Client for 32-bit Windows Administrator’s Guide

For other guides in this document set, go to the Document Center

MetaFrame Presentation Server Client for 32-bit Windows, Version 8.*x*

Citrix® MetaFrame® Presentation Server 3.0 for Windows®
Citrix® MetaFrame® Access Suite

Last Updated: February 24, 2004 11:43 am (BO)

Contents

Chapter 6 Securing Client Communication

Chapter 7 Updating the Clients

Introducing the MetaFrame Presentation Server Clients for 32-bit Windows

This manual is for system administrators responsible for installing, configuring, deploying, and maintaining the MetaFrame Presentation Server Clients for 32-bit Windows. These clients are available as Program Neighborhood, Program Neighborhood Agent, and the Web Client. This manual assumes knowledge of:

- The server farm to which your Clients connect
- The operating system on the client device (Windows 9x, Windows Me, Windows NT, Windows 2000, Windows XP, or Windows 2003)
- Installation, operation, and maintenance of network and asynchronous communication hardware, including serial ports, modems, and device adapters

This chapter introduces Version 8.*x* of the Citrix Clients for computers running 32-bit Windows operating systems. It is designed to help you decide which Clients to use in your computing environment and how to deploy them.

Note For information about Clients for other client devices and operating systems, see the documentation included on your MetaFrame Presentation Server Components CD or visit our Web site at http://www.citrix.com/.

This chapter covers the following topics:

- Overview
- Accessing Documentation
- Deciding Which Client to Use
- New in this Release

Overview

MetaFrame Presentation Server allows you to publish a variety of resources for remote access by users. These resources include applications (executables), content files (non-executables, such as text or video files), and entire server desktops. When referring to all types of resources you can publish, this document generally uses the term *published resources*. When referring to published content files or published desktops, this document uses the term *published content,* or *published desktop,* respectively.

MetaFrame Presentation Server Clients are the components of MetaFrame Presentation Server that users run on their computers to access resources published on servers running MetaFrame Presentation Server. The Clients combine ease of deployment and use, and offer quick, secure access to applications, content, and entire server desktops.

Accessing Documentation

This administrator's guide is part of the MetaFrame Presentation Server documentation set. The documentation set includes online guides that correspond to different features of MetaFrame Presentation Server. Online documentation is provided as Adobe Portable Document Format (PDF) files.

Use the *Document Center* to access the complete set of online guides. The Document Center provides a single point of access to the documentation that enables you to go straight to the section of documentation that you need. The Document Center includes:

- A list of common tasks and a link to each item of documentation.
- A search function that covers all the PDF guides. This is useful when you need to consult a number of different guides.
- Cross-references between documents. You can move between documents as often as you need using the links to other guides and the links to the Document Center.

Important To view, search, and print the PDF documentation, you need to have the Adobe Reader 5.0.5 with Search or a later version with Search. You can download Adobe Reader for free from Adobe Systems' Web site at http://www.adobe.com/.

If you prefer to access the guides without using the Document Center, you can navigate to the component PDF files using Windows Explorer. If you prefer to use printed documentation, you can also print each guide from Adobe Reader.

More information about Citrix documentation, and details about how to obtain further information and support, is included in *Getting Started with MetaFrame Presentation Server*.

Deciding Which Client to Use

Each Client offers a robust and easy-to-manage solution for delivering your published resources to users. To decide which Client or Clients best fit your needs, consider the way you want users to access your published resources and the way you want to manage this access.

For a complete list of Client features, refer to the Client Feature Matrix available from the Client Download page of the Citrix Web site (http://www.citrix.com).

Your Client Choices

Different enterprises have different corporate needs, and your expectations and requirements for the way users access your published resources may shift as your corporate needs evolve and grow. This section summarizes the clients you can use. The clients are discussed in more detail in the next section.

MetaFrame offers you a choice of several Clients for use on 32-bit Windows systems:

Program Neighborhood Agent

- Transparent integration of published resources with the user's desktop
- Central administration of user settings
- Native support for the full feature set of MetaFrame Presentation Server
- Requires the Web Interface for MetaFrame Presentation Server

Web Client (Two Versions)

- A smaller Client for quick distribution
- Web browser-based access to published resources from links on your Web page for Netscape and Internet Explorer users
- Support for most of the features of MetaFrame Presentation Server
- Available in .Cab file format for Internet Explorer users for quick download and installation
- A minimal installation version is available, packaged in .Cab format

Program Neighborhood

- Program Neighborhood user interface
- Requires initial user configuration
- Does not require the Web Interface for MetaFrame Presentation Server

Delivering Published Resources to Users

This section outlines the choices you have in delivering published resources to users, and which Client to use with each delivery method. For in-depth information about each Client, see the specific chapters for each Client later in this book.

Based on your corporate technology needs, you can choose among three different methods for delivering published resources to users. Published resources can be delivered on the desktop, through a Web browser, or through a user interface.

Access to Published Resources from Desktops

The Program Neighborhood Agent allows your users to access published resources entirely from a familiar Windows desktop environment.

Program Neighborhood Agent

User Experience. Users work with your published resources the same way they work with local applications and files. Published resources are represented throughout the client desktop, including the Start menu and the Windows system tray, by icons that behave just like local icons. Users can double-click, move, and copy icons, and create shortcuts in their locations of choice. The Program Neighborhood Agent works in the background. Except for a shortcut menu available from the system tray, it does not have a user interface.

Client Management and Administration. All users running the Program Neighborhood Agent connect to a central configuration file. Once launched, this client periodically downloads its configuration data from a configuration file on your server running the Web Interface. You can modify the configuration data at any time to dynamically manage and control your client population throughout your network from a single location and in real time.

The Program Neighborhood Agent requires the Web Interface and requires the presence on client devices of Microsoft Internet Explorer 5.0 or later, or of Netscape Navigator 4.78, 6.2, or later.

Because client-server data transfer occurs over standard HTTP or HTTPS protocols, you can use the Program Neighborhood Agent with firewalls using port 80 (for HTTP) or 443 (for HTTPS).

Note The configuration files stored on servers running the Web Interface can be edited using the Program Neighborhood Agent Console.

Access to Published Resources from Web Browsers

If you want users to access your published resources from within their familiar Web browsers, use the Web Client.

Web Client

User Experience. Users access published resources from within their Web browser by clicking links on a Web page you publish on your corporate intranet or the Internet. Clicking a link launches the published resource, either within the same browser window or in a new, separate browser window. The Web Client does not require user configuration. It works in the background and does not have a user interface.

Client Management and Administration. You can use the Web Client to access resources available from the Web Interface, and for access to resources published with traditional Application Launching and Embedding (ALE). Publish links to your resources with the Web Interface or by using an HTML wizard.

This Client requires the presence on client devices of Microsoft Internet Explorer 5.0 or later, or of Netscape Navigator 4.78, 6.2, or later.

The Web Client does not include a user interface or online Help files.This Client is quicker to download and install than the other Clients.

The Web Client is also packaged in .Cab file format for self-extraction and installation.

Web Client (Minimal Installation)

A smaller version of the Web Client, the minimal installation client is ideal for environments that do not require features such as COM port mapping, universal print driver, or client audio.

Access to Published Resources from a User Interface

If you want users to access your published resources from within a distinctive user interface, use Program Neighborhood.

Program Neighborhood

User Experience. Using Program Neighborhood, users can browse for sets of published resources (referred to as *application sets*) or create custom connections to individual published resources or to servers running MetaFrame Presentation Server. Icons representing application sets and custom ICA connections appear in the Program Neighborhood window.

Client Management and Administration. Unlike Program Neighborhood Agent, Program Neighborhood cannot be centrally configured. You must configure options for Program Neighborhood using its interface. For this reason, users running Program Neighborhood must be able to navigate through the interface easily and be able to understand the implications of any changes to their options.

Choose Program Neighborhood if you do not want to publish your resources using the Web Interface. If you choose to implement the Web Interface at a later time, Program Neighborhood users can also access resources published through the Web Interface. If you are planning to use the Web Interface and have not yet deployed any Clients, use the Program Neighborhood Agent or the Web Client.

New in this Release

Version 8.*x* of the Clients ships with MetaFrame Presentation Server 3.0 for Windows, and runs on Windows 9x, Windows NT 4 or later, Windows 2000, and Windows XP operating systems. It introduces a wide range of new features and performance improvements, and is fully backward compatible with earlier versions of Windows and MetaFrame XP feature releases.

Highlights of Version 8.*x* of the Clients include:

Digital Dictation Support. MetaFrame Presentation Server now supports client-side microphone input, which allows you to publish the latest dictation software for use in client sessions. Using local microphones, including a number of Philips SpeechMike speech processing devices, users can record dictations from a device in one location and then retrieve them for review or transcription from another device or location.

For example, a user away from the office can establish a client session to record notes using a laptop. Later in the day the user can retrieve the notes for review or transcription from the desktop device back at the office.

Digital dictation support is available with MetaFrame Presentation Server Advanced and Enterprise Editions. For information about configuring this feature, see the *MetaFrame Presentation Server Administrator's Guide*.

MetaFrame Presentation Server Client Packager. The MetaFrame Presentation Server Client Packager is an all-in-one MetaFrame Presentation Server Client for users of 32-bit Windows (Windows 95 and later) devices. It provides the following Clients in a single package:

- Program Neighborhood
- Program Neighborhood Agent
- Web Client

Note Each Client installation includes the Program Neighborhood Connection Center, allowing users to see information about their current ICA connections.

You can customize the client package to deploy and maintain any number and combination of MetaFrame Presentation Server Clients network-wide. Based on Windows Installer technology, the client packager lets you install, uninstall, modify, and repair MetaFrame Presentation Server Clients as well as perform controlled client upgrades. Place the client packager on a network share for users to download, or deliver it using Microsoft System Management Server or Active Directory Services. An easy-to-use wizard guides you through the configuration step by step.

You can access the client packager from the Autorun screen of the Components CD-ROM.

SpeedScreen Multimedia Acceleration. SpeedScreen Multimedia Acceleration optimizes streaming multimedia playback through published instances of remote desktop connections to Internet Explorer, Windows Media Player, and RealOne Player.

Session Reliability. Session Reliability enables sessions to remain open and on the screen when network connectivity is interrupted, thus allowing client users to view the application until the network connection is restored. This feature is especially useful for mobile users with wireless connections. If a user with a wireless link enters a tunnel and momentarily loses connectivity, the display on the client device freezes until connectivity resumes on the other side of the tunnel. Users continue to access the display during the interruption and can resume interaction with the application when the network connection is restored.

Workspace Control. Workspace control enables users to quickly switch between client devices and is especially useful to roaming or mobile users.

Dynamic Session Reconfiguration. This feature creates a smoother experience for users who switch between client devices with varying display modes by reconfiguring window appearance appropriately between devices. Users don't need to reconfigure the color depth and resolution for a session that they reconnect to on a client device with different display modes. Dynamic Session Reconfiguration automatically adapts the existing session's display mode to the reconnecting client device's display capabilities and mode preference.

Better Performance. New compression technology results in less data being sent over the network. This results in, for example, faster video rendering, file transfer, and printing, and provides a better overall user experience.

Zone Preference and Failover. A new policy rule enables you to direct user connections to preferred zones and set transparent failover to backup zones when preferred servers are unavailable. When users open applications, the Zone Preference and Failover policy rule directs their connections to the server with the highest zone preference and smallest load.

Compatibility with Asian Language Web Servers. In the past, some users of European language versions of MetaFrame Presentation Server Clients could not launch published applications from a Web Interface running on an Asian language Web server. This occured when the name of the application as it appeared on the Web Interface contained Asian language characters that the European language client was unable to recognize and process.

You can now configure the Web Interface to generate ICA files in Unicode, which increases the number of non-European language characters clients recognize. ICA files are text files that contain parameters and instructions for launching published applications.

Extended Unicode Keyboard Support. MetaFrame Presentation Server now supports handwriting and soft keyboard input. Soft keyboards are software-based, on-screen keyboards common in handheld devices such as PDAs running on Windows CE, and Tablet PCs.

In practical terms, this means that, in addition to the Western language characters available on hardware keyboards, users can input any of over 65,000 non-Western language characters that are defined in Unicode and available on soft keyboards only. This feature also provides support for voice input for users of Windows XP Tablet PC Edition.

Enhancements to Program Neighborhood Agent

Improved Support for User Name and Password Input. In earlier versions of Program Neighborhood Agent, when the Windows operating system running on the MetaFrame server failed to accept a user's user name or password, the user had to log off and launch a new session to continue. With the latest version of Program Neighborhood Agent, which ships with MetaFrame Presentation Server, users can now enter or reenter their Windows credentials within their session from the familiar Windows logon screen.

The Windows server operating system may fail to accept users or passwords for a number of reasons, including expired passwords and disabled user accounts.

Multiple Farm Support. You can now use Program Neighborhood Agent in MetaFrame deployments with more than one farm. When you configure the Web Interface to present users with applications from multiple farms, Program Neighborhood Agent automatically supports that configuration as well. For information about configuring the Web Interface, see the *Web Interface Administrator's Guide*.

Go to Document Center

Deploying Client Software to Users

The Components CD included in your MetaFrame Presentation Server media pack contains setup and installation files for all the MetaFrame Presentation Server Clients for 32-bit Windows. You can deliver Client software to your users using several methods, depending on the size of your organization and the available resources.

For a detailed discussion of the latest deployment methods available, see the *MetaFrame Presentation Server Administrator's Guide*. If you are using MetaFrame Presentation Server in conjunction with the Web Interface, see the *Web Interface Administrator's Guide* for information about deploying Clients in that environment.

This chapter covers the following topics:

- System Requirements
- The Installation Files for the Clients
- Using Microsoft Systems Management Server or Active Directory Services
- Deploying Clients over a Network
- Creating Client Installation Disks for Program Neighborhood
- Installation Options for the Clients
- Configuring the Windows Installer Package for Silent Installation
- Configuring the Self-Extracting Executables for Silent Installation

Important Do not install the Program Neighborhood Agent and Program Neighborhood on the same client device. Client features may be adversely affected and removal of one of the Clients will remove common registry settings.

System Requirements

To run Version 8.*x* of the Clients, client devices must meet the following requirements:

- Standard PC architecture, 80386 processor or greater as required for the operating system.
- Windows 9x, Windows Me, Windows NT 4 or later, Windows 2000, Windows XP, or Windows 2003.
- Available memory as recommended for the operating system by Microsoft.
- Internet Explorer Version 5.0 or later, or Netscape Navigator or Communicator Version 4.78, 6.2, or later.
- Microsoft mouse or 100% compatible mouse.
- VGA or SVGA video adapter with color monitor.
- High-density 3.5-inch disk drive (optional) and available hard drive space.
- Windows-compatible sound card for sound support (optional).
- For network connections to the server farm, a network interface card (NIC) and the appropriate network transport software are required.

  Supported connection methods and network transports are:

Protocol	Program Neighborhood Agent	Web Client	Program Neighborhood
TCP/IP+HTTP	X	X	X
SSL/TLS+HTTPS	X	X	X
TCP/IP		X	X
NetBIOS			X
IPX			X
SPX			X

  For information about configuring the Clients to use SSL or TLS to secure communications, see "Configuring and Enabling Clients for SSL and TLS" on page 86.
- For serial (dial-up) connections to the server farm (available with Program Neighborhood only), an internal modem or serial port and external modem using a 16550 Universal Asynchronous Receiver/Transmitter (UART) is recommended. This feature is not supported on Windows 2003 servers.

The Installation Files for the Clients

The setup and installation files for all the MetaFrame Presentation Server Clients for 32-bit Windows are located on the Components CD included in your MetaFrame Presentation Server media pack, in the following directory:

ICAWeb*language*\ica32, where *language* is the language of the Client software, such as:

- En (English)
- Fr (French)
- De (German)
- Ja (Japanese)
- Es (Spanish)

You can install the MetaFrame Presentation Server Clients for 32-bit Windows. using the following packages:

- Ica32Pkg.msi – the MetaFrame Presentation Server Client Packager for use with Windows 2000 Active Directory Services or Microsoft Systems Management Server, approximately 3MB in size
- Ica32a.exe – a self-extracting executable for Program Neighborhood Agent, approximately 3.55MB in size
- Ica32t.exe – a self-extracting executable for the Web Client, approximately 2.5MB in size
- Ica32.exe – a self-extracting executable for Program Neighborhood, approximately 4MB in size
- Wficat.cab – a cabinet file for the Web Client, approximately 2.1MB in size
- Wficac.cab – a cabinet file for the Web Client (minimal install), approximately 1.3MB in size
- Wfica.cab – a cabinet file for Program Neighborhood, approximately 3.8MB in size

An additional copy of the client packager, Ica32Pkg.msi, is located in the Icainst*language*\ica32\ directory of the Components CD.

Using Microsoft Systems Management Server or Active Directory Services

The MetaFrame Presentation Server Client Packager is an all-in-one client for users of 32-bit Windows (Windows 95 and later) devices. It wraps the following Clients into a single package:

- Program Neighborhood
- Program Neighborhood Agent
- Web Client

Note Each Client installation includes the Program Neighborhood Connection Center, allowing users to see information about their current ICA connections.

You can customize the client packager to deploy and maintain any number and combination of clients network-wide. Based on Windows Installer technology, the client packager lets you install, uninstall, modify, and repair clients as well as perform controlled client upgrades. An easy-to-use wizard guides you through the configuration step by step.

See your Windows 2000 or Systems Management Server documentation for more information.

Important To install the Client software using the Windows Installer package, the Windows Installer Service must be installed on the client device. This service is present by default on Windows 2000 systems. To install Clients on client devices running earlier versions of the Windows operating system, you must use the self-extracting executable or install the Windows Installer 2.0 Redistributable for Windows, available at http://www.microsoft.com/.

To uninstall a Client that was installed with a Windows Installer package, users must run the **Add/Remove Programs** utility from the Control Panel, or run the installer package again and select the **Remove** option.

Deploying Clients over a Network

In many environments, your users can access internal resources from network share points. You can centralize your client deployment from a single network share point, allowing you to provide users with setup files from one location.

To deploy Client executable software from a network share point

1. Create a share point on a file server that is accessible to your users.
2. Copy the required Client executable from the Components CD to the share point. For information about the executable files and their location, see "The Installation Files for the Clients" on page 19.
3. Supply your users with the path to the executable.
4. Users double-click the executable to begin the installation process.

To deploy the MetaFrame Presentation Server Client Packager from a network share point

1. Create a share point on a file server that is accessible to your users.
2. Copy the Client Packager from the Components CD to a local directory. For information about the Client Packager and its location, see "The Installation Files for the Clients" on page 19.
3. Create a share point on a file server that is accessible to your users, using the following command:
   Msiexec /a {path}file.msi

   where **{path}** is the UNC path to the share point and **file.msi** is the name of your install package.

   The Client Packer Setup wizard appears.
4. Enter the UNC path to the network share point.
5. Select your compression option and click **Next**.
6. Select one or more Clients to be included in the install package. If you select Program Neighborhood or Program Neighborhood Agent, the setup wizard for each Client appears.
7. On the **Upgrade Settings and Modify Options** page, choose whether the install package will upgrade or overwrite existing Clients.
8. Enable Remove, Repair, or Modify features.
9. On the **Select User Dialog Boxes** page, specify the dialog boxes displayed to users when they run the install package.

10. Verify your selections on the summary page and click **Finish**. The install package you specified in Step 2 is created in the specified UNC path.

You can use Active Directory Group Policy to install the client software or provide the network path to your users. See your Windows 2000 or Systems Management Server documentation for more information.

Important To install the Client software using the Windows Installer package, the Windows Installer Service must be installed on the client device. This service is present by default on Windows 2000 systems. To install Clients on client devices running earlier versions of the Windows operating system, you must use the self-extracting executable or install the Windows Installer 2.0 Redistributable for Windows, available at http://www.microsoft.com/.

Creating Client Installation Disks for Program Neighborhood

You can use the Client Creator to create client installation disks for Program Neighborhood. You will need three to four 3.5-inch, 1.44MB floppy disks to create the client installation disks.

To create Citrix Client installation disks (on servers running Windows 2000)

1. From a server running MetaFrame Presentation Server, select **ICA Client Creator** from the **Citrix** group folder. The **Make Installation Disk Set** dialog box appears.
2. In the Network Client or Service list, select the required Citrix Client. Select the **Format Disks** check box to format the disks when creating the installation media. Click **OK**.
3. Follow the on-screen instructions.

To create Citrix Client installation disks (on servers running Windows Server 2003)

Client Creator is not available on servers running Windows Server 2003. You must manually copy the files to preformatted floppies.

1. In the MetaFrame Presentation Server Component CD, navigate to the folder \ICAINST\en\ica32\disks\.
2. Copy the contents of each numbered folder onto corresponding preformatted floppy disks.

Installation Options for the Clients

For each package or setup executable, the Setup wizard guides you through the process of installing the Client software. When Setup begins, a series of information pages and dialog boxes prompts the user to select options and configure the product. In each installation, the user must accept the Citrix License Agreement before Setup will continue.

The following section describes the various options you configure during Setup. The options are presented in the order they appear for each Client. Depending on the components you choose to install, you may not encounter all configuration options described in this section, or you may encounter them in different order.

Installing the Program Neighborhood Agent

For information about the location of the Program Neighborhood Agent executable file, see “The Installation Files for the Clients” on page 19.

If users install the Program Neighborhood Agent with the self-extracting executable, they are presented with the following options:

Upgrade existing Client software. Setup searches the client device for previously installed versions of the Program Neighborhood Agent. If Setup detects a previous installation of the Program Neighborhood Agent, the user can either upgrade the existing client or create a new installation of the Program Neighborhood Agent. The default value is **Upgrade the existing client**.

Select Program Folder. Users can choose to use the default MetaFrame Presentation Server Client folder, specify the name of a new program folder, or add the Program Neighborhood Agent icon to an existing folder.

Specify the Server Address. Users must enter the URL of the server running the Web Interface to connect to in the format http://*servername* (for non-secure connections), or https://*servername* (for secure connections). Program Neighborhood Agent connects to the server at startup to get the latest configuration information including available published resources and permissions to change local settings.

Enable Pass-Through Authentication. Pass-through authentication allows the Client to access a user’s local Windows user name, password, and domain information and pass it to the server. Users are not prompted to log on to the Program Neighborhood Agent separately.

You must enable this logon mode in the configuration file on the server running the Web Interface to make it available to users. See “Configuring Program Neighborhood Agent from a Central Console” on page 32 for instructions.

Important If users select **No** during the installation process, they must reinstall the Program Neighborhood Agent if they decide to use pass-through authentication at a later time.

Specify the Client Name. Servers running MetaFrame Presentation Server use the client name to manage client printers and other system resources. By default, the machine name is used as the client name. If you do not assign a unique machine name to each client device, device mapping and application publishing may not operate correctly.

Installing the Web Client

The full Web Client is available as a self-extracting executable and as a .cab file. The Web Client setup files are significantly smaller than the other Clients. The smaller size allows users to more quickly download and install the client software. You can configure the Web Client for silent user installation.

The Web Client (Minimal Installation) is a smaller version of the Web Client designed to support the core functions of MetaFrame Presentation Server for users running Internet Explorer. This version of the Web Client is the smallest available for use with MetaFrame Presentation Server. Use this client when your environment requires a small download and minimal functionality, or in a "locked down" environment where installing a traditional client may not be allowed by security settings.

For information about the location of the Web Client installation and setup files, see "The Installation Files for the Clients" on page 19.

Installing the Web Client requires minimal user interaction. After a user accepts the Citrix License Agreement, Setup copies files to the client device. By default, the Web Client is installed in the Program Files\Citrix\Icaweb32 directory.

Important If users are running Netscape Navigator, they must restart the browser.

You can provide a link in a Web page for users to install the Web Client.

To insert the Web Client (Minimal Installation) in a Web page

Add the following code to a Web page to prompt the download of the Wficac.cab file:

```
<OBJECT
        classid="clsid:238f6f83-b8b4-11cf-8771-00a024541ee3"
        data="np.ica"
        CODEBASE="http://web-server-root/some-directory/
wficac.cab"
        width='640'
        height='480'
        hspace='2'
        vspace='2'>
        <param name="Start" value="Auto">
        <param name="Border" value="On">
</OBJECT>
```

Important Add the site or sites from which the .cab file is downloaded to the Trusted Sites zone.

If you are using the Web Client with the Web Interface, see the *Web Interface Administrator's Guide* for more information about deploying this Client.

Installing Program Neighborhood

For information about the location of the Program Neighborhood executable file, see “The Installation Files for the Clients” on page 19.

If users install Program Neighborhood with the self-extracting executable, they are presented with the following options:

Upgrade existing Client software. Setup searches the client device for previously installed versions of Program Neighborhood. If Setup detects a previous installation of Program Neighborhood, the user can upgrade the existing client or create a new installation of Program Neighborhood. The default value is **Upgrade the existing client**.

Choose Destination Location and Select Program Folder. Users can change the default installation path and the default Program folder.

Select Client Name. Servers running MetaFrame Presentation Server use the client name to manage client printers and other system resources. By default, the machine name is used as the client name. If you do not assign a unique machine name to each client device, device mapping and application publishing may not operate correctly.

Enable Pass-Through Authentication. Pass-through authentication allows the Client to access a user's local Windows user name, password, and domain information and pass it to the server. Users are not prompted to log on to Program Neighborhood separately.

Important If users do not enable pass-through authentication during the installation process, they must reinstall Program Neighborhood if they decide to use pass-through authentication at a later time.

Configuring the Windows Installer Package for Silent Installation

You can configure the MetaFrame Presentation Server Client Packager for "silent" user installation. Windows Installer informs the user when the client software is successfully installed. When prompted, the user must clear the Windows Installer message box.

Tip If you are deploying the Windows Installer Package from a network share, follow the instructions on page 21 and remove all dialog boxes displayed to the user.

For information about the location of the MetaFrame Presentation Server Client Packager, see "The Installation Files for the Clients" on page 19.

To configure the MetaFrame Presentation Server Client Packager for silent installation

1. At a command prompt, type:

   msiexec /I "<MSI_Package>" /qb- /L "<Log File Path>" [Key=Value]…

   where <MSI_Package> is the name of the installer package.

   Note Quotation marks are required only if the package name or log file path contains spaces.

2. Set the following keys as required:

PROGRAM_FOLDER_NAME=<Start Menu Program Folder Name>, where <Start Menu Program Folder Name> is the name of the Programs folder on the Start menu containing the shortcut to the Program Neighborhood Agent software. The default value is **Citrix\MetaFrame Access Clients**. This function is not supported during client upgrades.

INSTALLDIR=<Installation directory>, where <Installation directory> is the location where the client software is installed. The default value is **C:\Program Files\Citrix\ICA Client**.

CLIENT_NAME=<ClientName>, where <ClientName> is the name used to identify the client device to the server farm. The default value is **%COMPUTERNAME%**.

ENABLE_DYNAMIC_CLIENT_NAME={**Yes** | **No**}. To enable dynamic client name support during silent installation, the value of the property ENABLE_DYNAMIC_CLIENT_NAME in your installer file must be **Yes**. To disable dynamic client name support, set this property to **No**.

CLIENT_ALLOW_DOWNGRADE={**Yes** | **No**} By default, this property is set to **No**. This prevents an installation of an earlier version of the client. Set to **Yes** to allow an installation of an earlier version of the client.

CLIENT_UPGRADE={**Yes** | **No**} By default, this property is set to **Yes**. This installs the Client if an earlier version of the Client is already installed.

ENABLE_SSON={**Yes** | **No**}. The default value is **No**. If you enable the **SSON** (Pass-through authentication) property, set the **ALLOW_REBOOT** property to **No** to avoid automatic rebooting of the client system.

Important If you disable pass-through authentication, users must reinstall the Client if you decide to use pass-through authentication at a later time.

ALLOW_REBOOT={**Yes** | **No**}. The default value is **Yes**.

DEFAULT_NDSCONTEXT=<Context1 [,…]>. Include this parameter if you want to set a default context for Novell Directory Services (NDS). If you are including more than one context, place the entire value in quotation marks and separate the contexts by a comma.

Examples of correct parameters:

```
DEFAULT_NDSCONTEXT=Context1
```

```
DEFAULT_NDSCONTEXT="Context1,Context2"
```

Example of an incorrect parameter:

```
DEFAULT_NDSCONTEXT=Context1,Context2
```

SERVER_LOCATION=<Server_URL>. The default value is **Web Server**. Enter the URL of the server running the Web Interface that hosts the configuration file. The URL must be in the format http://<*servername*> or https://<*servername*>.

Note The Program Neighborhood Agent appends the default path and file name of the configuration file to the server URL. If you change the default location of the configuration file, you must enter the entire new path in the **SERVER_LOCATION** key.

Configuring the Self-Extracting Executables for Silent Installation

You can configure numerous settings before you deploy the Client software to your users. Silent installation allows users to install the Client without selecting many of the installation options. For Program Neighborhood, users can begin using it immediately without having to configure several of the required settings.

Important You can use any standard compression utility to extract the contents of the installer file. However, you must use commercially available software to repackage the contents for distribution to users.

For information about each of the executable files and their location, see "The Installation Files for the Clients" on page 19.

Configuring the Program Neighborhood Agent for Silent Installation

You can limit user interaction with the self-extracting executable setup program by entering values in the Install.ini file before you deploy the setup program to your users.

To configure Program Neighborhood for silent installation

1. Extract the Client files from Ica32a.exe using your preferred compression utility software, or by typing at a command line:

   Ica32a.exe -a -unpack:<*Directory Location*>

   where <*Directory Location*> is the path to the directory to which you want to extract the client files.

2. Locate and open the Install.ini file in a text editor.

   You can set the following parameters. When you enter values for these parameters, setup dialog boxes do not appear on the user's screen.

   ServerURL= URL for the server running the Web Interface. The default value is **Web Server**. Enter the URL of the server running the Web Interface in the format **http://*servername*** or, for SSL-secured communications, **https://*servername***.

   SetMachineNameClientName= Enter **Yes** to accept the Windows machine name as the client device name.

   Location= Enter the installation location. Use <PROGRAM_FILES> if you want to install the files in a directory in the Program Files folder.

   StartMenu= Enter the Start menu path. The path you enter here is appended to the Programs folder of the Start menu.

   InstallSingleSignOn= Enter **Yes** to enable pass-through authentication.

   Important If you disable pass-through authentication, users must reinstall the Client if you decide to use pass-through authentication at a later time.

   AcceptClientSideEULA= Enter **Yes** to accept the end-user license agreement.

3. Save the file and exit the text editor.
4. Repackage the client files for distribution to your users.

Configuring the Web Client for Silent Installation

You can limit user interaction with the Web Client Setup by suppressing the appearance of the initial user prompt and the Citrix License Agreement.

To configure the Web Client for silent installation

1. Extract the Client files from Ica32t.exe using your preferred compression utility.
2. Locate and open the Ctxsetup.ini file in any text editor.
3. To suppress the initial user prompt, locate the **InitialPrompt** parameter. Change the value of the setting from 1 to 0.
4. To suppress the Citrix License Agreement dialog box, locate the **DisplayLicenseDlg** parameter and set the value to 0.
5. Save the file and exit the text editor.
6. Repackage the client files for distribution to your users.

Configuring Program Neighborhood for Silent Installation

You can customize many Program Neighborhood settings, including default application sets, server location, screen display resolution, and encryption level, among others. General instructions for preconfiguring the client are included below. For definitions of parameters in the client .ini files, see the *Configuration Guide for the Clients* on the Citrix Web site at http://www.citrix.com/support; select **Product Documentation**.

Important When Program Neighborhood is installed on a client device, several of the .ini files you can modify are copied to the user’s profile directory. If you modify settings for a new version of Program Neighborhood prior to updating with the Client Auto Update feature, your changes are not migrated to the .ini files under the user’s profile directory.

To configure the self-extracting executable for silent installation

1. Obtain a copy of the Client installer file (Ica32.exe).
2. Extract the contents of the installer file to a new folder, A.
3. Install the Client using the same installer file you used in Step 2.
4. Start the client and customize the client settings, as desired, from the client user interface.
5. When you are done, open the %User Profile%\Application Data\ICAClient folder and copy all files with an .ini extension to a new folder, B.
6. In folder B, replace all .ini file extensions with .src (source) extensions.
7. Copy the contents of folder B to folder A, overwriting the existing .src files in folder A with their modified equivalents from folder B. The contents of folder A now represent your custom installer set of .ini files.
8. Repackage the contents of folder A for distribution to users.

Configuring the Program Neighborhood Agent

The Program Neighborhood Agent is a client designed for flexibility and ease of configuration. Shortcuts to published resources available from the Web Interface can be integrated into users' desktops. For example, links to published applications can be displayed in users' Start menus or on their desktops with no need to open a Web browser or an additional application to launch the applications. You can determine what, if any, configuration options your users can access and modify, such as audio, display, and logon settings.

Important The Program Neighborhood Agent requires the Web Interface for MetaFrame Presentation Server.

Configure the options and settings for Program Neighborhood Agent using the Program Neighborhood Agent Console on your server running the Web Interface. Each time users log on to the Program Neighborhood Agent, they see the most recent Program Neighborhood Agent configuration. Changes made while users are connected take effect when the client configuration is refreshed after a designated interval.

This chapter explains how to configure the Program Neighborhood Agent and use the Program Neighborhood Agent Console. The following topics are covered:

- Overview of the Program Neighborhood Agent
- Configuring Program Neighborhood Agent from a Central Console
- Configuring Settings Using the Console
- Providing Users with Workspace Control

Overview of the Program Neighborhood Agent

Using the Program Neighborhood Agent in conjunction with the Web Interface, you can integrate published resources with users' desktops. Users access remote applications, desktops, and content by clicking icons on their Windows desktop, in the Start menu, in the Windows system tray, or any combination thereof.

The Program Neighborhood Agent handles the following functions:

- **User authentication.** The Client provides user credentials to the Web Interface when users try to connect and every time they launch published resources.
- **Application and content enumeration.** The client presents users with their individual set of published resources.
- **Application launching.** The client is the local engine used to launch published applications.
- **Desktop integration**. The client integrates a user's set of published resources with the user's desktop.
- **User preferences.** The client validates and implements local user preferences.

For a complete list of Program Neighborhood Agent features, refer to the Client Feature Matrix available from the Client Download page of the Citrix Web site (http://www.citrix.com).

Configuring Program Neighborhood Agent from a Central Console

Users' logon methods, shortcuts, and access to user interface elements are determined by the options you set in the Program Neighborhood Agent Console. You can allow or deny users the ability to determine their own logon method, audio settings, shortcut placement, and display settings, depending on your company needs and security requirements.

The Program Neighborhood Agent Console

The Program Neighborhood Agent Console is a Web browser-based tool designed to allow you to customize the settings and functionality of the Program Neighborhood Agent for your users. You can determine default settings for shortcuts, display size and colors, audio, logon method, authentication, and other functions. In addition, you can allow or deny users the ability to customize many of these options. The Console is available from servers running the Web Interface.

Features of the Program Neighborhood Agent Console include:

- Concurrency control, allowing multiple administrators to view the same configuration file without overwriting each other's changes
- Restricted access, determined by Web Security settings

Each client reads the configuration data from the server when a user launches the Program Neighborhood Agent, and updates at specified intervals. This allows the client to dynamically display the options you want your users to see based on the data received.

Connecting to the Console

To access the Program Neighborhood Agent Console, connect to
http://*servername*/Citrix/PNAgentAdmin/
where *servername* is the name of a server running the Web Interface.

Instructions about how to use the Console are integrated into the interface, making it easy for you to work with the console from any location using your Web browser.

Configuration Files

The options configured with the Program Neighborhood Agent Console are stored in a configuration file on your server running the Web Interface. The configuration file controls the range of parameters that appear as options in the users's **Properties** dialog box. Users can choose from available options to set preferences for their ICA sessions, including logon mode, screen size, audio quality, and the locations of links to published resources. When a configuration file is loaded into the Console, the Console automatically creates a backup file (with the extension .bak).

A default configuration file, Config.xml, is installed with default settings and is ready for use without modification in most network environments. However, you can edit the file or create multiple configuration files to suit your needs using the Program Neighborhood Agent Console. This allows you to add or remove a particular option for users quickly and to easily manage and control your users' displays from a single location.

The Config.xml file is stored in the \Inetpub\wwwroot\Citrix\PNAgent directory on the server running the Web Interface. New and backup configuration files that you create using the Program Neighborhood Agent Console are stored in the same folder as the default configuration file.

You can create multiple configuration files to fill all of your organization's needs using the Program Neighborhood Agent Console. The settings you save in a single configuration file affect all users who read from that file. If you create a new configuration file, you must have your users change the server URL in Program Neighborhood Agent to point to the new file.

Important The settings in the configuration file are global; the settings affect all users connecting to that instance of the file. Changes you make to a configuration file affect all users served by it.

Configuring Settings Using the Console

The Program Neighborhood Agent Console is divided into several sections, allowing you to control and define different aspects of the user experience. You can:

- Control whether or not users see all the tabs in their Properties dialog box
- Control which options users see in their Properties dialog box
- Secure the communications between the client and the server running the Web Interface
- Determine the refresh frequency for client configurations
- Specify authentication options for users
- Determine where links to published resources appear
- Set display options for ICA sessions
- Allow users to change where links to published resources appear
- Allow users to manage their ICA sessions with the Program Neighborhood Agent.

Before deploying Program Neighborhood Agent throughout your network, you can test your configuration by installing Program Neighborhood Agent on a single client device. You can then evaluate the default settings and determine whether or not you want to make adjustments to fit your particular network needs.

Controlling Tab Display for the Properties Dialog Box

By default, users can access the **Program Neighborhood Agent Properties** dialog box from the Windows system tray. You can choose to hide or display tabs using the **Client Tab Control** section of the Program Neighborhood Agent Console.

The **Properties** dialog box can display up to five tabs: **Server**, **Application Display**, **Application Refresh**, **Session Options**, and **Reconnect Options**. Depending on your network needs, you may not want certain options to be available to users. You can modify or disable a particular option, or hide a tab altogether.

Configuring Server URL Options

The Server URL points the Program Neighborhood Agent to the correct configuration file. The default path is determined based on the server address entered by the user during installation of the agent. Using the Program Neighborhood Agent Console, you can configure options related to the Server URL from the **Server Settings** page.

You can allow users to change the server URL. If you do, users see the Server URL option on the **Server** tab of their Properties dialog box.

You can configure server connection and configuration refresh settings, such as redirection of users to a server running the Web Interface using its Fully Qualified Domain Name (FQDN) or a user-provided server URL. In addition, you can define how often the client should refresh its configuration settings.

You can define when users are redirected to a different server — at connection time or a scheduled client refresh.

Securing Client Communication with the Web Interface

Smart card logon and SSL/TLS-secured communications between the client and the server running the Web Interface are not enabled by default. You can enable SSL/TLS communication from the **Server Settings** page, forcing URLs to apply the HTTPS protocol automatically.

In addition, you must enable SSL on the server running MetaFrame Presentation Server. See “Securing the Program Neighborhood Agent with SSL/TLS” on page 90 for more information.

Specifying Available Logon Methods

Providing a choice of multiple logon modes may be necessary in environments where multiple users employ the same client device but different logon modes. This allows you to determine what logon methods are available to users, to force a default logon method, and to allow users to save their passwords. The available logon methods are **Anonymous logon**, **Smart card logon, Smart card pass-through authentication**, **Prompt user**, and **Pass-through authentication**.

If you select multiple logon methods, users can choose their preferred logon method from a drop-down list. You can allow Novell Directory Services (NDS) credentials from the specified tree in conjunction with the **Prompt user** and **Pass-through authentication** logon methods.

By default, users who are prompted for credentials cannot save their passwords. To enable this function, select the **Allow user to save password** check box in the **Logon Methods** page of the Program Neighborhood Agent Console.

If you do not want users to have access to any of these options, you can use the **Client Tab Control** section of the Program Neighborhood Agent Console to hide the **Server** tab altogether. You can show or hide the tab at any time. For instructions about hiding and showing this tab, see "Controlling Tab Display for the Properties Dialog Box" on page 34.

Important If users do not enable pass-through authentication when they install the Program Neighborhood Agent, they must reinstall the client software before they can use pass-through authentication.

Controlling Application Display Options

Users can change the options available on the **Application Display** tab to insert links to published resources in various locations of their client device, including the Windows desktop, the **Start** menu, the Windows system tray, or any combination thereof.

Using the Program Neighborhood Agent Console, you can define which settings users are allowed to customize. The client queries the configuration file at connection time to validate each user preference against its controlling element in the file.

You can configure when shortcuts are deleted and how published resources are displayed on users' Start menus, desktops, or in their system trays.

If you do not want users to have access to any of these options, you can use the **Client Tab Control** section of the Program Neighborhood Agent Console to hide the **Application Display** tab altogether. You can show or hide the tab at any time. For instructions about hiding and showing this tab, see "Controlling Tab Display for the Properties Dialog Box" on page 34.

Refreshing Published Resource Information

The Program Neighborhood Agent periodically queries the server running the Web Interface to obtain an up-to-date list of published resources. Users can control how often this "refresh" is performed by the agent on the **Application Refresh** tab of their Properties dialog box.

The **Application Refresh** tab is hidden from the **Properties** dialog box by default. If you want to give users control over the refresh rate, you need to enable the tab first. For instructions about hiding and showing this tab, see "Controlling Tab Display for the Properties Dialog Box" on page 34.

By default, if you enable the **Application Refresh** tab, all options on the tab can be customized by users accessing the same configuration file. From the **Application Refresh** page of the Console, you can configure which individual options appear in the users' **Application Refresh** tab.

Setting Session Options

Using the **Session Options** page of the Program Neighborhood Agent Console, you can define the window size, color depth, and sound quality of ICA sessions and which settings are available to the user.

The preferences users set for color depth and sound quality affect the amount of bandwidth the ICA session consumes. To limit bandwidth consumption, you can force the server default for some or all of the options on this tab.

Forcing the server default removes all settings for the corresponding option, other than **Default**, from the interface. The settings configured on the server running the Web Interface are applied to connections from each client.

If you do not want users to have access to any of these options, you can use the **Client Tab Control** section of the Program Neighborhood Agent Console to hide the **Session Options** tab altogether. You can show or hide the tab at any time. For instructions about hiding and showing this tab, see "Controlling Tab Display for the Properties Dialog Box" on page 34.

Providing Users with Workspace Control

Workspace Control enables users to move between client devices and gain access to all of their applications when they log on. Workspace Control provides users the ability to quickly disconnect from all running applications, to reconnect to applications, or to log off from all running applications.

From the **Workspace Control** page of the Program Neighborhood Agent Console, you can configure—and allow users to configure—how they reconnect:

- **At log on.** By default, Workspace Control enables users to automatically reconnect to all running applications when logging on, bypassing the need to reopen individual applications, including applications from which the user disconnected, as well as any active applications currently running on another client device. Disconnecting from an application leaves the application running on the server. If you have roaming users who need to keep some applications running on one client device while they reconnect to a subset of their applications on another client device, you can configure the logon reconnection behavior to open only the applications from which the user disconnected previously.

- **Using a Reconnect button.** After logging on to the server farm, users can reconnect to all their applications at any time by clicking **Reconnect**. By default, Reconnect opens both applications that are disconnected plus any active applications currently running on another client device. You can configure Reconnect to open only those applications from which the user disconnected previously.

If you allow users to override your settings, the settings are available to users from the **Reconnect Options** tab of their Properties dialog box.

Workspace Control is enabled in the server farm by default and is available only for users accessing applications through the Web Interface or the Program Neighborhood Agent. For more information about enabling and configuring Workspace Control for users, see the *Web Interface Administrator's Guide*.

Customizing the Program Neighborhood Agent as a User

This section presents general information about customizing user preferences on the client device running the Program Neighborhood Agent.

To customize user preferences for the Program Neighborhood Agent

1. In the Windows system tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.
2. On each tab, make the desired configuration changes.
3. Click **OK** to save your changes.

For more detailed information, see the online Help for the Program Neighborhood Agent.

Changing the Server URL

The Program Neighborhood Agent requires the URL to a configuration file (Config.xml is the default configuration file) on the server running the Web Interface. You may ask your users to change the server URL as you create new configuration files or delete old ones.

Tip To prevent users from accidentally changing their server URL, disable the option or hide the Server tab entirely.

To change the server URL in Program Neighborhood Agent

1. In the Windows system tray, right-click the Program Neighborhood Agent icon and choose **Properties**. The **Server** tab displays the currently configured URL.

2. Click **Change** and enter the server URL in the format http://<*servername*>*,* or https://<*servername*> to encrypt the configuration data using SSL.
3. Click **Update** to apply the change and return to the **Server** tab.
4. Click **OK** to close the **Properties** dialog box.

To delete memorized server URLs

1. In the Windows system tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.
2. Select the **Server** tab and click **Change**.
3. Click the down arrow to view the entire list of memorized server URLs.
4. Right-click the URL you want to delete and select **Delete**.
5. Click **Update**.
6. Click **OK**.

Configuring Program Neighborhood

Use Program Neighborhood if you are not using the Web Interface to deliver published resources. Program Neighborhood has its own user interface—the Program Neighborhood window — from which users browse for application sets or create custom ICA connections to servers running MetaFrame Presentation Server or to published applications.

This Client can be used with the Web Interface. However, if you are planning to use the Web Interface and have not yet deployed any Clients, use the Program Neighborhood Agent or the Web Client.

This chapter explains how to configure Program Neighborhood. The following topics are covered:

- Connecting to Published Resources
- Improving Performance Over Low-Bandwidth Connections

Important Unlike Program Neighborhood Agent, Program Neighborhood cannot be centrally configured. You must configure options for Program Neighborhood using its user interface. For this reason, users running Program Neighborhood must be able to navigate through the interface easily and be able to understand the implications of any changes to their options.

For step-by-step instructions about how to configure Program Neighborhood, see the Program Neighborhood online help.

Connecting to Published Resources

With Program Neighborhood, users can connect to published resources and servers running MetaFrame Presentation Server using:

- Application sets
- Custom ICA connections

An *application set* is a user’s view of the resources published on a given server farm that the user is authorized to access. Resources published in an application set are preconfigured for such session properties as window size, number of colors, supported encryption levels, and audio compression rate.

If these settings are not required to run the published application (such as, for example, the audio compression rate), you can change them on the client device at the application set level.

Important Application set functionality is not available for applications published on servers running MetaFrame Presentation Server for UNIX. To connect to an application published on these servers, users must create a custom ICA connection.

A *custom ICA connection* is a user-defined shortcut to a published application or server desktop. While you can create custom ICA connections to connect to any server desktop or published application, you must use custom ICA connections to connect to:

- A server running MetaFrame Presentation Server outside of a server farm scope of management
- An application published prior to the installation of a MetaFrame 1.8 server that cannot be migrated into a server farm
- An application published on a server running MetaFrame Presentation Server for UNIX

Applications published in this way are not enabled for automatic configuration of Program Neighborhood sessions.

With Program Neighborhood, users can connect to a server in the server farm using one of the following methods:

- Dialing into a server running MetaFrame Presentation Server using a modem installed on the client device (serial connection).
- Establishing a custom ICA connection over a direct serial cable (serial connection).

- Using the local or wide-area network connection between the client device and the server running MetaFrame Presentation Server. This method uses one of the following network protocols:
  - TCP/IP+HTTP
  - SSL/TLS+HTTPS
  - TCP/IP
  - IPX
  - SPX
  - NetBIOS

Note Remote users can connect to servers running Windows Server 2003 over TCP/IP only. Terminal Services in Windows Server 2003 does not support remote connections over IPX/SPX, NetBIOS, and asynchronous transports.

With Microsoft Remote Access Service (RAS) or Dial-Up Networking (DUN) in combination with the Client, users can connect to a server running MetaFrame Presentation Server. For this type of connection, the client device must meet the following requirements:

- The RAS or DUN client software must be installed on the client device
- The RAS server or third-party PPP server must be located on the same network as the server running MetaFrame Presentation Server

ICA Browsing

ICA browsing is a process in which a Client transmits data to locate servers on the network and get information about the applications published in the server farm.

For ICA browsing, Clients communicate with the Citrix XML Service or the ICA browser, depending on the browsing protocol selected in the Client.

ICA browsing occurs when:

- Users launch published applications. The client sends a request to locate the application on a server.
- Program Neighborhood users display the **Application Set** list in the Find New Application Set wizard.
- Program Neighborhood users display the **Server** or **Published Application** list in the Add New ICA Connection wizard to create a custom ICA connection.

Communicating with the Citrix XML Service

Citrix XML Service, a component of MetaFrame Presentation Server, is installed by default on all servers running MetaFrame Presentation Server. The XML Service communicates published application information to Clients using HTTP protocol and XML data. The XML Service also communicates published application information to servers running the Web Interface.

For example, when a user launches a published application from Program Neighborhood, the Client sends a request for the application. The XML Service responds with the address of a server on which the application is published.

With the Web Interface, for example, a user connects to a Web page using a Web browser. The XML Service provides a list of available resources to the server running the Web Interface. The server then displays the available resources on the user's personalized application Web page.

For more information about the Citrix XML Service, see the *MetaFrame Presentation Server Administrator's Guide*.

Specifying the Network Protocol for ICA Browsing

As described above, ICA browsing is a process that locates servers and published applications in response to requests from a Client. Changing the network protocol setting allows you to control the way the Client searches for servers running MetaFrame Presentation Server and how it communicates with them.

For step-by-step instructions about configuring the network protocol, see the Program Neighborhood online help. For more information about using SSL to secure client-to-server communication, see "Configuring and Enabling Clients for SSL and TLS" on page 86.

This section discusses the TCP/IP+HTTP, SSL/TLS+HTTPS, and TCP/IP protocols and their implementation in a server farm. For additional information about configuring server farms for ICA browsing, see the *MetaFrame Presentation Server Administrator's Guide* included in your MetaFrame Presentation Server media pack.

Important SSL/TLS+HTTPS or TCP/IP+HTTP retrieves information only on a per-server farm basis. To retrieve information from more than one server farm, you must configure server location settings for each application set. For custom ICA connections, you must configure server location settings for each ICA connection. Do not place addresses from separate farms in the same server location list.

Using TCP/IP+HTTP for ICA Browsing

Program Neighborhood uses TCP/IP+HTTP as the default network protocol. The Client uses the HTTP protocol to search for servers running MetaFrame Presentation Server. Select this protocol when Clients connect over the Internet or through a firewall or proxy server.

Using the TCP/IP+HTTP protocol for ICA browsing provides the following advantages for most server farms:

- TCP/IP+HTTP uses XML data encapsulated in HTTP packets, that the Client sends to port 80 by default. Most firewalls are configured so port 80 is open for HTTP communication.
- TCP/IP+HTTP does not use UDP (User Datagram Protocol) or broadcasts to locate servers in the server farm.
- Routers pass TCP/IP packets between subnets, which allows Clients to locate servers that are not on the same subnet.

By default, if no server is specified, the Client attempts to resolve the name “ica” to an IP address. This is indicated by the virtual server location “ica” in the **Address List** box. This feature allows the Domain Name System (DNS) or Windows Internet Naming Service (WINS) administrator to configure a host record that maps “ica” to a valid server IP address that can service XML requests from Clients.

You must map a server running MetaFrame Presentation Server to the default name of “ica” on your network or you must specify at least one IP address in Program Neighborhood.

Tip You can configure the DNS server for the Clients to use round-robin DNS to map the name “ica” to a set of servers that can service the XML requests. Use this approach to avoid individually configuring server location addresses on your client devices.

You can specify servers to contact for ICA browsing by entering IP addresses or DNS names of servers running MetaFrame Presentation Server in the **Address List** box in Program Neighborhood. You can define up to three groups of servers: 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 and the first server to respond is the one to which you connect.

To locate the Citrix XML Service, the Client makes an HTTP connection to port 80 on the server running MetaFrame Presentation Server. If the user is launching a published application, for example, Citrix XML Service sends to the Client the address of a server running MetaFrame Presentation Server that has the application published.

Using SSL/TLS+HTTPS for ICA Browsing

With SSL/TLS+HTTPS as the network protocol, the Client uses the HTTPS protocol to search for a list of servers running MetaFrame Presentation Server. The client communicates with the server using ICA with SSL/TLS. SSL/TLS+HTTPS provides strong encryption of ICA traffic and server authentication. Select this option when using SSL or TLS communication over the Internet or through a firewall or proxy server.

Important By default, IIS and SSL/TLS for ICA connections are set to port 443. If your users are configured to use port 443 for IIS, you must specify another port number for SSL Relay after you install the certificate for SSL/TLS.

If you select **SSL/TLS+HTTPS** as the network protocol, you must enter the fully qualified domain name of the server hosting the digital certificate.

Note The TCP/IP+HTTP and SSL/TLS+HTTPS protocols can be used only with compatible servers running MetaFrame Presentation Server. See the *MetaFrame Presentation Server Administrator's Guide* for Windows or UNIX for information about configuring the server running MetaFrame Presentation Server to use SSL/ TLS.

Using TCP/IP for ICA Browsing

With TCP/IP as the network protocol, Clients send UDP broadcasts to the ICA browser service on port 1604 to locate published applications and servers running MetaFrame Presentation Server. Select this option if all servers running MetaFrame Presentation Server and all Clients are located on the same network.

With TCP/IP as the network protocol, the default setting for server location is **(Auto-Locate)**. The auto-locate function works as follows:

1. The Client broadcasts a "Get Nearest MetaFrame server" packet. The first server running MetaFrame Presentation Server to respond returns the address of the master ICA browser, which is used in the next step.
2. The Client sends a request for the server and published application lists to the master ICA browser.
3. The master ICA browser responds with a list of all servers on the network running MetaFrame Presentation Server and a list of all published applications.

To eliminate broadcasts on your network, or if your network configuration uses routers or gateways, you can set a specific server address for the server that functions as the master browser.

Important By default, server farms operating in native mode do not respond to Clients that use UDP broadcasts for ICA browsing. Therefore, if Clients are configured to use TCP/IP and to auto-locate servers, they will fail to locate servers or published applications in a server farm running MetaFrame XP or later.

Because UDP broadcast packets do not traverse subnets, using broadcasts for ICA browsing works only if a server that responds to broadcasts is on the same subnet as the Client. When the Client locates a server, it communicates using directed (not broadcast) UDP to port 1604.

Because of broadcast limitations, you might prefer to enter one or more IP addresses or DNS names of servers running MetaFrame Presentation Server in the **Address List** box in Program Neighborhood. You must do this if the Client is not on the same subnet as a data collector.

Improving Performance Over Low-Bandwidth Connections

If you are using a low-bandwidth connection, such as a modem, you can make a number of changes to your client configuration and the way you use the Client to improve performance.

In addition, Citrix recommends that you use the latest version of MetaFrame Presentation Server and its Clients. Citrix continually enhances and improves performance with each release. Many performance features require the latest client and server software to function.

Changing Your Client Configuration

On devices with limited processing power, or in circumstances where only limited bandwidth is available, there is a trade-off between performance and functionality. The Client provides both user and administrator with the ability to choose an acceptable mixture of rich functionality and interactive performance. Making one or more of these changes can reduce the bandwidth your connection requires and improve performance:

- **Enable data compression.** Compression reduces the size of the data that is transferred over the ICA connection. Enable compression and specify the maximum compression parameter.
- **Enable the bitmap cache.** Bitmap caching stores commonly used bitmaps (images) locally on your client so that they do not have to be transferred over the ICA connection every time they are needed.

- **Queue mouse movements and keystrokes.** When queuing is enabled, the Client sends mouse and keyboard updates less frequently to the server running MetaFrame Presentation Server. Enabling this option improves performance only if you are using a low-bandwidth connection.
- **Enable SpeedScreen latency reduction.** SpeedScreen latency reduction improves performance over high latency connections by providing instant feedback to the user in response to typed data or mouse clicks.
- **Reduce the window size.** Change the window size to the minimum size you can comfortably use.
- **Reduce the number of colors.** Reduce the number of colors to 256.
- **Reduce sound quality.** If client audio mapping is enabled, reduce the sound quality to the minimum setting.

Changing Client Use

ICA technology is highly optimized and typically does not have high CPU and bandwidth requirements. However, if you are using a very low-bandwidth connection, the following tasks can impact performance:

- **Accessing large files using client drive mapping.** When you access a large file with client drive mapping, the file is transferred over the ICA connection. On slow connections, this may take a long time.
- **Printing large documents on local client printers.** When you print a document on a local client printer, the print file is transferred over the ICA connection. On slow connections, this may take a long time.
- **Playing multimedia content.** Playing multimedia content uses a lot of bandwidth and can cause reduced performance.

Configuring Features Common to the Clients

This chapter explains how to configure and use features common to the Clients. The following topics are covered:

- Configuring Features New to Version 8.x of the Clients
- Configuring Existing Features Common to the Clients
- Connecting to MetaFrame Presentation Servers for UNIX

Configuring Features New to Version 8.*x* of the Clients

This section discusses Client configuration of new features. If you want more information about server configuration for these features, see the *MetaFrame Presentation Server Administrator's Guide* or, if your users access applications through the Web Interface, see the *Web Interface Administrator's Guide.*

Session Reliability

With session reliability, users continue to see a published application's window if the connection to the application experiences an interruption. For example, wireless users entering a tunnel may lose their connection when they enter the tunnel and regain it when they come out on the other side. During such interruptions, the session reliability feature enables the session window to remain displayed while the connection is being restored.

To reduce the likelihood that users continue to click links or type text while the connection is being restored, mouse pointers become hourglass icons while the application is unresponsive.

Users of Program Neighborhood can select the **Enable session reliability** option in their application or connection settings. Enabling or disabling **Enable session reliability** at the client overrides the session reliability settings for the server farm. Users of Program Neighborhood Agent and the Web Client cannot override the server settings for session reliability.

To enable session reliability in Program Neighborhood, select **Enable session reliability** on the **Options** tab of the Settings for the application set.

SSPI/Kerberos Security for Pass-Through Authentication

Rather than sending user passwords over the network, pass-through authentication now leverages Kerberos authentication in combination with Security Support Provider Interface (SSPI) security exchange mechanisms. Kerberos is an industry-standard network authentication protocol built into Microsoft Windows operating systems.

Kerberos logon offers security-minded users the convenience of pass-through authentication combined with secret-key cryptography and data integrity provided by industry-standard network security solutions. With Kerberos logon, the Client does not need to handle the password and thus prevents Trojan horse-style attacks on the client device to gain access to users' passwords.

Users can log on to the client device with any authentication method, for example a biometric authenticator such as a fingerprint reader, and still access published resources without further authentication.

System requirements. Kerberos logon requires MetaFrame Presentation Server 3.0 or later, MetaFrame Presentation Server Clients for 32-bit Windows 8.*x* or later, and works only between clients and servers that belong to the same or to trusted Windows 2000 or Windows 2003 domains. Servers must also be *trusted for delegation*, an option you configure through the Active Directory Users and Computers management tool.

Kerberos logon is *not available* in the following circumstances:

- Connections for which you select any of the following options in Terminal Services Configuration:
  - On the **General** tab, the **Use standard Windows authentication** option
  - On the **Logon Settings** tab, the **Always use the following logon information** option or the **Always prompt for password** option
- Connections you route through the Secure Gateway for MetaFrame Presentation Server
- If the server running MetaFrame Presentation Server requires smart card logon
- If the authenticated user account requires a smart card for interactive logon

Important SSPI requires XML Service DNS address resolution to be enabled for the server farm, or reverse DNS resolution to be enabled for the Active Directory domain. For more information, see the *MetaFrame Presentation Server Administrator's Guide*.

Configuring Kerberos Authentication

The client, by default, is not configured to use Kerberos authentication when logging on to the server. You can set the client configuration to use Kerberos with or without pass-through authentication. Using Kerberos without pass-through authentication is more secure than using Kerberos with pass-through authentication.

- **Kerberos without pass-through authentication**

  With this configuration the user logs on using Kerberos authentication only. If Kerberos Logon fails for any reason, the user is prompted for credentials. Kerberos can fail due to a missing operating system requirement, such as the requirement that the server be trusted for delegation. This configuration is supported only for Web Interface or Custom ICA Connections made through Program Neighborhood. For Program Neighborhood Application Sets and the Program Neighborhood Agent, the user is prompted for credentials. To configure Kerberos logon for the Web Interface, see the *Web Interface Administrator's Guide*.

  To deploy Kerberos without pass-through authentication, Citrix recommends that you create a "Kerberos only" client package using the MetaFrame Presentation Server Client Packager. To create a client package, you can execute Autorun.exe on the Components CD and select the option to Create a custom Windows client installation package. During Setup, configure clients to use the local name and password for logging on and select the option to **Use Kerberos only**.

  Tip During the client packager Setup, you can select dialog boxes that you want to be displayed to users. You should accept the default configuration that the "Single Sign On" dialog box is Hidden. Otherwise, users can override your configuration and set their client configuration to use pass-through authentication (single sign-on).

  You can also configure Kerberos by modifying the settings of the Wfclient.ini file in the Citrix\ICA Client directory on a client device. For this method of configuration, you must modify Wfclient.ini on each client device for which you want to use Kerberos without pass-through authentication.

To configure the Wfclient.ini file on the client device for Kerberos logon:

1. Ensure that the Wfclient.ini has the setting **SSPIEnabled=off.**
2. From Program Neighborhood open **ICA Settings** from the **Tools** menu and clear **Pass-Through Authentication** and click **OK**.
3. Log off from the client device and log back on.
4. With a text editor, open the Wfclient.ini from the Citrix\ICA Client directory and modify the following settings to:

   SSPIEnabled=on

   UseSSPIOnly=on
5. From Program Neighborhood open **ICA Settings** from the **Tools** menu. **Pass-Through Authentication** is now selected and you should select **Use local credentials to log on**.

- **Kerberos with pass-through authentication**

  When client configurations are set to use Kerberos with pass-through authentication, the client attempts to use Kerberos authentication first and uses pass-through authentication if Kerberos fails.

CAUTION This configuration is less secure than using Kerberos without pass-through authentication. The user cannot disable this client configuration from the user interface. You must use Kerberos with pass-through authentication if you want to use Kerberos with Program Neighborhood Application Sets or with Program Neighborhood Agent.

You can configure a client device to use Kerberos with pass-through authentication by modifying the settings of the Wfclient.ini file in the Citrix program files on a client device. Change **SSPIEnabled=off** to **SSPIEnabled=on** in the [WFClient] section of Wfclient.ini.

If you change these settings for use with the Program Neighborhood Agent, the Program Neighborhood Agent must be closed and restarted on the client device in order for the settings to be applied.

Workspace Control

Workspace Control provides users with the ability to quickly disconnect from all running applications, reconnect to applications, or log off from all running applications. You can move between client devices and gain access to all of your applications when you log on. For example, health care workers in a hospital can move quickly between workstations and access the same set of applications each time they log on to MetaFrame Presentation Server. These users can disconnect from multiple applications at one client device and open all the same applications when they reconnect at a different client device.

Important Workspace control is available only to users connecting to published resources with Program Neighborhood Agent or through the Web Interface.

User policies and client drive mappings change appropriately when you move to a new client device. Policies and mappings are applied according to the client device where you are currently logged on to the session. For example, if a health care worker logs off from a client device in the emergency room of a hospital and then logs on to a workstation in the hospital's X-ray laboratory, the policies, printer mappings, and client drive mappings appropriate for the session in the X-ray laboratory go into effect for the session as soon as the user logs on to the client device in the X-ray laboratory.

Important Workspace control cannot be used with earlier versions of the Clients for 32-bit Windows and works only with sessions connected to servers running MetaFrame Presentation Server Version 3.0

To configure Workspace Control settings

If the Workspace Control configuration settings of the Presentation Server Console or the Web Interface Console are configured to allow users to override the server settings, users can configure Workspace Control in the Settings options of the Web Interface or the **Reconnect Options** tab of Program Neighborhood Agent Properties.

The following options are available in Program Neighborhood Agent Properties on the **Reconnect Options** tab:

- **Enable automatic reconnection at logon** allows you to reconnect to disconnected applications or both disconnected and active applications
- **Enable automatic reconnection from Reconnect menu** allows you to reconnect to disconnected applications or both disconnected and active sessions

For users launching applications through the Web Interface, similar options are available from the **Settings** page:

- **Enable automatic reconnection at logon** allows you to reconnect to disconnected applications or both disconnected and active applications
- **Enable automatic reconnection from Reconnect menu** allows you to reconnect to disconnected applications or both disconnected and active sessions
- **Customize Log Off button** allows you to configure whether or not the log off command will include logging you off from applications that are running in the session

If users log on with smart card, smart card with single sign-on, or single sign-on authentication, you must set up a trust relationship between the server running the Web Interface and any other server in the farm that the Web Interface accesses for published applications. For more information about Workspace Control requirements and server configuration, see the *MetaFrame Presentation Server Administrator's Guide* or the *Web Interface Administrator's Guide.*

Digital Dictation Support

MetaFrame Presentation Server now supports client-side microphone input. This allows you to publish dictation software for use in client sessions. Using local microphones, including a number of Philips SpeechMike speech processing devices, users can record dictations with applications running on the server.

For example, a user away from the office can establish a client session to record notes using a laptop. Later in the day the user can retrieve the notes for review or transcription from the desktop device back at the office.

Digital dictation support is available with MetaFrame Presentation Server Advanced and Enterprise Editions. For information about configuring this feature, see the *MetaFrame Presentation Server Administrator's Guide*.

Users of Program Neighborhood and Program Neighborhood Agent can disable their microphones by selecting **No** in the Client Audio Security dialog box available from the Program Neighborhood Connection Center (for seamless connections), or from either the Program Neighborhood Connection Center or the client’s system menu (for non-seamless connections). Web Client users are presented with the same dialog box automatically at the beginning of their sessions.

On the Client, users control audio input and output in a single step—by selecting an audio quality level from the **Settings** dialog box (for Program Neighborhood) or from the **Properties** dialog box (for Program Neighborhood Agent).

Important When using the Winscribe software with a Philips foot pedal device, you do not need to configure the Winscribe software to use a foot pedal– t works automatically.

Configuring Existing Features Common to the Clients

This section explains how to configure existing features that are common to the Clients. For configuration instructions specific to each Client, see the appropriate chapter about the Client you plan to use.

The following topics are discussed in this section:

- Dynamic Client Name Support
- SpeedScreen Browser Acceleration
- Windows NT Challenge/Response (NTLM) Support
- Certificate Revocation List Checking
- User-to-User Shadowing
- Smart Card Support
- Auto Client Reconnect
- Novell Directory Services Support
- Disabling DNS Name Resolution
- Mapping Client Drives
- Mapping Client Printers
- Mapping Client COM Ports
- Mapping Client Sound Support
- Configuring Multiple Monitors

Dynamic Client Name Support

Dynamic client name support allows the client name to be the same as the machine name. When users change their machine name, the client name changes to match. This allows you to name machines to suit your naming scheme and find connections more easily when managing your server farm.

If the client name is not set to match the machine name during installation, the client name does not change when the machine name is changed.

Configuring Dynamic Client Name Support During Installation

Users enable dynamic client name support by selecting **Enable Dynamic Client Name** during client installation. Doing so sets the client name the same as the machine name.

To enable dynamic client name support during silent installation, the value of the property ENABLE_DYNAMIC_CLIENT_NAME in your installer file must be **Yes**. Set the property to **No** to disable dynamic client name support.

SpeedScreen Browser Acceleration

This function, available to users running Internet Explorer 5.5 or later, enhances the speed at which images are downloaded and displayed. To enable SpeedScreen browser acceleration, set **SpeedScreenBA** to **ON** (or **OFF** to disable it) in your .ica file.

SpeedScreen browser acceleration must be enabled on the server to be available to the Client. If SpeedScreen browser acceleration is enabled on the Client, but not the server, SpeedScreen browser acceleration is disabled.

Important To increase performance and media coverage for SpeedScreen MultiMedia Acceleration, Citrix recommends you upgrade client devices to the latest version of Microsoft's DirectX runtime. SpeedScreen MultiMedia Acceleration could experience problems that cause playback to default to legacy audio and display settings on devices with versions of DirectX earlier than 7.0. For example, client devices running NT 4.0 cannot be upgraded beyond DirectX 6.0.

Windows NT Challenge/Response (NTLM) Support

Version 7.0 and later of the Clients provide support for networks using Windows NT Challenge/Response (NTLM) for security and authentication. NTLM authentication is supported by default on machines running Windows NT, Windows 2000, and Windows XP.

On client devices running Windows 95, Windows 98, and Windows Me, manually enable the **User Control Package** to use NTLM authentication. Change this by going to **Control Panel > Network > Access Control**. On the **Access Contro**l screen, select **User-level access control**. If the User Control Package is not enabled, the Client uses basic authentication, not NTLM authentication.

Certificate Revocation List Checking

When certificate revocation list checking is enabled, the Clients check whether or not the server's certificate is revoked. This feature improves the cryptographic authentication of the server running MetaFrame Presentation Server and improves the overall security of the SSL/TLS connections between a Client and a server running MetaFrame Presentation Server.

You can enable several levels of certificate revocation list checking. For example, you can configure the client to check only its local certificate list or to check the local and network certificate lists. In addition, you can configure certificate checking to allow users to log on only if all Certificate Revocation Lists are verified.

To enable certificate revocation list checking

1. On the server running the Web Interface, locate and open the Template.ica file.
2. Configure the **SSLCertificateRevocationCheckPolicy** setting to one of the following options:
   - **NoCheck -** No certificate revocation list checking is performed
   - **CheckWithNoNetworkAccess -** The local list is checked
   - **FullAccessCheck -** The local list and any network lists are checked
   - **FullAccessCheckAndCRLRequired -** The local list and any network lists are checked; users can log on if all lists are verified

If you do not set **SSLCerticicationRevocationCheckPolicy**, it defaults to **NoCheck** for Windows NT 4. For Windows 2000 Server and Windows XP, the default setting is **CheckWithNoNetworkAccess**.

User-to-User Shadowing

No client-side configuration is required to use this feature. You shadow a user from a client device using the published Shadow Taskbar.

For information about using the Shadow Taskbar, see the Shadow Taskbar help. For information about enabling and configuring this feature, see the *MetaFrame Presentation Server Administrator's Guide*.

Smart Card Support

MetaFrame Presentation Server smart card support is based on Microsoft Personal Computer/Smart Card (PC/SC) standard specifications. MetaFrame Presentation Server supports smart cards and smart card devices only that are, themselves, supported by the underlying Windows operating system. A discussion of security issues related to PC/SC standards compliance is beyond the scope of this document.

To select smart card-based logon (Program Neighborhood Agent)

1. In the Windows system tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.
2. Select the **Server** tab.
3. From the **Logon mode** menu, select **Smart card logon** or **Smart card with Single sign-on authentication**.

   With **Smart card logon** selected, the Client prompts the user for a smart card PIN (Personal Identification Number) when it starts up and every time the user requests a published resource.

   With **Smart card with Single sign-on authentication** selected, the Client prompts the user for a smart card PIN when it starts up. The Client then caches the PIN and passes it to the server every time the user requests a published resource. The user does not have to subsequently reenter a PIN to access published resources.
4. Click **OK** to close the **Properties** dialog box.

To set smart card-based logon using the Program Neighborhood Agent Console, see "Configuring Settings Using the Console" on page 34.

To select smart card-based logon (Program Neighborhood)

1. For an application set, select the application set and click **Properties** on the Program Neighborhood toolbar. For a custom ICA connection, select the custom ICA connection and click **Settings** on the Program Neighborhood toolbar.
2. Select the **Logon Information** tab.
3. Select **Smart Card**.
4. Select **Pass-through authentication** to cache the PIN and pass it to the server every time the user requests a published resource.

Note Microsoft strongly recommends that only smart card readers tested and approved by the Microsoft Windows Hardware Quality Lab (WHQL) be used on computers running qualifying Windows operating systems. Visit http://www.microsoft.com/ for additional information about hardware PC/SC compliance.

MetaFrame Presentation Server does not control smart card PIN management. PIN management is controlled by the cryptographic service provider for your cards.

Auto Client Reconnect

Users can be disconnected from their ICA sessions because of unreliable networks, highly variable network latency, or range limitations of wireless devices. With the auto client reconnection feature, the Client can detect unintended disconnections of ICA sessions and automatically reconnect users to the affected sessions.

When this feature is enabled on a server running MetaFrame Presentation Server, users do not have to reconnect manually to continue working. The Client attempts to reconnect to the session until there is a successful reconnection or the user cancels the reconnection attempts. If the MetaFrame administrator requires user authentication, a dialog box requesting credentials is displayed to a user during automatic reconnection. Automatic reconnection does not occur if users exit applications without logging off. Users can reconnect only to disconnected sessions.

Changing Default Auto Reconnect Settings

If you want to disable auto client reconnect for a particular user, you must add the following line to the **[WFClient]** section of the Appsrv.ini file located in the user's %User Profile%\Application Data\ICA Client directory:

TransportReconnectEnabled=Off

Novell Directory Services Support

When launching Client software, users can log on and be authenticated using their NDS credentials. Supported NDS credentials are user name (or distinguished name), password, directory tree, and context.

NDS support is integrated into the following:

- **Program Neighborhood Agent and Program Neighborhood Client.** If NDS is enabled in the server farm, NDS users enter their credentials on an NDS tab on the Client logon screen. If users have the Novell Client (Version 4.8) installed, they can browse the NDS tree to choose their context. See "Novell Directory Services Support" on page 59 for information about additional configuration necessary to enable NDS support for the Program Neighborhood Agent.
- **Pass-Through Authentication.** If users have the Novell Client (Version 4.8) installed, you can pass their credentials to the server running MetaFrame Presentation Server, eliminating the need for multiple system and application authentications.

  To enable pass-through authentication, configure the following policy options in the User Package in ZENworks for Desktops:

  1. Enable the **Dynamic Local User** policy option.
  2. Set the **Use NetWare Credentials** value to **On.**
- **Custom ICA Connections.** When users run the Add New ICA Connection wizard, they must enter a distinguished name in the user name field and a password in the password field. Users must leave the domain field blank.
- **The Web Interface for MetaFrame Presentation Server.** NDS users enter their credentials on an NDS logon screen provided by the Web Interface. See the *Web Interface Administrator's Guide* for information about configuring your server for NDS.

Note To use NDS logon information with earlier versions of Clients, enter the NDS tree name in the **Domain** field and a distinguished name in the **User** field on the Client logon screen.

Setting a Default Context for NDS

You can set a default context for NDS for Program Neighborhood and for the Program Neighborhood Agent. To set a default context for NDS, you must configure the particular installer file you are using to deploy the Clients:

MetaFrame Presentation Server Client Packager

For instructions about setting a default context for NDS in the MetaFrame Presentation Server Client Packager, see "Configuring the Windows Installer Package for Silent Installation" on page 26.

Self-Extracting Executable

For general information about configuring the self-extracting executable for Program Neighborhood, see “Configuring Program Neighborhood for Silent Installation” on page 30.

To set a default context for NDS in the self-extracting executable

1. Extract the client file set from Ica32.exe as outlined in “Configuring the Self-Extracting Executables for Silent Installation” on page 28.
2. Locate and open Appsrv.src in a text editor.
3. Add the following parameter to the **[WFClient]** section:

   DefaultNDSContext=<Context1 [,…]>.

   If you are including more than one context, separate the contexts by a comma.
4. Save and close the file.

Using Windows NT Credentials with the Novell Client and Pass-Through Authentication

If Program Neighborhood is configured to use pass-through authentication on a client device that has the Novell Client installed, Program Neighborhood, by default, uses the NDS credentials to authenticate the user to the server running MetaFrame Presentation Server. If you want the Client to use the user’s Windows NT credentials with pass-through authentication instead, you must add a parameter to the Appsrv.ini file on the client device. You can make the addition to the Windows Installer package before distributing it, or you can configure clients on individual client devices after installation is complete.

Configuring the Windows Installer Package Prior to Installation

For information about configuring the Window Installer package for use of Windows NT credentials with pass-through authentication on client devices that have the Novell Client installed, see “Configuring the Windows Installer Package for Silent Installation” on page 26.

Configuring Individual Clients After Installation

1. Locate and open the user-level Appsrv.ini file in a text editor. By default, this file is located in the %User Profile%\Application Data\ICA Client directory.
2. Add the following parameter to the **[WFClient]** section:

   SSOnCredentialType=NT
3. Save and close the Appsrv.ini file.

DNS Name Resolution

You can configure Clients that use the Citrix XML Service to connect to the server farm to request a Domain Name System (DNS) name for a server instead of an IP address.

Important Unless your DNS environment is configured specifically to use this feature, Citrix recommends that you do not enable DNS name resolution in the server farm.

Program Neighborhood is configured to use TCP/IP+HTTP (the XML Service) browsing by default. Clients connecting to published applications through the Web Interface also use the XML Service. For Clients connecting through the Web Interface, the Web server resolves the DNS name on behalf of the Client.

DNS name resolution is disabled by default in the server farm and enabled by default on the Clients. When DNS name resolution is disabled in the farm, any client request for a DNS name will return an IP address. There is no need to disable DNS name resolution on the client.

Disabling DNS Name Resolution

If you are using DNS name resolution in the server farm and are having problems with specific client workstations, you can disable DNS name resolution for those workstations using the following procedure.

To disable DNS name resolution on the Clients for Win32

1. Open the user-level Appsrv.ini file. By default, this file is located in the %User Profile%\Application Data\ICA Client directory.
2. Change the line **xmlAddressResolutionType=DNS-Port** to **xmlAddressResolutionType=IPv4-Port**.
3. Save and close the Appsrv.ini file.
4. Repeat Steps 1 through 3 for each user of the client workstation.

Enabling Extended Parameter Passing

With extended parameter passing you can associate a file type on a client device with an application published on a server running MetaFrame Presentation Server. When a user double-clicks a locally saved file, the file is opened by the application associated with it on the server running MetaFrame Presentation Server.

For example, if you associate all text-type files on the client device with the application "Notepad" published on the server running MetaFrame Presentation Server, opening a locally saved text-type file on the client device causes Notepad to open on the server running MetaFrame Presentation Server.

Note The Program Neighborhood Agent supports content redirection, a feature introduced in MetaFrame XP, Feature Release 2. Functionally equivalent to extended parameter passing, content redirection allows you to enforce all underlying file type associations from the server running MetaFrame Presentation Server, eliminating the need to configure extended parameter passing on individual client devices.

If all users are running the Program Neighborhood Agent, and if you want to take advantage of the administrative ease of content redirection from client to server, see the *MetaFrame Presentation Server Administrator's Guide* for more information.

Enabling extended parameter passing requires both server- and client-side configuration. On the server, add the %* (percent and asterisk symbols) tokens to published applications. These tokens act as placeholders for client-passed parameters. For instructions about configuring support for parameter passing, see the *MetaFrame Presentation Server Administrator's Guide*.

On the client side, you must replace the **open** command for the file type with a command line that passes the file name and path to the server running MetaFrame Presentation Server. You must enable extended parameter passing on each client device you want to use this feature.

Configuring Extended Parameter Passing

File type association data is stored in the Windows registry. To associate a file type on the client device with the published application, you need to replace the **open** command for the file type with a command line that passes the file name and path to the application published on the server running MetaFrame Presentation Server.

Important MetaFrame Presentation Server supports the ISO8859-1 character code for western European languages, including English, and the ShiftJIS character code for Japanese. You must use one of these two character codes to establish file type associations.

The command line you create must include the following elements:

- The file name of the Client executable used to launch the published application
- The name of the published application to launch, in the correct syntax
- The parameter passing arguments

The next section explains how to determine which Client executable to include in the command line.

Determining the Client Executable

Users can connect to published applications using the following methods:

- Finding and launching an application in an application set using Program Neighborhood
- Creating and launching a custom ICA connection using Program Neighborhood
- Launching an .ica file (.ica files are placed on the client device when the user connects using the Web Interface)

Each of these methods launches the published application using a different executable on the client device. The following table lists which executable you must include in the parameter passing command line based on the user’s connection method.

Connection method	ICA Client executable
Custom ICA connections (using Program Neighborhood Client)	Wfcrun32.exe
Applications identified in ICA files (including connecting using the Web Interface)	Wfica32.exe
Applications in application sets (using Program Neighborhood)	Pn.exe

The following section explains how to identify the published application with the correct syntax.

Identifying Published Applications

Each Client executable uses different command line syntax to specify configuration data when launching published applications. When creating your command line, you must use the command line syntax to correctly identify the published application.

Note To view the required command line syntax for an executable from a command prompt, change directories to the installation directory of the Client and then type the name of the executable followed by **/?** (forward slash question mark).

Command Line Syntax for Wfcrun32.exe

To use Wfcrun32.exe to launch a custom ICA connection, specify:

C:\Program Files\Citrix\ICA Client\wfcrun32.exe "<application name>"

Command Line Syntax for Wfica32.exe

To use Wfica32.exe to launch a published application described in an ICA file, specify:

C:\Program Files\Citrix\ICA Client\wfica32.exe <file_name>.ica

Command Line Syntax for Pn.exe

To use Pn.exe to launch a custom ICA connection, specify:

C:\Program Files\Citrix\ICA Client\pn.exe /app:"<application name>"

To use Pn.exe to launch an application published in an application set, specify:

C:\Program Files\Citrix\ICA Client\pn.exe /pn:"<application set name>" / app:"<application name>"

Note To use Pn.exe to launch an application in an application set, the application must exist in the Pn.exe application cache.

Including Parameter Passing Arguments

When you determine the launching executable and identify the application, you must include the parameter passing arguments **/param:"%1"**.

The sample command line below associates text-type files with the published application "Notepad Text Editor" in the application set "Production Farm."

C:\Program Files\Citrix\ICA Client\pn.exe /pn:"Production Farm" /app:"Notepad Text Editor" /param:"%1"

Entering Parameter Passing in the Windows Registry

When you assemble the required elements of the new command line, you must enter the new command in the Windows registry. You can access the **open** command for the file types you want to associate through the **Folder Options** dialog box in Control Panel. For instructions about editing the **open** command for a file type, see the online Help for the Windows operating system of the client device.

The following example command lines combine the required elements into a working Client command line.

To associate text files with a custom published application named "Notepad Text Editor" launched using Pn.exe, specify:

C:\Program Files\Citrix\ICA Client\pn.exe /app:"Notepad Text Editor" /param:"%1"

To associate text files with an application named "Notepad Text Editor" that is published in an application set called "Production Farm," specify:

C:\Program Files\Citrix\ICA Client\pn.exe /pn:"Production Farm" /app:"Notepad Text Editor" /param:"%1"

To associate text files with a custom published application named "Notepad Text Editor" launched using Wfcrun32.exe, specify:

C:\Program Files\Citrix\ICA Client\wfcrun32.exe "Notepad Text Editor" /param:"%1"

To associate text files with an application identified in an ICA file named Notepad.ica, using Wfica32.exe as the launching executable, specify:

C:\Program Files\Citrix\ICA Client\wfica32.exe Notepad.ica /param:"%1"

Important The above examples assume that the client devices are connecting to servers running MetaFrame Presentation Server that contain remapped server drives. If your server drives are not remapped, you must add the following text to the argument: \\client\; for example: **/param:"\\client\%1"**.

Mapping Client Devices

The MetaFrame Presentation Server Client supports mapping devices on client devices so they are available from within an ICA session. Users can:

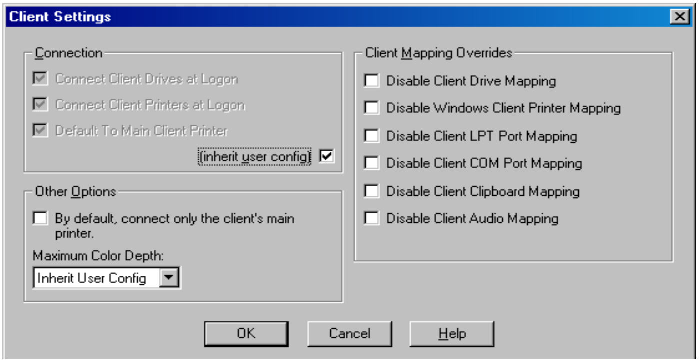
- Transparently access local drives, printers, and COM ports
- Cut and paste between the ICA session and the local Windows clipboard
- Hear audio (system sounds and .wav files) played from the ICA session

During logon, the Client informs the server running MetaFrame Presentation Server of the available client drives, COM ports, and LPT ports. By default, client drives are mapped to server drive letters and server print queues are created for Client printers so they appear to be directly connected to the server running MetaFrame Presentation Server. These mappings are available only for the current user during the current session. They are deleted when the user logs off and recreated the next time the user logs on.

You can use the **net use** and **change client** commands to map client devices not automatically mapped at logon. See your MetaFrame Presentation Server documentation for information about the **change client** command.

Turning off Client Device Mappings

On the server running MetaFrame Presentation Server, specify client device mapping options in the **Client Settings** dialog box in Citrix Connection Configuration.



Connection
☑ Connect Client Drives at Logon
☑ Connect Client Printers at Logon
☑ Default To Main Client Printer
(inherit user config) ☑

Other Options
□ By default, connect only the client's main printer.
Maximum Color Depth: Inherit User Config

Client Mapping Overrides
□ Disable Client Drive Mapping
□ Disable Windows Client Printer Mapping
□ Disable Client LPT Port Mapping
□ Disable Client COM Port Mapping
□ Disable Client Clipboard Mapping
□ Disable Client Audio Mapping

*This screen capture shows the Client Settings dialog box with Connection options disabled because **inherit user config** is selected. Client Mapping Overrides and Other Options are enabled.*

The **Connection** options control whether or not drives and printers are mapped to client drives and printers. If these options are cleared, the devices are still available but must be mapped to drive letters and port names manually.

Use the **Client Mapping Overrides** options to disable client device connections.

Option	Description
Connect Client Drives at Logon	If this option is selected, the client device drives are mapped automatically at logon.
Connect Client Printers at Logon	If this option is selected, the client device printers are mapped automatically at logon. This option applies only to Windows clients and maps only printers already configured in Print Manager on the client device.
Default to Main Client Printer	If this option is selected, the user's default client printer is configured as the default printer for the ICA session.
(inherit user config)	If this option is selected, the per-user settings in User Manager override these settings.

Mapping Client Drives

Client drive mapping allows drive letters on the server running MetaFrame Presentation Server to be redirected to drives that exist on the client device. For example, drive H in a Citrix user session can be mapped to drive C of the local device running the Client.

Client drive mapping is transparently built into the standard Citrix device redirection facilities. These mappings can be used by the File Manager or Explorer and your applications just like any other network mappings.

Important Client drive mapping is not supported when connecting to MetaFrame Server 1.0 for UNIX Operating Systems.

The server running MetaFrame Presentation Server can be configured during installation to automatically map client drives to a given set of drive letters. The default installation mapping maps drive letters assigned to client drives starting with V and works backwards, assigning a drive letter to each fixed disk and CD-ROM drive. (Floppy drives are assigned their existing drive letters.) This method yields the following drive mappings in a client session:

Client drive letter	Is accessed by the server running MetaFrame Presentation Server as:
A	A
B	B
C	V
D	U

The server running MetaFrame Presentation Server can be configured so that the server drive letters do not conflict with the client drive letters; in this case the server drive letters are changed to higher drive letters. For example, changing server drives C to M and D to N allows client devices to access their C and D drives directly. This method yields the following drive mappings in a client session:

Client drive letter	Is accessed by the server running MetaFrame Presentation Server as:
A	A
B	B
C	C
D	D

The drive letter used to replace the server drive C is defined during Setup. All other fixed disk and CD-ROM drive letters are replaced with sequential drive letters (for example; C->M, D->N, E->O). These drive letters must not conflict with any existing network drive mappings. If a network drive is mapped to the same drive letter as a server drive letter, the network drive mapping is not valid.

When a client device connects to a server running MetaFrame Presentation Server, Client mappings are reestablished unless automatic client device mapping is disabled. Automatic client device mapping can be configured for ICA connections and users. In the **Client Settings** dialog box, you can enable or disable automatic client device mapping for an ICA connection. The **User Configuration** dialog box in User Manager for Domains allows you to enable or disable automatic client device mapping for a user.

Mapping Client Printers

The MetaFrame Presentation Server Client supports auto-created printers. With auto-created printers, users find their local printers mapped to their sessions and ready for use as soon as they connect.

Published applications and ICA server connections configured to run a specified initial program offer users the same access to their local printers. When connected to published applications, users can print to local printers in the same way they would print to a local printer when using local applications.

Important For information about configuring Client printing on servers running MetaFrame Presentation Server for UNIX, see the *MetaFrame Presentation Server for UNIX Administrator’s Guide*.

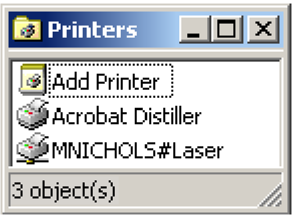
If the **Connect Client Printers at Logon** check box is selected in the terminal connection or user profile, the client printers are automatically connected when users log on and are deleted when they log off if the printers do not contain any print jobs. If print jobs are present, the printers (and the associated print jobs) are retained.

If you do not want a user's automatically created printers to be deleted when the user logs off, modify or delete the **Auto Created Client Printer** entry in the **Comment** field of a client printer's **Properties** dialog box. If you modify or delete this description, the printer is not deleted when the user logs off. Each time the user logs on, the printer that is already defined is used. If users change the Windows printer settings, they will not be set automatically. If users have custom print settings, you may not want to delete the automatically created printers.

If your user and terminal connection profile do not specify **Connect Client Printers at Logon**, you can use the Add Printer wizard to connect to a client printer. These printers are not deleted automatically when you log off.

To view mapped client printers

While connected to the server running MetaFrame Presentation Server, from the **Start** menu, choose **Settings > Printers**. The **Printers** window opens:



Add Printer
Acrobat Distiller
MNICHOLS#Laser

3 object(s)

This screen capture shows the Printers window with three objects available: Add Printer, Acrobat Distiller, and a printer named MNICHOLS#Laser.

The **Printers** screen displays the local printers mapped to the ICA session. The name of the printer takes the form *clientname#printername,* where *clientname* is the unique name given to the client device during Client Setup and *printername* is the Windows printer name. In this example ICA session, a client machine called "MNICHOLS" has access to its local printer named "Laser." This name cannot be changed and is used to locate the specific printer. Because the Windows printer name is used and not the port name (as with DOS Client printing), multiple printers can share a printer port without conflict.

Mapping Client COM Ports

Client COM port mapping allows devices attached to the COM ports of the client device to be used during ICA sessions on a server running MetaFrame Presentation Server. These mappings can be used like any other network mappings.

Note Client COM port mapping is not supported when connecting to MetaFrame Server 1.0 and 1.1 for UNIX Operating Systems.

To map a client COM port

1. Start the Client and log on to the server running MetaFrame Presentation Server.
2. At the command prompt, type

   net use com*x*: \\client\com*z*:

   where *x* is the number of the COM port on the server (ports 1 through 9 are available for mapping) and *z* is the number of the client COM port you want to map. Press **Enter**.
3. To confirm the operation, type

   net use

   at the command prompt. The list that appears contains mapped drives, LPT ports, and mapped COM ports.

   To use this COM port in a session on a server running MetaFrame Presentation Server, install your device to the mapped name. For example, if you map COM1 on the client to COM5 on the server, install your COM port device on COM5 during the session on the server. Use this mapped COM port as you would a COM port on the client device.

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

Mapping Client Audio

Client audio mapping enables applications executing on the server running MetaFrame Presentation Server to play sounds through a Windows-compatible sound device installed on the client device. You can set audio quality on a per-connection basis on the server running MetaFrame Presentation Server and users can set it on the client device. If the client and server audio quality settings are different, the lower setting is used.

Client audio mapping can cause excessive load on the servers running MetaFrame Presentation Server 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.

You control the amount of bandwidth client audio mapping uses from the Citrix Connection Configuration tool, which is available from the ICA Administrator Toolbar on the server running MetaFrame Presentation Server. The tool lets you choose among three different audio quality settings and you can disable client audio mapping altogether. See the *MetaFrame Presentation Server Administrator's Guide* for more information about client audio mapping.

Note Client sound support mapping is not supported when connecting to MetaFrame Server 1.0 and 1.1 for UNIX.

Configuring Multiple Monitors

If your client operating system with video hardware and drivers provides multiple monitor support with the Windows taskbar on the primary (left) monitor (Windows 98 and 2000 mode of multiple monitor support), there are restrictions in the level of support when using the client configured with seamless windows. Multiple monitors are fully supported when the client is configured in a non-seamless mode and set with the same color depth on all monitors in use.

Note Secondary windows sometimes appear in the primary monitor (uppermost, left).

System Hardware Requirements

To enable multiple monitor support, the system must have the following:

- Multiple PCI video boards, compatible with the Client on the appropriate Windows platform

  -or-
- A special multiple monitor video board, such as the Matrox G400, compatible with the Client on the appropriate Windows platform

The following hardware configurations were tested with multiple monitor support.

Important Citrix highly recommends that you test these configurations on your own hardware to ensure that they function properly for your specific machine configuration.

- On Windows 98, Matrox G400 is fully supported as a Windows 98/2000-style multiple monitor

- On Windows 2000, Matrox G400 works in a Windows NT 4.0/Windows 95-style multiple monitor
- Both Windows 98 and Windows 2000 support Matrox G200 PCI (multiple cards installed) as a Windows 98/2000-style multiple monitor
- Windows 98 supports a wide variety of PCI video boards, including many models from ATI and Cirrus Logic

Connecting to MetaFrame Presentation Servers for UNIX

Using the Window Manager

The window manager allows users to adjust the ICA session display for published resources on servers running MetaFrame Presentation Server for UNIX. With the window manager, users can minimize, resize, position, and close windows, as well as access full screen mode.

About Seamless Windows

In seamless window mode, published applications and desktops are not contained within an ICA session window. Each published application and desktop appears in its own resizable window, as if it is physically installed on the client device. Users can switch back and forth between published applications and the local desktop.

You can also display seamless windows in “full screen” mode, which places the published application in a full screen-sized desktop. This mode lets you access the ctxwm menu system.

Switching Between Seamless and Full Screen Modes

Press SHIFT+F2 to switch between seamless and full screen modes.

Minimizing, Resizing, Positioning, and Closing Windows

When users connect to published resources, window manager provides buttons to minimize, resize, position, and close windows.

Seamless windows are minimized as buttons on the taskbar. Other windows are minimized as desktop icons. Users open minimized windows by clicking the icons.

When the user closes the last application in a session, the session disconnects automatically after twenty seconds.

Using the Citrix Window Manager Menus

In remote desktop and seamless full screen windows, you can use the ctxwm menu system to log off, disconnect, and exit from published applications and connection sessions.

To access the ctxwm menu system

1. On a blank area of the remote desktop window, click and hold down the left mouse button. The ctxwm menu appears.
2. Drag the mouse pointer over **Shutdown** to display the shutdown options.

To choose an option from the ctxwm menu

Drag the pointer over the required option to select it. Release the mouse button to select the option.

To	Choose
Terminate the connection and all running applications	Logoff
Disconnect the session but leave the application running	Disconnect
Disconnect the session and terminate the application	Exit

Note Your server running MetaFrame Presentation Server may be configured to terminate any applications that are running if a session is disconnected.

Cutting and Pasting Graphics Using ctxgrab and ctxcapture

If you are connected to an application published on a server running MetaFrame Presentation Server for UNIX, use ctxgrab or ctxcapture to cut and paste graphics between the ICA session and the local desktop. These utilities are configured and deployed from the server running MetaFrame Presentation Server for UNIX.

Using ctxgrab

The ctxgrab utility is a simple tool you can use to cut and paste graphics from published applications to applications running on the local client device. This utility is available from the command prompt or, if you are using a published application, from the ctxwm window manager.

To access the ctxgrab utility from the window manager

1. In seamless mode, right-click the **ctxgrab** button in the top, left-hand corner of the screen to display a menu and choose the **grab** option.

   In full screen mode, left click to display the ctxwm menu and choose the **grab** option.

2. When ctxgrab starts, a dialog box appears.

To copy from an application in a Client window to a local application

1. From the **ctxgrab** dialog box, click **From screen**.
2. To:

   Select a window: move the cursor over the window you want to copy and click the middle mouse button.

   Select a region: hold down the left mouse button and drag the cursor to select the area you want to copy.

   Cancel the selection: click the right mouse button. While dragging, cancel the selection by clicking the right mouse button before releasing the first button.

3. Use the appropriate command in the local application to paste the object.

Using ctxcapture

The ctxcapture utility is a more fully-featured utility for cutting and pasting graphics between published applications and applications running on the local client device.

With ctxcapture you can:

- Grab dialog boxes or screen areas and copy them between an application in an ICA Client window and an application running on the local client device, including non-ICCCM-compliant applications
- Copy graphics between the Client and the X graphics manipulation utility xvf

If you are connected to a published desktop, ctxcapture is available from the command prompt. If you are connected to a published application and the MetaFrame administrator has made it available, you can access ctxcapture through the ctxwm window manager.

To access the ctxcapture utility from the window manager

1. Left click to display the **ctxwm** menu and choose the **screengrab** option.
2. When ctxcapture is started, a dialog box is displayed.

To copy from a local application to an application in a Client window

1. From the **ctxcapture** dialog box, click **From screen**.
2. **To select a window**: move the cursor over the window you want to copy and click the middle mouse button.

   To select a region: hold down the left mouse button and drag the cursor to select the area you want to copy.

   To cancel the selection: click the right mouse button. While dragging, cancel the selection by clicking the right mouse button before releasing the first button.
3. From the **ctxcapture** dialog box, click **To ICA**. The **xcapture** button changes color to indicate that it is processing the information.
4. When the transfer is complete, use the appropriate command in the published application window to paste the information.

To copy from an application in a Client window to a local application

1. From the application in the Client window, copy the graphic.
2. From the **ctxcapture** dialog box, click **From ICA**.
3. When the transfer is complete, use the appropriate command in the local application to paste the information.

To copy from xv to an application in a Client window or local application

1. From xv, copy the graphic.
2. From the **ctxcapture** dialog box, click **From xv** and **To ICA**.
3. When the transfer is complete, use the appropriate command in the Client window to paste the information.

To copy from an application in a Client window to xv

1. From the application in the Client window, copy the graphic.
2. From the **ctxcapture** dialog box, click **From ICA** and **To xv**.
3. When the transfer is complete, use the paste command in xv.

Go to Document Center

Securing Client Communication

This chapter discusses measures you can take to secure the communication between your MetaFrame Presentation Server farm and the Clients. The following topics are covered:

- Connecting Through a Proxy Server
- Using the Clients with the Secure Gateway for MetaFrame Presentation Server or SSL Relay
- Connecting to a Server Through a Firewall

Integrating the Clients with Your Security Solutions

You can integrate the Clients with a range of security technologies, including proxy servers, firewalls, and SSL/TLS-based systems. This section describes:

- Connecting through a SOCKS proxy server or Secure proxy server (also known as *security proxy server*, HTTPS proxy server, or SSL tunneling proxy server)
- Integrating the Clients with the Secure Gateway for MetaFrame Presentation Server or SSL Relay solutions with Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols
- Connecting to a server through a firewall

Connecting Through a Proxy Server

Proxy servers are used to limit access into and out of your network, and to handle connections between Clients and servers running MetaFrame Presentation Server. The Clients support SOCKS and secure proxy protocols.

Program Neighborhood Agent and the Web Client

When communicating with the server farm, the Program Neighborhood Agent and the Web Client use proxy server settings that are configured remotely on the server running the Web Interface. See the *Web Interface Administrator's Guide* for information about configuring proxy server settings for these Clients.

In communicating with the Web server, the Program Neighborhood Agent and the Web Client use the proxy server settings that are configured through the Internet settings of the default Web browser on the client device. You must configure the Internet settings of the default Web browser on the client device accordingly.

Program Neighborhood

Program Neighborhood uses proxy server settings you configure locally from the Client’s toolbar. You can configure proxy server settings in two ways:

- Enable auto-client proxy detection
- Manually specify the details of your proxy server

Enabling Auto-Client Proxy Detection

If you are deploying the Client in an organization with multiple proxy servers, consider using auto-client proxy detection. Auto-client proxy detection communicates with the local Web browser to discover the details of the proxy server. It is also useful if you cannot determine which proxy server will be used when you configure the client. Auto-client proxy detection requires Internet Explorer 5.0 or later, or Netscape for Windows 4.78, 6.2, or later.

To enable auto-client proxy detection

1. Start Program Neighborhood.
2. If you are configuring an application set:

   Right-click the application set you want to configure and select **Application Set Settings**. A **Settings** dialog box for the application set appears.

   If you are configuring an *existing* custom ICA connection:

   Right-click the custom ICA connection you want to configure and select **Properties**. The **Properties** dialog box for the custom connection appears.

   If you are configuring *all future* custom ICA connections:

   Right-click in a blank area of the **Custom ICA Connections** window and select **Custom Connections Settings**. The **Custom ICA Connections** dialog box appears.
3. On the **Connection** tab, click **Firewalls**.
4. Select **Use Web browser proxy settings**.
5. Click **OK** twice.

Manually Specifying the Details of Your Proxy Server

Note If you are configuring the proxy manually, confirm these details with your security administrator. ICA connections cannot be made if these details are incorrect.

To manually specify the details of your proxy server

1. Start Program Neighborhood.
2. If you are configuring an application set:

   Right-click the application set you want to configure and select **Application Set Settings**. A **Settings** dialog box for the application set appears.

   If you are configuring an *existing* custom ICA connection:

   Right-click the custom ICA connection you want to configure and select **Properties**. The **Properties** dialog box for the custom connection appears.

   If you are configuring *all future* custom ICA connections:

   Right-click in a blank area of the **Custom ICA Connections** window and select **Custom Connections Settings**. The **Custom ICA Connections** dialog box appears.
3. On the **Connection** tab, click **Firewalls**.
4. Select the proxy protocol type (**SOCKS** or **Secure (HTTPS)**).

5. Enter the proxy address and the port number for the proxy server.
   - The default port for SOCKS is 1080
   - the default port for secure proxy is 8080
6. Click **OK** twice.

Configuring the User Name and Password

Some proxy servers require authentication, prompting you for a user name and password when you enumerate resources or open an ICA connection. You can avoid these prompts by configuring the Client to pass the credentials without user intervention. You can create settings that:

- Apply to one or several existing custom ICA connections

  -or-

- Act as the default for all future custom ICA connections to be created using the Add ICA Connection wizard

To create a setting for one or several existing custom ICA connections

1. Exit Program Neighborhood if it is running. Make sure all Program Neighborhood components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the **[*ServerLocation*]** section, where *ServerLocation* is the name of the connection you want to configure.

4. Locate the **DoNotUseDefaultCSL** property of that **[*ServerLocation*]** section.

   If the value of **DoNotUseDefaultCSL** is On, perform the following steps:

   Add the following lines to that **[*ServerLocation*]** section:

   ProxyUsername=<*user name*>

   ProxyPassword=<*password*>

   where *user name* is the user name recognized by the SOCKS server and *password* is the password associated with the user name recognized by the proxy server.

   If the value of **DoNotUseDefaultCSL** is Off, or if the parameter is not present, perform the following steps:

   Add the following lines to the **[WFClient]** section:

   ProxyUsername=<*user name*>

   ProxyPassword=<*password*>

   where *user name* is the user name recognized by the SOCKS server and *password* is the password associated with the user name recognized by the proxy server.

5. Repeat Steps 3 and 4 for any additional connections if applicable.
6. Save your changes.

Note Users can override the default setting from within a particular custom ICA connection's **Properties** dialog box.

To create a default for all future custom ICA connections

1. Exit Program Neighborhood if it is running and make sure all Program Neighborhood components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the section named **[WFClient]**.
4. Add the following lines to the list of parameters and values in the **[WFClient]** section:

   ProxyUsername=<*user name*>

   ProxyPassword=<*password*>

   where *user name* is the user name recognized by the SOCKS server and *password* is the password associated with the user name recognized by the proxy server.

5. Save your changes.

Note Users can override the default setting from within a particular custom ICA connection's **Properties** dialog box.

Using the Clients with the Secure Gateway for MetaFrame Presentation Server or SSL Relay

You can integrate the Clients with the Secure Gateway or SSL Relay service. The clients support both SSL and TLS protocols.

- SSL provides strong encryption to increase the privacy of your ICA connections and certificate-based server authentication to ensure the server you are connecting to is a genuine server.
- TLS (Transport Layer Security) is the latest, standardized version of the SSL protocol. The Internet Engineering Taskforce (IETF) renamed it TLS when it took over responsibility for the development of SSL as an open standard. TLS secures data communications by providing server authentication, encryption of the data stream, and message integrity checks. Because there are only minor technical differences between SSL Version 3.0 and TLS Version 1.0, the certificates you use for SSL in your MetaFrame installation will also work with TLS. Some organizations, including US government organizations, require the use of TLS to secure data communications. These organizations may also require the use of validated cryptography, such as FIPS 140. FIPS 140 (Federal Information Processing Standard) is a standard for cryptography.

The Secure Gateway

You can use the Secure Gateway in either *Normal* mode or *Relay* mode. No Client configuration is required if you are using the Secure Gateway in Normal mode and users are connecting through the Web Interface.

If you are using the Secure Gateway in Relay mode, the Secure Gateway server functions as a proxy and you must configure the Client to use:

- The fully qualified domain name (FQDN) of the Secure Gateway server
- The port number of the Secure Gateway server

Note Relay mode is not supported by the Secure Gateway, Version 2.0.

Program Neighborhood Agent and Web Client

The Program Neighborhood Agent and the Web Client use settings that are configured remotely on the server running the Web Interface to connect to servers running the Secure Gateway. See the *Web Interface Administrator's Guide* for information about configuring proxy server settings for these Clients.

Program Neighborhood

To configure the details of your Secure Gateway server

1. Make sure the client device meets all system requirements outlined in this guide.
2. Start Program Neighborhood.
3. If you are configuring an application set:

   Right-click the application set you want to configure and select **Application Set Settings**. A configuration dialog box for the application set appears.

   If you are configuring an *existing* custom ICA connection:

   Right-click the custom ICA connection you want to configure and select **Properties**. The **Properties** dialog box for the custom connection appears.

   If you are configuring *all future* custom ICA connections:

   Right-click in a blank area of the **Custom ICA Connections** window and select **Custom Connections Settings**. The **Custom ICA Connections** dialog box appears.
4. If you are configuring an application set or an *existing* custom ICA connection:

   From the **Network Protocol** menu, select **SSL/TLS+HTTPS**.

   If you are configuring *all future* custom ICA connections:

   From the **Network Protocol** menu, select **HTTP/HTTPS**.
5. On the **Connection** tab, click **Firewalls**.

6. Enter the fully qualified domain name (FQDN) of the Secure Gateway server in the **Secure gateway address** box.

Important The fully qualified domain name (FQDN) must list, in sequence, the following three components:

- Host name
- Intermediate domain
- Top-level domain

For example: *my_computer.my_company.com* is an FQDN, because it lists, in sequence, a host name (my_computer), an intermediate domain (my_company), and a top-level domain (com). The combination of intermediate and top-level domain (my_company.com) is generally referred to as the domain name.

7. Enter the port number in the **Port** box.
8. Click **OK** twice.

Citrix SSL Relay

By default, Citrix SSL Relay uses TCP port 443 on the server running MetaFrame Presentation Server for SSL/TLS-secured communication. When the SSL Relay receives an SSL/TLS connection, it decrypts the data before redirecting it to the server running MetaFrame Presentation Server, or, if the user selects SSL/TLS+HTTPS browsing, to the Citrix XML Service.

You can use Citrix SSL Relay to secure communications:

- Between an SSL/TLS-enabled Client and a server running MetaFrame Presentation Server. Connections using SSL/TLS encryption are marked with a padlock icon in Citrix Connection Center.
- With a server running the Web Interface, between the server running MetaFrame Presentation Server and the Web server.

For information about configuring and using SSL Relay to secure your MetaFrame Presentation Server installation, see the *MetaFrame Presentation Server Administrator's Guide*. For information about configuring the server running the Web Interface to use SSL/TLS encryption, see the *Web Interface Administrator's Guide*.

System Requirements

In addition to the system requirements listed for each Client in its respective chapter, you also must ensure that:

- The client device supports 128-bit encryption
- The client device has a root certificate installed that can verify the signature of the Certificate Authority on the server certificate
- The Client is aware of the TCP listening port number used by the SSL Relay service in the server farm

Verifying Cipher Strength/128-bit Encryption

If you have Internet Explorer installed on your system, you can determine the encryption level of your system as follows:

1. Start Internet Explorer.
2. From the **Help** menu, click **About Internet Explorer**.
3. Check the Cipher Strength value. If it is less than 128-bit, you need to obtain and install a high encryption upgrade from the Microsoft Web site. Go to http://www.microsoft.com/ and search for “128-bit” or “strong encryption.”
4. Download and install the upgrade.

If you do not have Internet Explorer installed, or if you are not certain about the encryption level of your system, visit Microsoft’s Web site at http://www.microsoft.com/ to install a service pack that provides 128-bit encryption.

Note The Clients support certificate key lengths of up to 4096 bits. Ensure that the bit lengths of your Certificate Authority root and intermediate certificates, and those of your server certificates, do not exceed the bit length your Clients support, or connection may fail.

About Root Certificates

See “Installing Root Certificates on the Clients” on page 89 for information about root certificates.

Important All secure systems need to be maintained. Ensure that you apply any service packs or upgrades that Microsoft recommends.

Using Citrix SSL Relay with Non-Standard TCP Ports

By default, Citrix SSL Relay uses TCP port 443 on the server running MetaFrame Presentation Server for SSL/TLS-secured communication. If you configure SSL Relay to listen on a port other than 443, you must make the Client aware of the non-standard listening port number.

In Program Neighborhood, users can change the port number in the Firewall Settings dialog box. For step-by-step instructions, see the Program Neighborhood online help.

To apply a different listening port number for all connections

1. Make sure all Client components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the **[WFClient]** section.

   Set the value of the SSLProxyHost parameters as follows:

   SSLProxyHost=*:<*SSL relay port number*>,

   where <*SSL relay port number*> is the number of the listening port.
4. Save and close the file.

To apply a different listening port number to particular connections only

1. Make sure all Client components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the particular **[Connection_Section]**.

   Set the value of the SSLProxyHost parameters as follows:

   SSLProxyHost=*:<*SSL relay port number*>,

   where <*SSL relay port number*> is the number of the listening port.
4. Repeat Step 3 for all connection sections for which you want to specify a different listening port number.
5. Save and close the file.

Configuring and Enabling Clients for SSL and TLS

SSL and TLS are configured in the same way, use the same certificates, and are enabled simultaneously.

When SSL and TLS are enabled, each time you initiate a connection the client tries to use TLS first, then tries SSL. If it cannot connect with SSL, the connection fails and an error message appears.

Forcing TLS Connections for all Clients

To force the Clients (including the Web Client) to connect with TLS, you must specify TLS on the Secure Gateway server or SSL Relay service. See the *Secure Gateway Administrator's Guide* or SSL Relay service documentation for more information.

To configure Program Neighborhood to use SSL/TLS

1. Make sure the client device meets all system requirements outlined in this guide.
2. Open Program Neighborhood.
3. If you are configuring an application set to use SSL/TLS:

   Right-click the application set you want to configure and select **Application Set Settings**. A **Settings** dialog box for the application set appears.

   If you are configuring an *existing* custom ICA connection to use SSL/TLS:

   Right-click the custom ICA connection you want to configure and select **Properties**. The **Properties** dialog box for the custom connection appears.

   If you are configuring *all future* custom ICA connection to use SSL/TLS:

   Right-click in a blank area of the **Custom ICA Connections** window and select **Custom Connections Settings**. The **Custom ICA Connections** dialog box appears.

4. If you are configuring an application set or an *existing* custom ICA connection:

   From the **Network Protocol** menu, select **SSL/TLS+HTTPS**.

   If you are configuring *all future* custom ICA connections:

   From the **Network Protocol** menu, select **HTTP/HTTPS**.

5. Add the fully qualified domain name of the SSL/TLS-enabled servers running MetaFrame Presentation Server to the Address List.
6. Click **OK**.

To configure the Program Neighborhood Agent to use SSL/TLS

1. Make sure the client device meets all system requirements outlined in this guide.
2. To use SSL/TLS to encrypt application enumeration and launch data passed between the Program Neighborhood Agent and the server running the Web Interface, configure the appropriate settings in the Program Neighborhood Agent Console on the Web server. The configuration file must include the machine name of the server running MetaFrame Presentation Server that is hosting the SSL certificate.

3. To use secure HTTP (HTTPS) to encrypt the configuration information passed between the Program Neighborhood Agent and the server running the Web Interface, enter the server URL in the format https://<*servername*> on the **Server** tab of the Program Neighborhood Agent **Properties** dialog box.

To configure the Appsrv.ini file to use TLS

1. Exit the Client if it is running. Make sure all Client components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the section named **[WFClient]**.

   Set the values of these two parameters as follows:

   SSLCIPHERS={GOV | All}

   SECURECHANNELPROTOCOL={TLS | Detect}. Set the value to **TLS**, or **Detect** to enable TLS. If **Detect** is selected, the Client tries to connect using TLS encryption. If a connection using TLS fails, the client tries to connect using SSL.
4. Save your changes.

Meeting FIPS 140 Security Requirements

To meet FIPS 140 security requirements, you must include the following parameters in the Template.ica file on the server running the Web Interface or in the user-level Appsrv.ini file of the local client device. See the *Web Interface Administrator's Guide* for additional information about the Template.ica file.

To configure the Appsrv.ini file to meet FIPS 140 security requirements

1. Exit the Client if it is running. Make sure all Client components, including the Connection Center, are closed.
2. Open the individual's user-level Appsrv.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.
3. Locate the section named **[WFClient]**.
4. Set the values of these three parameters as follows:

   SSLENABLE=On

   SSLCIPHER=GOV

   SECURECHANNELPROTOCOL=TLS
5. Save your changes.

Installing Root Certificates on the Clients

To use SSL/TLS to secure communications between SSL/TLS-enabled Clients and the server farm, you need a root certificate on the client device that can verify the signature of the Certificate Authority on the server certificate.

The Clients support the Certificate Authorities that are supported by the Windows operating system. The root certificates for these Certificate Authorities are installed with Windows and managed using Windows utilities. They are the same root certificates that are used by Microsoft Internet Explorer.

If you use your own Certificate Authority, you must obtain a root certificate from that Certificate Authority and install it on each client device. This root certificate is then used and trusted by both Microsoft Internet Explorer and the Client.

Depending on your organization's policies and procedures, you may want to install the root certificate on each client device instead of directing users to install it. If you are using Windows 2000 with Active Directory on all client devices, you can deploy and install root certificates using Windows 2000 Group Policy. See your Microsoft Windows 2000 documentation for more information.

Alternatively, you may be able to install the root certificate using other administration or deployment methods, such as:

- Using the Microsoft Internet Explorer Administration Kit (IEAK) Configuration Wizard and Profile Manager
- Using third-party deployment tools

Make sure that the certificates installed by your Windows operating system meet the security requirements for your organization, or use the certificates issued by your organization's Certificate Authority.

Note The following steps assume that your organization has a procedure in place for users to check the root certificate as they install it. It is important to verify the authenticity of a root certificate before installing it.

To install a root certificate on a client device

1. Double-click the root certificate file. The root certificate file has the extension .cer, .crt, or .der.
2. Verify that you are installing the correct root certificate.
3. Click **Install Certificate**. The Certificate Import wizard starts.
4. Click **Next**.
5. Choose the **Place all certificates in the following store** option and then click **Browse**.

6. On the **Select Certificate Store** screen, select **Show physical stores**.
7. Expand the Trusted Root Certification Authorities store and then select **Local Computer**. Click **OK**.
8. Click **Next** and then click **Finish**. The root certificate is installed in the store you selected.

Securing the Program Neighborhood Agent with SSL/TLS

Make sure the client device meets all system requirements outlined in this guide.

To use SSL/TLS encryption for all communications between the Program Neighborhood Agent, MetaFrame Presentation Server, and the server running the Web Interface, the following configuration is necessary.

What You Need to do on the Server Running the Web Interface

To use SSL/TLS to secure the communications between the Program Neighborhood Agent and the Web server

1. In the Program Neighborhood Agent Console, select **Server Settings** from the **Configuration settings** menu.
2. Select **Use SSL/TLS for communications between clients and the Web server.**
3. Save your changes.

Selecting SSL/TLS changes all URLs to use HTTPS protocol.

What You Need to do on MetaFrame Presentation Server

To use SSL/TLS to secure the communications between the Program Neighborhood Agent and MetaFrame Presentation Server

Make sure you select the **Enable SSL** option on the **ICA Client Options** tab of the **Application Properties** dialog box in the Management Console for every application you want to secure. For more information, see the *MetaFrame Presentation Server Administrator's Guide*.

To use the SSL Relay to secure communications between MetaFrame Presentation Server and the server running the Web Interface

You must specify the machine name of the server hosting the SSL certificate in your configuration file. See the *Web Interface Administrator's Guide* for more information about using SSL/TLS to secure communications between MetaFrame Presentation Server and the Web server.

What You Need to do on the Client Device

This section assumes that a valid root certificate is installed on the client device. See “Installing Root Certificates on the Clients” on page 89 for more information.

To use SSL/TLS to secure the communications between the Program Neighborhood Agent and the server running the Web Interface

1. In the Windows system tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.
2. The **Server** tab displays the currently configured URL. Click **Change** and enter the server URL in the dialog box that appears. Enter the URL in the format https://<*servername*> to encrypt the configuration data using SSL/TLS.
3. Click **Update** to apply the change and return to the **Server** tab, or click **Cancel** to cancel the operation.
4. Click **OK** to close the **Properties** dialog box.
5. Enable SSL/TLS in the client browser. For more information about enabling SSL/TLS in the client browser, see the online Help for the browser.

Enabling Smart Card Logon

This section assumes that smart card support is enabled on the server running MetaFrame Presentation Server, and that the client device is properly set up and configured with third party smart card hardware and software. Refer to the documentation that came with your smart card equipment for instructions about deploying smart cards within your network.

The smart card removal policy set on MetaFrame Presentation Server determines what happens if you remove the smart card from the reader during an ICA session. The smart card removal policy is configured through and handled by the Windows operating system.

Enabling Smart Card Logon with Pass-Through Authentication

Pass-through authentication requires a smart card inserted in the smart card reader at logon time only. With this logon mode selected, the Client prompts the user for a smart card PIN (Personal Identification Number) when it starts up. Pass-through authentication then caches the PIN and passes it to the server every time the user requests a published resource. The user does not have to subsequently reenter a PIN to access published resources or have the smart card continuously inserted.

If authentication based on the cached PIN fails or if a published resource itself requires user authentication, the user continues to be prompted for a PIN.

To enable smart card logon with pass-through authentication

1. From the Program Neighborhood Agent Console, select **Logon Method** from the **Configuration settings** menu.
2. Click **Smart card pass-through authentication** to select the option.
3. Save your changes.

For more information about pass-through authentication, see "SSPI/Kerberos Security for Pass-Through Authentication" on page 50.

Enabling Smart Card Logon without Pass-Through Authentication

Disabling pass-through authentication requires a smart card to be present in the smart card reader whenever the user accesses a server. With pass-through disabled, the Client prompts the user for a smart card PIN when it starts up and every time the user requests a published resource.

To enable smart card logon without pass-through authentication

1. From the Program Neighborhood Agent Console, select **Logon Method** from the **Configuration settings** menu.
2. Click **Smart card logon** to select the option.
3. Verify that **Pass-through authentication** is not selected.
4. Save your changes.

Enabling NDS Logon Support

See "Novell Directory Services Support" on page 59 for additional information about enabling NDS support.

1. From the Program Neighborhood Agent Console, select **Logon Method** from the **Configuration settings** menu.
2. Click **Use NDS credentials for Prompt user and Pass-through authentication** to select the option.
3. Enter the default tree name.
4. Save your changes.

Connecting to a Server Through a Firewall

Network firewalls can allow or block packets based on the destination address and port. If you are using the Clients through a network firewall that maps the server’s internal network IP address to an external Internet address, do the following:

1. Open Program Neighborhood.
2. If you are configuring an application set:

   > Right-click the application set you want to configure and select **Application Set Settings**. A configuration dialog box for the application set appears.

   If you are configuring a custom ICA connection:

   > Right-click the custom ICA connection you want to configure and select **Custom Connections Settings**. The **Custom ICA Connections** dialog box appears.
3. Click **Add**. The **Add Server Location Address** dialog box appears.
4. Enter the external Internet address of the server running MetaFrame Presentation Server.
5. Click **OK**. The external Internet address you added appears in the Address List.
6. Click **Firewalls**.
7. Select **Use alternate address for firewall connection**.
8. Click **OK** twice.

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

Updating the Clients

This chapter explains how to deploy Client updates across your network. The following topics are covered:

- The Client Update Process
- Using the ICA Client Update Configuration Utility
- Specifying a Default Client Update Database
- Configuring Default Client Update Options
- Adding Clients to the Client Update Database
- Removing a Client From the Client Update Database
- Changing the Properties of the Client

About the Client Auto Update Feature

The Clients for 32-bit Windows are available from a single Microsoft Windows Installer (.msi) package referred to as the MetaFrame Presentation Server Client Packager. If your network is based on Windows 2000 or later, you can take advantage of Microsoft Systems Management Server or Active Directory to deploy updated versions of these clients using the MetaFrame Presentation Server Client Packager. For more information about the MetaFrame Presentation Server Client Packager, see “Using Microsoft Systems Management Server or Active Directory Services” on page 20.

To deploy updates to the Web Client, or if your network does not have Systems Management Server or Active Directory Services available, you can use the Client Auto Update feature to deploy and install Client updates using self-extracting executable (.exe) files.

The Client Auto Update feature is a convenient network management tool. It monitors client version numbers as users log on to a server running MetaFrame Presentation Server and updates client installations network-wide as appropriate to a version you specify.

Typically, you use Client Auto Update to deploy the latest release of the Clients on your network. You can also use this feature to revert to a previous version of a Client. Visit the Download page of the Citrix Web site (http://www.citrix.com) frequently for the latest releases and documentation of the Clients.

As new versions of Clients become available, you add them to the client update database. When a Client logs on to a server running MetaFrame Presentation Server, the server queries the client to detect its version number. If the version matches the one in the client update database, the logon continues. Otherwise, the user is informed that an update to the client is available for download. The Client is then updated according to the options you set in the database.

Important You cannot automatically update previous versions of the Client installed with Windows Installer (.msi) packages. You must redeploy a Client installer package when a new version of the Client is released.

Client Auto Update works with all network protocols supported by ICA (TCP/IP, IPX, NetBIOS, and asynchronous). Client Auto Update also:

- Automatically detects Client versions
- Copies new files over any ICA connection without user intervention
- Provides administrative control of update options for each Client
- Updates Clients from a single database on a network share point
- Safely restores older Client versions when needed

The Client Update Process

You identify Clients by platform and by product and model number. The version number is assigned when new Clients are released.

The process of updating Clients with new versions uses the standard ICA protocol:

- By default, MetaFrame Presentation Server informs the user of newly available client updates and asks to perform the update. Optionally, you can specify that the update be performed without informing the user and without allowing the user to cancel the update.
- By default, users can choose between waiting for the download to complete or downloading the files in the background while they continue to work. Users connecting to the server farm with a modem obtain better performance waiting for the update process to complete. Optionally, you can force the client update to complete before allowing the user to continue.

- During the update, new client files are copied to the user’s computer. Optionally, you can force the user to disconnect and complete the update before continuing the session. To continue working, the user must log onto the server farm again.
- When the user disconnects from the server and closes all client programs, the Client update process finishes.
- As a safeguard, the existing Client files are saved to a folder named Backup in the Citrix\ICA Client subdirectory of the Program Files folder on the user’s local drive.

Configuring the Client Update Database

You can configure a client update database on each server in a server farm or configure one database to update the Clients for multiple servers running MetaFrame Presentation Server.

The client update database contains several Clients. As Citrix releases new versions of the Clients, you add them to the client update database.

Using the ICA Client Update Configuration Utility

Use the ICA Client Update Configuration utility to manage the client update database. From this utility, you can:

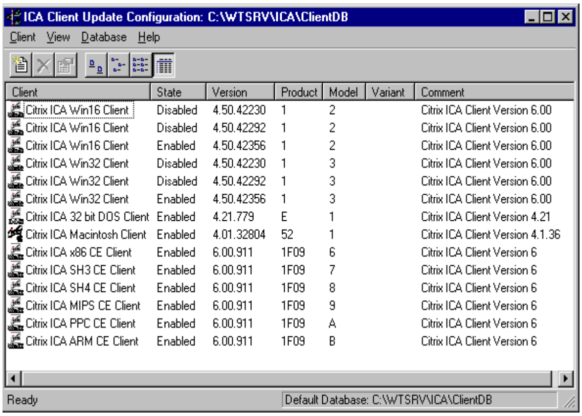
- Create a new update database
- Specify a default update database
- Configure the properties of the database
- Configure client update options
- Add new Clients to the database
- Remove outdated or unnecessary Clients
- Change the properties of a Client in the database

The following sections give an overview of the Client Update Configuration utility. For details, see the utility’s online help.

To start the ICA Client Update Configuration utility

- On a server running MetaFrame Presentation Server, select **ICA Client Update Configuration** from the **Citrix** program group in the **Start** menu.
- From a MetaFrame XP server: From the **Start** menu, choose **Programs > Citrix > MetaFrame XP > ICA Client Update Configuration**.
- From a MetaFrame 1.8 server: From the **Start** menu, choose **Programs** > **MetaFrame Tools > ICA Client Update Configuration.**

In the Client Update Configuration window, the status bar shows the location of the current update database, that MetaFrame Presentation Server uses to update Clients. The window shows the Clients in the database.


ICA Client Update Configuration: C:\WTSRV\ICA\ClientDB
Client View Database Help

Client	State	Version	Product	Model	Variant	Comment
Citrix ICA Win16 Client	Disabled	4.50.42230	1	2		Citrix ICA Client Version 6.00
Citrix ICA Win16 Client	Disabled	4.50.42292	1	2		Citrix ICA Client Version 6.00
Citrix ICA Win16 Client	Enabled	4.50.42356	1	2		Citrix ICA Client Version 6.00
Citrix ICA Win32 Client	Disabled	4.50.42230	1	3		Citrix ICA Client Version 6.00
Citrix ICA Win32 Client	Disabled	4.50.42292	1	3		Citrix ICA Client Version 6.00
Citrix ICA Win32 Client	Enabled	4.50.42356	1	3		Citrix ICA Client Version 6.00
Citrix ICA 32 bit DOS Client	Enabled	4.21.779	E	1		Citrix ICA Client Version 4.21
Citrix ICA Macintosh Client	Enabled	4.01.32804	52	1		Citrix ICA Client Version 4.1.36
Citrix ICA x86 CE Client	Enabled	6.00.911	1F09	6		Citrix ICA Client Version 6
Citrix ICA SH3 CE Client	Enabled	6.00.911	1F09	7		Citrix ICA Client Version 6
Citrix ICA SH4 CE Client	Enabled	6.00.911	1F09	8		Citrix ICA Client Version 6
Citrix ICA MIPS CE Client	Enabled	6.00.911	1F09	9		Citrix ICA Client Version 6
Citrix ICA PPC CE Client	Enabled	6.00.911	1F09	A		Citrix ICA Client Version 6
Citrix ICA ARM CE Client	Enabled	6.00.911	1F09	B		Citrix ICA Client Version 6

Ready

Default Database: C:\WTSRV\ICA\ClientDB

This screen capture shows the Details view of the ICA Client Update Configuration window, which lists installed Clients, their states, versions, product codes, model numbers, variants, and comments.

Note Citrix MetaFrame Presentation Server for UNIX Operating Systems does not use the Client Update Database. To use the client update database, you must have MetaFrame 1.8, MetaFrame XP, or MetaFrame Presentation Server for Windows running on server in your server farm.

Creating a New Client Update Database

The ICA Client Distribution wizard creates the Client Update Database in the directory %Program Files%\Citrix\ICA\ClientDB. You can create a new update database in any location on a server drive or on a network share point.

To create a new update database

1. From the **Database** menu, choose **New**.
2. In the **Path for the new Client Update Database** dialog box, type the path for the new update database and click **Save**.

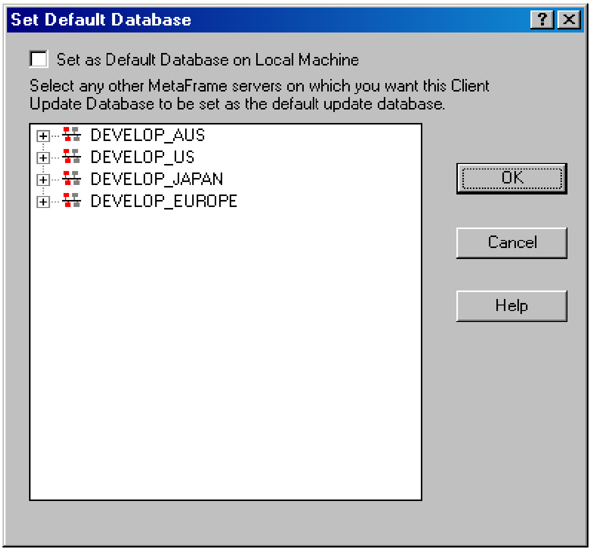
The utility creates a new update database in the specified location and opens the new database.

Specifying a Default Client Update Database

You can configure one client update database to be used by multiple servers running MetaFrame Presentation Server. If the client update database is on a shared network drive, use the ICA Client Update Configuration utility to configure your servers to use the same shared database.

To set the default database for MetaFrame Presentation Servers

1. From the **Database** menu, choose **Open**.
2. In the **Open Existing Database** dialog box, type the path to the default database and click **Open**.
3. From the **Database** menu, choose **Set Default**.



Set Default Database

☐ Set as Default Database on Local Machine

Select any other MetaFrame servers on which you want this Client Update Database to be set as the default update database.

⊞ DEVELOP_AUS
⊞ DEVELOP_US
⊞ DEVELOP_JAPAN
⊞ DEVELOP_EUROPE

This screen capture shows the Set Default Database dialog box, which provides a tree view of Citrix servers, one of which you can choose to have the Client Update Database set as its default.

4. Select **Set as Default Database on Local Machine** to make the currently opened database the default database. You can also set other servers to use the currently open database as the default database.
5. Expand the node for a domain name to view the servers in that domain. Click a server to set its default database to the currently open database. You can select multiple servers by holding down the CTRL key and clicking each server.
6. Click **OK**.

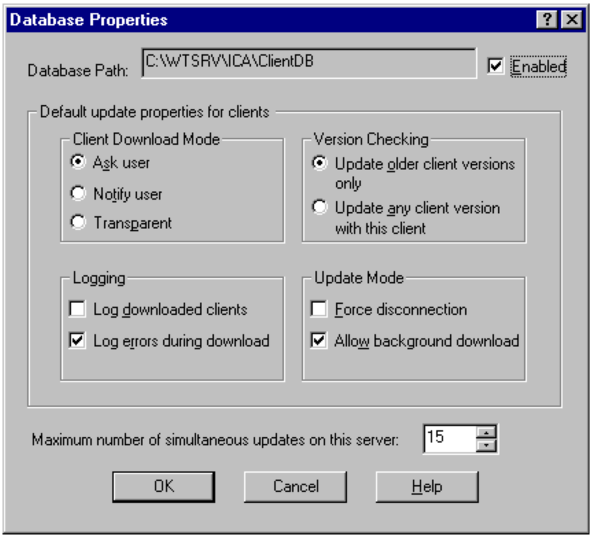
Configuring Default Client Update Options

Use the **Database Properties** dialog box to configure overall database-wide settings for the current client update database.

To configure database properties

1. From the **Database** menu, choose **Properties**.

   The **Database Path** box displays the path and file name of the database you are configuring.



Database Properties

Database Path: C:\WTSRV\ICA\ClientDB ☑ Enabled

Default update properties for clients

Client Download Mode
● Ask user
○ Notify user
○ Transparent

Version Checking
● Update older client versions only
○ Update any client version with this client

Logging
□ Log downloaded clients
☑ Log errors during download

Update Mode
□ Force disconnection
☑ Allow background download

Maximum number of simultaneous updates on this server: 15

This screen capture shows the Database Properties dialog box, which displays the database path being configured and provides options for setting default update properties for clients and for setting the maximum number of simultaneous updates allowed on the server.

2. For this database to perform Client updates, select **Enabled**.

Tip If the Clients do not need to be updated, disable the database to shorten logon time for your users.

The options in the **Default update properties for clients** section specify the default behavior for the Clients added to the database. You can also set properties for individual Clients (as described later in this chapter). Individual Client properties override the database properties.

- Under **Client Download Mode**, select **Ask user** to give the user the choice to accept or postpone the update process. Select **Notify user** to notify the user of the update and require the client update. Select **Transparent** to update the user's Client without notifying or asking the user.
- Under **Version Checking**, select **Update older client versions only** to update only client versions that are older than the new client. Select **Update any client version with this client** to update both earlier and later versions of the client to this version; choose this option to force an older client to replace a newer client.
- Under **Logging**, select **Log downloaded clients** to write an event to the event log when a client is updated. By default, errors that occur during a client update are written to the event log. Clear the **Log errors during download** check box to turn this option off.
- Under **Update Mode**, select the **Force disconnection** option to require users to disconnect and complete the update process after downloading the new client. The **Allow background download** option is selected by default to allow users to download new client files in the background while they continue to work. Clear this check box to force users to wait for all client files to download before continuing.

3. Specify the number of simultaneous updates on the server. When the specified number of updates is reached, new client connections are not updated. When the number of client updates is below the specified number, new client connections are updated.
4. Click **OK** when you finish configuring the database settings.

Adding Clients to the Client Update Database

When you want to deploy a newer version of the client software, add it to the Client Update Database. You can download the latest client software from the Citrix Web site at http://www.citrix.com/download.

To add client software to the Client Update Database

1. From the **Client** menu, click **New** to display the **Description** screen.
2. In the **Client Installation File** box, browse to or enter the path to the client installation file Update.ini. If you ran the ICA Client Distribution wizard, you can find the Update.ini file in the location %Program_Files%\Citrix\ICA\ClientDB. You can also find the Update.ini file on the Components CD.
3. The client name, product number, model number, and version number are displayed. The **Comment** text box displays a description of the new client. You can modify this comment. Click **Next** to continue.
4. The **Update Options** dialog box appears. The options in this dialog box specify how the client update process occurs for this client. The database-wide update options are displayed. You can specify different behavior for individual clients. For definitions of the options in this dialog box, see the online help for this dialog box. Click **Next** when you finish configuring the client update options.
5. The **Event Logging** dialog box appears.

   The database-wide logging options are displayed. You can specify different behavior for individual clients. Select **Log Downloaded Clients** to write an event to the event log when this client is updated. By default, errors that occur during a client update are written to the event log. Clear the **Log Errors During Download** check box to turn this option off. Click **Next** to continue.
6. The **Enable Client** dialog box appears.

   The Client Update Database can contain multiple versions of client software with the same product and model numbers. For example, when Citrix releases a new version of the MetaFrame Presentation Server Client for 32-bit Windows, you add it to the Client Update Database. However, only one version of the client can be enabled. The enabled client is used for client updating.
7. Click **Finish** to copy the client installation files to the Client Update Database.

Removing a Client From the Client Update Database

It is important to delete Clients that are not used from the client update database. A database with multiple versions of the same client unnecessarily slows the checking procedure that is carried out each time a user connects to the server.

To remove a Client from the database

1. In the Client Update Configuration window, select the Client you want to remove from the database.
2. From the **Client** menu, choose **Delete**. A message box asks you to confirm the deletion.
3. To remove the Client, click **Yes**.

Changing the Properties of the Client

Use the **Properties** dialog box to set properties for an individual Client. Individual Client properties override the database properties.

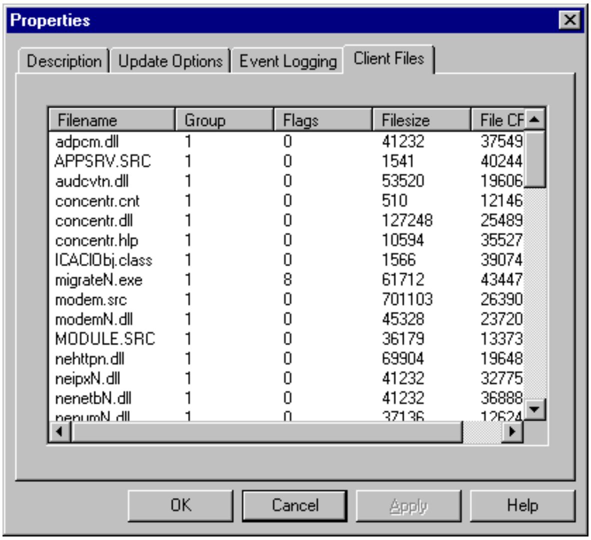
To change the properties of a Client

1. In the ICA Client Update Configuration window, select the Client whose properties you want to edit.
2. On the **Client** menu, choose **Properties**.

   The **Properties** dialog box contains tabs labeled **Description**, **Update Options**, **Event Logging**, and **Client Files**.

   The **Description** tab of the **Properties** dialog box lists the client name, product number, model number, and version number.
3. To update the same platform Client to this version, select **Enabled**.
4. Use the **Update Options** tab to configure update options for the client.
   - Under **Client Download Mode**, select **Ask user** to give the user the choice of accepting or postponing the update process. Select **Notify user** to notify the user of the update and require the client update. Select **Transparent** to update the user's Client software without notifying or asking the user.
   - Under **Version Checking**, select **Update older client versions only** to update only client versions that are older than the new client. Select **Update any client version with this client** to update all client versions to this version. Select this option to force an older client to replace a newer client.
   - Select the **Force Disconnection** option to require users to disconnect and complete the update process after downloading the new client.
   - Select the **Allow Background Download** option to allow users to download new client files in the background while they continue to work. Clear this check box to force users to wait for all client files to download before continuing.
   - Type a message to be displayed to users when they connect to the server.

5. Use the **Event Logging** tab to configure logging settings for this client.
    - Select the **Log Downloaded Clients** option to write an event to the event log when a client is updated
    - Select the **Log Errors During Download** option to write errors that occur during a client update to the event log
6. Use the **Client Files** tab to view the list of files associated with this client.



Filename	Group	Flags	Filesize	File CF
adpcm.dll	1	0	41232	37549
APPSRV.SRC	1	0	1541	40244
audcvtn.dll	1	0	53520	19606
concentr.cnt	1	0	510	12146
concentr.dll	1	0	127248	25489
concentr.hlp	1	0	10594	35527
ICAClObj.class	1	0	1566	39074
migrateN.exe	1	8	61712	43447
modem.src	1	0	701103	26390
modemN.dll	1	0	45328	23720
MODULE.SRC	1	0	36179	13373
nehttpn.dll	1	0	69904	19648
neipxN.dll	1	0	41232	32775
nenetbN.dll	1	0	41232	36888
nenumN.dll	1	0	37136	12624

This screen capture shows the Properties dialog box listing information about client files associated with the current client. Information the Properties dialog box displays about each client file is file name, group, flags, file size, and file CRC.

7. Click **OK** when you finish configuring the settings for the client.

Index