. If the ICA Client does not find one or all of the parameters to be set, the ICA Java Client checks if the IniDir parameter is specified. If IniDir is specified, the ICA Client then searches that location for each of the .ini files not specified on the command line.
3. Finally, if IniDir is not specified on the command line, the ICA Java Client searches for .ini files in an Ini directory off the current working directory. For example, if you are running the ICA Java Client from a directory called C:\Java and have not specified IniDir on the command line, the ICA Client looks for .ini files in C:\Java\Ini\. If your working directory does not contain an Ini directory, you might find it useful to create an Ini directory and not specify .ini parameters on the command line when you run the ICA Client. Each time you run the ICA Client, it searches the Ini directory and uses any .ini files the directory contains.

Storing User Settings

User settings are stored in a variety of locations, based upon the operating system and Java Virtual Machine/browser environment. The rules for storing user settings (.ini files) are as follows:

- If any of the parameters Iniappsrv, Iniwfclient, or Inimodule are specified, the directory indicated is used for the respective file (appsrv.ini for Iniappsrv, wfclient.ini for Iniwfclient, and module.ini for Inimodule).
- If the IniDir parameter is specified and the directory exists, then this location is used for storing .ini files. If the IniDir parameter is specified without a directory, then no attempt is made to save user settings. If the IniDir parameter is specified with a directory that is non-existent or non-accessible, an attempt is made to create the directory.
- If the IniDir parameter is not specified and the ./Ini directory exists and is writable, then this directory is used for .ini files.
- If the IniDir parameter is not specified and the ./Ini directory does not exist (or is not writable) then an attempt will be made to use one of the following locations:
  - For Unix type systems (where the file separator is a slash (/)): *user.home*/.citrix/ where *user.home* is the path to the user's home directory.

- For all other systems: *user.home*<*separator*>Citrix<*separator*> where *user.home* is the path to the user's home directory and <*separator*> is the file separator used by the operating system.

  For example, on systems running Unix, the .ini files are stored in ~/.citrix/. On systems running Windows 95, Windows 98, and Windows NT 4.0, C:\Citrix\ may be used. If the home directory does not exist, or if the Citrix or .citrix directory cannot be created, then no attempt is made to save user settings.

Note For systems running Windows 95, Windows 98, and Windows NT 4.0, C:\Citrix\JavaClient is the default installation directory for the ICA Java Client, so it is likely that it will exist for users running the ICA Java Client in application mode.

On systems running Windows 95, Windows 98, and Windows NT 4.0 with JavaSoft's Java Development Kit (JDK), the **user.home** property defaults to %HOME%, which is not set by default. The Java Virtual Machine then attempts to use %HOMEDRIVE%%HOMEPATH%, which is set to the root directory of the drive containing the operating system (usually C:\).

For client systems used by more than one person, each user should set their HOME environment variable to a user-specific directory. For systems running Windows NT 4.0, we recommend setting HOME to the user's profile directory.

If printer.ini exists and contains a LocalClientPrinters section, then printers defined in printer.ini will be used and new printers defined by the client will be saved here. When printer.ini is used, command-line parameters must use "printer" as the metasection name. Printer definitions beginning with "user" are ignored. See "Printing" for more information.

If printer.ini does not exist or exists but does not contain printer definitions, then printer definitions are read from and saved to appsrv.ini. In this case, the metasection name must be "user." See "Printing" earlier in this chapter for more information.

For additional information see "ICA Java Client Advanced Syntax" and "ICA Java Client Parameters" earlier in this chapter.

Configuring a SOCKS Proxy Connection

You can configure the ICA Java Client to connect to a Citrix server through a SOCKS proxy server. This section describes:

- Why you use a SOCKS proxy server
- Where to locate your SOCKS proxy server
- How to configure the ICA Java Client to work with SOCKS proxy servers

Using SOCKS to Direct ICA Traffic through Firewalls

To limit access into and out of your Citrix servers, configure a SOCKS proxy server to handle connections between clients and the server. You can place the proxy server on either side of the firewall, or in some situations, on both sides of the firewall.

The benefits of using a SOCKS proxy server include:

- Information hiding, where system names inside the firewall are not made known to systems outside the firewall through DNS (Domain Name System)
- Authentication between an ICA Client and SOCKS proxy servers
- Authentication between two SOCKS proxy servers
- Relaying between two SOCKS proxy servers
- Channeling different TCP connections through one connection
- UDP proxying

Note The ICA Java Client supports only clear text username and password authentication.

The general procedure for connecting the ICA Java Client through a proxy is:

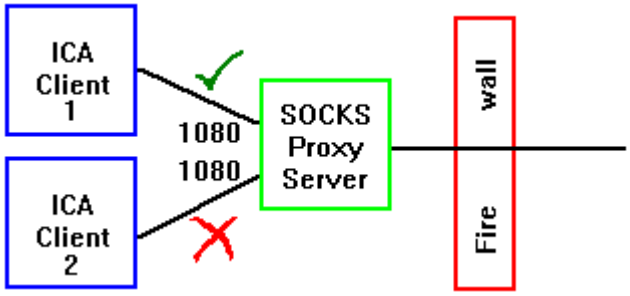
1. Be sure that your firewall is configured and working properly.
2. Install your SOCKS proxy server and test that it works with Web browsers.
3. Configure and deploy the ICA Java Client.

Locating Your Proxy Server

You can locate your proxy server on either side of your firewall. In some situations, you may want to locate a proxy server on both sides of the firewall. Typical SOCKS proxy configurations are described below. See your proxy documentation for further details on placement and implementation of your proxy server.

Setting Up a Proxy between Clients and a Firewall (for Outbound Connections)

To restrict clients from connecting directly to servers outside your firewall, install a proxy server between the client systems and the firewall, as shown below.

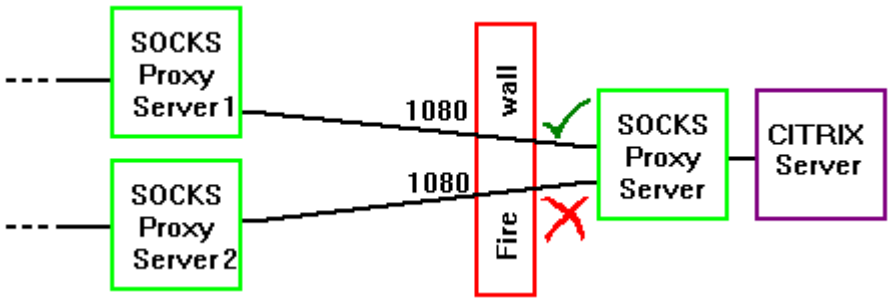


The proxy server uses its authentication features to determine whether ICA Clients can access networks outside the firewall. Configure the firewall to pass only network traffic that comes from the SOCKS proxy server.

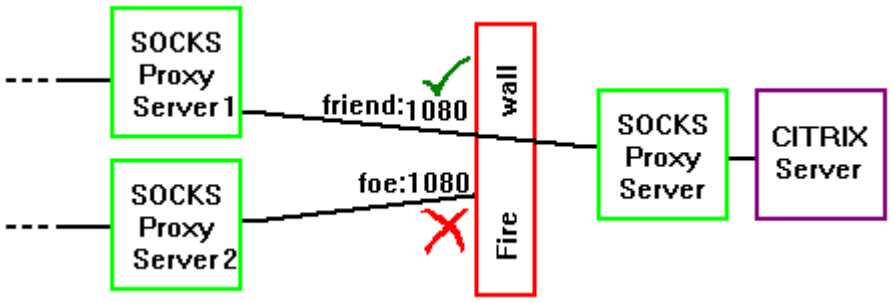
Setting Up a Proxy Between Citrix Servers and a Firewall (for Inbound Connections)

To protect your Citrix servers, install a proxy server between your servers and the firewall. You can configure the firewall in two ways:

Maximize Trust. Configure the firewall to pass only network traffic that is directed to the SOCKS proxy server. The proxy server performs the authentication of the ICA Client.

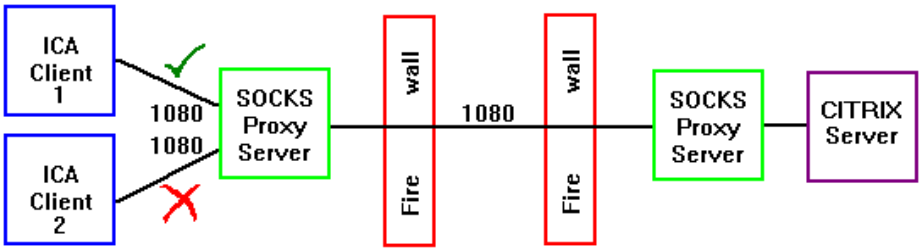


Minimize Risk. Configure the firewall to allow only connections from specific machines in addition to passing only network traffic that is directed to the SOCKS proxy server.



Setting Up a Virtual Private Network Using Two Proxy Servers

You can create a Virtual Private Network (VPN) between two sites by configuring a proxy server inside the firewall at both the client and server sites. Set up the firewalls to allow only directed UDP traffic between the two SOCKS proxy servers and TCP on the SOCKS port. For additional security, configure the SOCKS proxy server on the ICA Client side to authenticate with the SOCKS proxy server on the Citrix server side.



Configuring the ICA Java Client for Use with a SOCKS Proxy Server

You can configure the ICA Java Client to work with a SOCKS proxy server in both applet and application mode. For application mode, you can either edit the launching scripts or start Program Neighborhood and select **Properties** to configure the ICA Java Client. In all cases, you need to know the IP address and the port number (usually 1080) of the SOCKS proxy server.

Configuring the ICA Java Client in Applet Mode

To configure the ICA Java Client in applet mode, create an HTML page using the following parameters:

Parameter	Description
ICASOCKSProxyPortNumber	Port number of the SOCKS proxy server (usually 1080)
ICASOCKSProxyHost	IP address of the SOCKS proxy server

For example, to connect an ICA Java Client running in applet mode to a server named Fountain using a SOCKS proxy with an address of 10.45.1.3, create an HTML page like the following:

```
<html>
<body>
<applet code-com.citrix.JICA.class
width=640 height=480>
<param name=address value=Fountain>
<param name=ICASOCKSProxyPortNumber value=1080>
<param name=ICASOCKSProxyHost value=10.45.1.3>
</applet>
</body>
</html>
```

Configuring the ICA Java Client in Application Mode by Editing the Launching Script

To make a connection in application mode

1. Locate your launching scripts for the ICA Java Client (usually in either \Citrix\JavaClient or /Citrix/JavaClient, depending on your platform).
2. Open for editing your script that launches a non-Program Neighborhood session. For example, on Windows-based systems, the file name is Jicasession.bat.
3. The last line in the script looks similar to this:

```
"jview" /cp:p "C:\Citrix\JavaClient\.;C:\Citrix\JavaClient\JICAEngJ.jar"
com.citrix.JICA %1 %2 %3 %4 %5 %6 %7 %8 %9
```

Edit this line to include the ICASOCKSProxyPortNumber and ICASOCKSProxyHost parameters. Place the parameters immediately after com.citrix.JICA. Your edited line should look similar to this:

```
"jview" /cp:p "C:\Citrix\JavaClient\.;C:\Citrix\JavaClient\JICAEngJ.jar"
com.citrix.JICA -ICASOCKSProxyPortNumber:1080 -ICASOCKSProsyHost:n.n.n.n
%1 %2 %3 %4 %5 %6 %7 %8 %9
```

where *n.n.n.n* is the IP address of your proxy server. Replace 1080 with the actual SOCKS proxy port number, if different.

Note When you start a non-Program Neighborhood session, a GUI command line window appears. You can enter these parameters into the GUI command line window instead of editing the launching script.

Configuring the ICA Java Client in Application Mode Using Program Neighborhood

To configure a connection to a SOCKS proxy server using Program Neighborhood

1. Start a Program Neighborhood session by running the appropriate launching script. For example, on Windows based systems, the file name is Pnsession.bat.
2. Click the application set or individual connection you want to configure. From the tool bar, click on Properties.
3. If you are configuring an application set, click **Firewalls** in the properties box.
4. Click the check box next to Connect via SOCKS proxy. Enter the SOCKS proxy's IP address in the Address of proxy to use box. Enter the proxy's port number (if different than 1080) in the Port box. See the online help for more information.

Note Because you can enter only one SOCKS proxy server address, you cannot configure Business Recovery with separate SOCKS settings for different business recovery servers.

Troubleshooting

The following list describes some known problems and suggests a workaround when applicable.

Note All problems in this list have been reported to the respective institutions.

Internet Explorer 4.0

- When you press the Back button when using Internet Explorer 4.0, the applet stops. If you revisit the page, the applet starts over again.

  To navigate to other pages while using Internet Explorer, open a separate Internet Explorer (or other browser) window.

  Notes Other browsers exhibit similar behavior under certain circumstances. For example, Netscape Navigator allows a limited number of applets to run. If you navigate off your ICA session and run several applets, when you navigate back your ICA session may start over again.

  Also, the EndSessionTimeout parameter controls how long you can leave an ICA session before it disconnects. The default value is 300 seconds. You can use this parameter to specify a different value in seconds.

- When using the ICA Java Client in applet mode on Internet Explorer 4.0, the browser does not correctly route ALT key events. For example, pressing ALT+F when an application running in the ICA session has focus sends the event to both the application and to the browser.
- When using scroll bars on Internet Explorer 4.0 for Windows NT 4.0, images sometimes repaint incorrectly.
- Internet Explorer 4.0 on UNIX client systems does not fully support signed .cab files.

  The Web server and the Citrix server must be the same machine.
- To display dead key (for example 10^2) characters using Microsoft Internet Explorer 4.0, you must press the chosen dead key and then press the space bar twice when connecting to a MetaFrame server.

Netscape Navigator

- Netscape Navigator 4.04 for OS/2 does not support applets in frames.
- Netscape 4.0 for Macintosh does not fully support 1.1 applets.

  Download the 1.0-compliant version of the ICA Java Client from http://www.citrix.com.
- Using the ICA Java Client as an applet on Netscape 4.0 Japanese with the Java 1.1 patch produces a “class not found error.”

  Download and install the Java Plugin from http://www.javasoft.com. Follow its instructions on how to set up the ICA Java Client applet tag for your browser.

OS/2

- OS/2 is not able to print as an applet due to Java security (Netscape 2.02 for OS/2 does not fully support signed .jar files).
- OS/2 using Netscape 4.04 and JDK 1.1.4 does not support Netscape security classes.

  Use JICAEngJ.jar. The Web server and the Citrix server must be the same machine.

HotJava

- HotJava browser 1.1 Final on Windows NT 4.0 fails to display applets when restoring a window that was previously maximized.

  You can resize the browser window and the applet will appear.
- HotJava browser 1.1.2 Final on Windows NT 4.0 does not run signed 1.1 applets.

  Use HotJava browser 1.1 Final or use the extracted class files.
- UNIX machines using HotJava Browser 1.1.2 or Netscape 4.0 cannot detect the Print Screen, Pause, or Scroll Lock keys.
- HotJava Browser 1.1.2 on UNIX does not detect the number pad keys correctly when connecting to a MetaFrame server.

  Use the conventional number keys.

Apple MRJ 2.1.x

- CTRL keys on Macintosh cannot be correctly detected when connecting to a Citrix server.
- To display dead key (for example 10^2) characters on Macintosh systems, you must press the chosen dead key and then press the space bar twice when connecting to a Citrix server.
- On a Macintosh client system with MRJ 2.1.x, the Shift key will not produce lower case letters when Caps Lock is on when connecting to a MetaFrame server.

  Do not use Caps Lock when doing mixed-case typing.
- Macintosh client systems using German keyboards do not make a distinction between the period and the number pad period key when connecting to a MetaFrame server.
- On Macintosh client systems using Microsoft Internet Explorer 4.0 with MRJ 2.1.x, applets flash when scrolling.

- Option keys on Macintosh may not be supported by the current Windows font in your ICA session. The ICA Java Client attempts a best match for the character.

  If the character produced is not the expected character, choose a Windows font in your ICA session that supports the character. After producing the desired character, you can switch back to your usual font.
- On Macintosh client systems using IE 4.0 with the Microsoft virtual machine (as an applet) or MRJ 2.1.x (as an application), the Esc key cannot be detected when connecting to a MetaFrame or *WINFRAME* server.
- Using MRJ 2.1.x on MacOS, typing the Number Pad equal sign (=) produces a minus when connecting to a *WINFRAME* server.
- Using MRJ 2.1.x on MacOS, typing the underscore key produces a minus when connecting to a *WINFRAME* server.
- Internet Explorer 4.0 on Macintosh client systems does not fully support signed .cab files.

  The Web server and the Citrix server must be the same machine.

JDK 1.1.4 and 1.1.5

- Using JDK 1.1.5 on Solaris 2.6, the PrintScreen, Pause, and Scroll Lock keys do not work with NumLock off when connecting to a *WINFRAME* server.

  Turn NumLock on.
- Using JDK 1.1.5 on Solaris 2.6, the F11 and F12 keys do not work when connecting to a *WINFRAME* server.
- Using JDK 1.1.5 on Windows NT, the Windows and Menu keys produce [,], / when connecting to a *WINFRAME* server.
- Using JDK 1.1.4 on OS/2 Warp 4.0, the Windows and Menu keys produce [,], / when connecting to a *WINFRAME* server.
- Using JDK 1.1.4 on OS/2 Warp 4.0, Alt + Enter does not work when connecting to a *WINFRAME* server.

Installation

- When installing on Windows NT, InstallShield Java Edition uses an .inf file to enter uninstall information into the end user's registry. If the end user is not logged on as an administrator, Windows NT displays an error message stating that the installation has failed. This message appears because the end user does not have access to the registry key that needs to be modified with uninstall information. For further information, see the Microsoft Knowledge Base Article Q169123, "Error Message: Installation Failed While Installing from .INF."

Log on with an administrative account or give the user in question (or the User's local group) full control permission to the **HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce** registry key.

As an alternative, you can install the ICA Java Client without having administrator privileges. The above error message will be displayed but can be ignored. However, if you installed the ICA Java Client without administrator privileges, you must follow the manual uninstall procedure in Chapter 2 to uninstall the client.

Audio

- Audio mapping feature does not work.

  Check with the vendor of your Java Virtual Machine to ensure that it supports the sun.audio package.
- Starting the ICA Java Client while an audio application is running on your desktop can cause the ICA Java Client to hang.

  Do not run audio applications while starting the ICA Java Client.
- After starting an ICA Java Client with audio enabled, local desktop audio applications do not have access to the audio system. Only the first invocation of the command line session has control of the audio system. Note that additional sessions started from the first command line session also have control of the audio system.

Miscellaneous

- Using Microsoft's Jview, buttons on dialog boxes do not automatically have keyboard focus.
- When connecting to a MetaFrame server, the ICA Java Client cannot detect the state of the Scroll Lock key when the client receives focus.

Common Errors

Running in Applet Mode

The error message "Can't find classes" appears.

Make sure the class files (either extracted or archived) are in the same directory as the HTML page. If they are not, make sure the codebase parameter specifies the URL of the class files. See "Creating a Web Page for Applet Mode Installations" in Chapter 2 for a description of how to set the codebase parameter.

If you are using an archive, make sure the HTML has the archive parameter and that it specifies the archive you are using. See “Creating a Web Page for Applet Mode Installations” in Chapter 2 for a description of how to set the archive parameter. See “Supported Archives” earlier in this chapter for a list of archives supported by some popular browsers.

The error message “Java security does not permit a connection to this application server” appears.

To connect to a Citrix server that is not physically on the same machine as the Web server that is the source of the ICA Java Client class files, you must use a signed archive. Make sure you specify a signed archive in the archive parameter of your HTML page.

Make sure your browser supports the type of signed archive you are attempting to use. See “Java 1.1 Compliance” in Chapter 1 for a list of popular browsers and the archives they support.

The error message “Class not found error” appears.

This error occurs when running the ICA Java Client, which is a 1.1 compliant Java application, with a 1.0-compliant Web browser. You must use a 1.1-compliant Web browser to run this version of the ICA Java Client. A 1.0-compliant version of the ICA Java Client is available from http://www.citrix.com for free download. The 1.0-compliant version of the ICA Java Client can be used with Java 1.0 Web browsers.

Running in Application Mode

The error message “Can’t find classes” appears.

If you are attempting to run the application from an archive, make sure you specify the archive as part of the classpath, as in C:\Application\JICAEng.zip.

Make sure your JVM supports running Java programs from archives. If you are not sure whether your JVM supports archives, extract one of the archives and run the application from the extracted classes. Extract the archive into the same directory the archive is in and set your classpath to that directory.

The error message “The application or server specified could not be found” appears.

Make sure you entered the Citrix server name or IP address correctly on the command line. Address parameter should be stated -address:*MFServer*, where *MFServer* is the name of the Citrix server.

The Citrix server or network connection might be down.

The error message "Class not found error" appears.

This error occurs when running the ICA Java Client, which is a 1.1-compliant Java application, on a 1.0-compliant Java virtual machine. You must use a 1.1-compliant Java virtual machine to run this version of the ICA Java Client. A 1.0-compliant version of the ICA Java Client is available from http://www.citrix.com for free download. The 1.0-compliant version of the ICA Java Client can be used with 1.0 Java virtual machines.

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

Y