Administrator's Guide

Citrix® ICA® Win32 Clients

Version 6.30

Citrix Systems, Inc.

Contents

Before You Begin

This manual is for system administrators responsible for installing, configuring, deploying, and maintaining the Citrix ICA Win32 Clients (also called the Citrix ICA Clients for 32-bit Windows). This manual assumes knowledge of:

• The MetaFrame server to which your ICA Clients connect

• The operating system on the client device (Windows 9x, Windows Me, Windows NT, Windows 2000, or Windows XP)

• Installation, operation, and maintenance of network and asynchronous communication hardware, including serial ports, modems, and device adapters

How to Use this Guide

To get the most out of the *Citrix ICA Win32 Clients Administrator's Guide,* review the table of contents to familiarize yourself with the topics discussed.

This guide contains the following sections:

Chapter	Contents
Chapter 1, "Before You Begin"	Gives an overview of this document.
Chapter 2, "Introducing the Citrix ICA Win32 Clients"	Highlights new and existing features of the ICA Win32 Clients; includes a discussion of which ICA Client(s) to use.
Chapter 3, "Installing and Configuring the ICA Win32 Program Neighborhood Agent"	Includes instructions for installing, configuring, and running the ICA Win32 Program Neighborhood Agent.
Chapter 4, "Installing and Configuring the ICA Win32 Web Client"	Includes instructions for installing, configuring, and running the ICA Win32 Web Client.
Chapter 5, "Installing and Configuring the ICA Win32 Program Neighborhood Client"	Includes instructions for installing, configuring, and running the ICA Win32 Program Neighborhood Client.

Chapter	Contents
Chapter 6, "Configuring Features Common to the ICA Win32 Clients"	Includes instructions for configuring new and existing features common to the ICA Win32 Clients.
Chapter 7, "Implementing Security Measures for the ICA Win32 Clients"	Includes information about directing ICA traffic through proxy servers and firewalls, and configuring the ICA Clients to use Citrix Secure Gateway or SSL Relay.
Chapter 8, "Updating the ICA Win32 Clients"	Includes instructions for deploying ICA Win32 Client updates across your network using the Client Auto Update feature.

The MetaFrame Application Server product family 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 computer desktops. When referring to all types of resources you can publish with MetaFrame, 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.

Document Conventions

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

Convention	Meaning
Boldface	Commands, names of interface items such as text boxes and option buttons, and user input.
Italics	Placeholders for information or parameters that you provide. For example, *filename* in a procedure means you type the actual name of a file. Italics also are used for new terms and the titles of books.
UPPERCASE	Keyboard keys, such as CTRL for the Control key and F2 for the function key that is labeled F2.
Monospace	Text displayed at a command prompt or in a text file.
%SystemRoot%	The Windows system directory, which can be WTSRV, WINNT, WINDOWS, or other name specified when Windows is installed.
{ braces }	A series of items, one of which is required in command statements. For example, **{ yes	no }** means you must type **yes** or **no**. Do not type the braces themselves.

Convention	Meaning
[brackets]	Optional items in command statements. For example, [**/ping**] means that you can type **/ping** with the command. Do not type the brackets themselves.
| (vertical bar)	A separator between items in braces or brackets in command statements. For example, **{ /hold | /release | /delete }** means you type **/hold** or **/release** or **/delete**.
… (ellipsis)	You can repeat the previous item or items in command statements. For example, **/route:***devicename*[,...] means you can type additional *devicenames* separated by commas.
▶	Step-by-step procedural instructions.

Finding More Information

This manual contains conceptual information and installation and configuration steps for the ICA Win32 Clients. For additional information, consult the following:

- The online help for the ICA Client you deploy

- The *Citrix ICA Client Administrator's Guides* for the other ICA Clients you deploy

- The *Configuration Guide for the ICA Win32 Clients*, available from the Citrix Web site at http://www.citrix.com/support

- The documentation included in your MetaFrame server package for instructions about installing, configuring, and maintaining your MetaFrame servers

This book and other Citrix documentation are available in Adobe PDF format in the following locations:

- The documentation directory of your MetaFrame XP Components CD

- http://www.citrix.com/download; click the ICA Client platform for which you want information

- http://www.citrix.com/support; click Product Documentation and choose ICA Client

Using the Adobe Acrobat Reader, you can view and search the documentation electronically or print it for easy reference. You can download the Adobe Acrobat Reader for free from the Adobe Web site at http://www.adobe.com/.

Important   Always consult the Readme files for your MetaFrame server and the Citrix ICA Client for any last-minute updates, installation instructions, and corrections to the documentation.

Citrix on the World Wide Web

The Citrix Web site, at http://www.citrix.com/, offers a variety of information and services for Citrix customers and users. From the Citrix home page, you can access Citrix online Technical Support Services and other information designed to assist you, including the following:

- Downloadable Citrix ICA Clients (at http://www.citrix.com/download).

- Citrix Product Documentation Library, containing the latest documentation for all Citrix products (at http://www.citrix.com/support, select Product Documentation). You can download updated editions of the documentation that ships with Citrix products and supplemental documentation that may be available only from the Citrix Web site.

- Program information about Citrix Preferred Support Services options.

- An FTP server containing the latest service packs, hotfixes, utilities, and product literature for download.

- An online Solution Knowledge Base containing an extensive collection of application notes, technical articles, troubleshooting tips, and white papers.

- Interactive online Solution Forums for discussion of technical issues with other users.

- Frequently Asked Questions pages with answers to common technical and troubleshooting questions.

- Information about programs and courseware for Citrix training and certifications.

- Contact information for Citrix headquarters, including worldwide, European, Asia Pacific, and Japan headquarters.

- The Citrix Developer Network (CDN) at http://www.citrix.com/cdn. This open enrollment membership program provides access to developer tool kits, technical information, and test programs, for software and hardware vendors, system integrators, ICA licensees, and corporate IT developers who incorporate MetaFrame server-based computing solutions into their products.
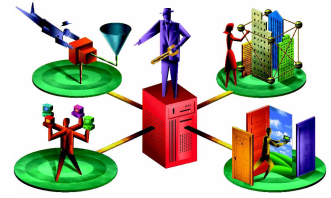
Providing Feedback About this Guide

We strive to provide accurate, clear, complete, and usable documentation for Citrix products. If you have any comments, corrections, or suggestions for improving our documentation, we want to hear from you.

You can send email to the documentation authors at documentation@citrix.com. Please include the product name, version number, and the title of the document in your message.

Introducing the Citrix ICA Win32 Clients

This book introduces the Citrix ICA Win32 Clients. Use the ICA Win32 Clients on computers running 32-bit Windows operating systems. For information about ICA Clients for other client devices and operating systems, refer to the documentation included on your MetaFrame XP installation CDs, or visit our Web site at http://www.citrix.com/.

This chapter focuses on Version 6.30 of the ICA Win32 Clients. It is designed to help you decide which ICA Clients to use in your computing environment and how to deploy them.

The following topics are covered:

- Overview

- Deciding Which ICA Win32 Client to Use

- New in this Release

- Improved in this Release

- ICA Win32 Client Features

- Preconfiguring and Deploying the ICA Win32 Clients

Overview

ICA Clients are the client-side software components of the MetaFrame Application Server product family. Combining the ease of use you expect with the security features you require, ICA Clients offer point-and-click, secure access to applications, content, and entire computer desktops published on MetaFrame servers.

Your Choices of ICA Win32 Clients

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. In response to this diversity and flux, MetaFrame offers you a choice of three ICA Win32 Clients, and an ICA Java Client for use on 32-bit Windows systems:

ICA Win32 Program Neighborhood Agent

- The state-of-the-art ICA Win32 Client
- Transparent integration of published resources with the client desktop
- Native support for the full feature set of MetaFrame XP
- Native support for the Windows 2000 and Windows XP operating systems
- Requires Citrix NFuse Classic 1.7

ICA Win32 Web Client

- The lean ICA Win32 Client for quick distribution
- Web browser-based access to published resources from links on your Web page
- No user interface, completely hands-off user experience
- Limited support for the full feature set of MetaFrame XP
- Native support for the Windows 2000 and Windows XP operating systems
- Requires Citrix NFuse Classic or Application Launching and Embedding (ALE)

ICA Win32 Program Neighborhood Client

- Program Neighborhood user interface
- Requires initial user configuration
- Native support for MetaFrame XP features
- Native support for the Windows 2000 and Windows XP operating systems
- Does not require NFuse Classic

ICA Java Client

- The platform-independent, zero-install ICA Client
- No user interface, completely hands-off user experience
- Limited support for the full feature set of MetaFrame XP

Deciding Which ICA Win32 Client to Use

Each ICA Win32 Client offers a robust and easy-to-manage solution for delivering your published resources to users. To decide which ICA Client or ICA Clients best fits your needs, consider the way you want users to access your published resources and the way you want to manage your ICA Client population.

Your Choices in Delivering Your Published Resources

This section outlines the choices you have in delivering published resources to users, and which ICA Win32 Client to use with each delivery method. For information about using the ICA Java Client to deliver published resources to users, see the *ICA Java Administrator's Guide*. For in-depth information about each ICA Client, see the client-specific chapters later in this book.

Having a choice of three different ICA Win32 Clients, you can choose among three different methods for delivering published resources to users. Each method provides a different user experience when accessing published resources:

- Desktop-based access to published resources

- Web-based access to published resources

- User interface-based access to published resources

Desktop-Based Access to Published Resources

If you want users to access your published resources entirely from within their familiar Windows desktop environment, use the ICA Win32 Program Neighborhood Agent.

ICA Win32 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 double click them, move them, copy them, 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   The configuration data for all Program Neighborhood Agents is defined in a single configuration file. As a result, you can dynamically manage and control your client population network-wide from a single location and in real time.

This ICA Client requires NFuse Classic, which is included in your MetaFrame XP, Feature Release 2 media pack. It also requires the presence on client devices of Microsoft Internet Explorer 5.0 or later, or of Netscape Navigator 5.0 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).

Depending on the deployment method you choose for this client, the size of its installation file ranges between approximately 1.9MB and 2.8MB.

For information about installing, configuring, and using the Program Neighborhood Agent, see "Installing and Configuring the ICA Win32 Program Neighborhood Agent" on page 37 of this guide.

Web-Based Access to Published Resources

If you want users to access your published resources from within their familiar Web browser, use the ICA Win32 Web Client.

ICA Win32 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 ICA 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 ICA Web Client for access to resources enabled with NFuse Classic, and for access to resources published with traditional Application Launching and Embedding (ALE). You publish links to your resources with NFuse Classic or by using an HTML wizard. Both products are included in your MetaFrame XP, Feature Release 2 media pack.

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

The ICA Web Client does not include a user interface or online Help files. Approximately 1.8MB in size, this ICA Client is quicker to download and install than the other ICA Win32 Clients.

For information about installing, configuring, and using the ICA Web Client, see "Installing and Configuring the ICA Win32 Web Client" on page 73.

User Interface-Based Access to Published Resources

If you want users to access your published resources from within a distinct user interface, use the ICA Win32 Program Neighborhood Client.

ICA Win32 Program Neighborhood Client

User Experience.   Using the ICA Program Neighborhood Client, users browse for application sets or create custom ICA connections to MetaFrame servers or to individual published resources. Icons representing application sets and custom ICA connections appear in the ICA Client's Program Neighborhood window.

Client Management and Administration.   Choose this ICA Client if you do not want to publish your resources using NFuse Classic. Although this client does not require NFuse Classic, it fully supports work with resources published through NFuse Classic or Application Launching and Embedding (ALE). You can configure numerous settings before you deploy the client software to users. This allows users to install the client and begin using it immediately, without having to configure settings.

Depending on how you choose to deploy this client, its installation file ranges in size between approximately 2.2MB and 3.2MB.

For information about installing, configuring, and using the Program Neighborhood Client, see "Installing and Configuring the ICA Win32 Program Neighborhood Client" on page 79.

If you are planning to use NFuse Classic and have not yet deployed any ICA Clients, use the ICA Win32 Program Neighborhood Agent or the ICA Win32 Web Client.

Your Choices in Managing Your ICA Client Population

Managing your ICA Client population entails the following aspects:

- Preconfiguring and Deploying ICA Clients
- Controlling ICA Client Behavior

Preconfiguring and Deploying ICA Clients

You can configure all ICA Win32 Clients for convenient, near-silent user installation and distribute them over the Internet, your corporate intranet, or from a network share point. This facilitates ICA Client deployment across your network and puts the latest ICA Client updates at users' fingertips.

For more information about preconfiguring and deploying ICA Clients, see the section "Preconfiguring and Deploying the ICA Win32 Clients" on page 31.

Controlling ICA Client Behavior

The ICA Win32 Program Neighborhood Agent offers a unique client management feature: Once launched, the ICA Client continuously downloads its configuration data from a configuration file on your NFuse Classic server. You can modify the configuration data at any time as a means to manage and control your client population network-wide from a single location and in real time.

The ICA Win32 Program Neighborhood and ICA Win32 Web Clients do not offer this capability.

New in this Release

Version 6.30 of the ICA Win32 Clients provides native support for MetaFrame XP 1.0 for Windows, Feature Release 2, and for the 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.

Feature Highlights

Highlights of Version 6.30 of the ICA Win32 Clients include:

- Faster data transfers to and from MetaFrame XP servers

- Support for the latest Internet security standards

- Improved collaborative capabilities

- Enhanced user experience

For information about enabling and configuring the new features on MetaFrame XP servers, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

For detailed instructions about enabling and configuring the new features on client devices, see "Configuring Features Common to the ICA Win32 Clients" on page 79 or the chapter about the specific ICA Win32 Client you plan to deploy.

FR2   **Important**   Features marked with the FR2 symbol require the MetaFrame XP server to be licensed for Feature Release 2. Features not so marked require Service Pack 2 for MetaFrame XP to be installed.

User-to-User Shadowing

FR2 Shadowing is the process of monitoring a user's session remotely and, optionally, participating in the session using your own keyboard and mouse. Previously reserved for MetaFrame administrators, this feature no longer requires administrative rights. You can now make shadowing available to users, adding powerful collaborative capabilities to a variety of applications, including:

- Help desks, where trained personnel troubleshoot applications

- Training organizations, where students observe instructors, and instructors monitor student performance

- Presentation sessions, where multiple remote users attend a presentation hosted by a single user

Shadowing does not require client configuration. This feature is governed by user policies set on the MetaFrame XP server. For information about enabling and configuring this feature, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

Smart Card Support

FR2 The ICA Win32 Clients offer support for a number of smart card readers. If smart card support is enabled on both the server and client sides, you can use smart cards for the following purposes:

Smart card logon authentication support.   Use smart cards to authenticate users to MetaFrame XP servers.

Smart card application support.   Allow users to digitally sign email within smart card-aware published applications. As of this writing, the following applications are smart card-aware:

- Microsoft Outlook 2000; Microsoft Outlook XP

- Microsoft Internet Explorer 5.5; Microsoft Internet Explorer 6.0

Smart card data is security-sensitive. MetaFrame XP offers a variety of security solutions to protect the transmission of your security-sensitive data. See Chapter 7 of this guide for more information.

Smart card devices and published applications must comply with the PC/SC industry standard.

Smart card support requires configuration on all ICA Win32 Clients.

Content Redirection

FR2 Using MetaFrame XP server-based file type association, you can redirect application launching from server to client or from client to server. As a result, the server controls whether a published or a local application is invoked when a user opens a particular file. MetaFrame XP supports the following types of content redirection:

Server-to-client content redirection: Using server-to-client content redirection, you can leverage local applications to offload MetaFrame server resources. Server-to-client content redirection forces Web or multimedia links embedded in published applications to invoke the associated client-side application.

For example: Using a published application, you are viewing a text file that contains an embedded URL to a multimedia file. When you click the URL, the multimedia file invokes the associated multimedia player on the client device rather than the player published on the MetaFrame server. If the associated multimedia player is not installed on the client device, the file invokes the published player.

This variation of content redirection does not require client configuration.

ICA Win32 Program Neighborhood Agent only

Client-to-server content redirection: Double-clicking a client-side file invokes the associated published application. Opening a local text document, for example, invokes the associated published application.

For example: You receive an email attachment, but do not have an application installed on your local device that can open the attachment. Instead of presenting you with an error message when you try to open the attachment, it invokes the associated published application on the MetaFrame server.

Functionally equivalent to extended parameter passing, a feature introduced in Feature Release 1 of MetaFrame XP 1.0 for Windows, client-to-server content redirection allows you to enforce all underlying file type associations from the MetaFrame server, eliminating the need to configure extended parameter passing on individual client devices.

This variation of content redirection does not require client configuration.

Enhanced Content Publishing

ICA Win32 Program Neighborhood Agent or any ICA Win32 Client with Citrix NFuse Classic

FR2 You no longer need to download published content to the client device and open it with a local application. Instead, you can associate and open published content with a published application on the MetaFrame server. This allows you to access published content from client devices that do not have applications other than a Web browser installed.

For example: Opening a published image file invokes the associated published application.

Support for Citrix NFuse

All ICA Win32 Clients provide full support for Citrix NFuse Classic Version 1.7 and for Citrix Enterprise Services for NFuse.

Support for TLS Encryption of ICA Traffic

The ICA Win32 Clients now support TLS 1.0, the successor to SSL 3.0, for environments that demand it. TLS (Transport Layer Security) is the standardized form of SSL (Secure Sockets Layer). Both are cryptographic security protocols designed to ensure the integrity and privacy of data transfers across public networks. SSL and TLS are functionally equivalent. Certain organizations have a security policy that requires TLS rather than SSL.

Support for Citrix Secure Gateway 1.1

The ICA Win32 Clients provide full support for Citrix Secure Gateway Version 1.1. Citrix Secure Gateway acts as a secure Internet gateway between SSL/TLS-enabled ICA Clients and MetaFrame XP servers. The Internet portion of ICA traffic between client devices and the Citrix Secure Gateway server is SSL/TLS-encrypted. SSL/TLS is the Internet standard 128-bit encryption technology used for client-server authentication, and to ensure the integrity and privacy of data transfers across public networks.

Enhanced Internet Proxy Support

As an alternative to SOCKS Proxy, the ICA Win32 Clients also support Secure Proxy (also known as *Security Proxy, HTTPS Proxy,* and *SSL-tunneling*). Proxy authentication is also supported.

The ICA Win32 Clients can auto-detect proxy servers by obtaining the details of proxy servers on the network from the Web browser.

Roaming User Reconnect

FR2 This feature adds roaming capabilities to ICA sessions. Previously, ICA sessions were identified by the name of the client device from which they were initiated, and limited to that device. Feature Release 2 of MetaFrame XP makes ICA sessions portable: You can start a session on one client device and resume your work from any ICA-enabled device anywhere, anytime.

Roaming User Reconnect does not require client configuration.

Improved in this Release

In addition to offering a range of new features, Version 6.30 of the ICA Win32 Clients provides you with performance enhancements in the following areas:

- Increased performance

- Enhanced printer support and management

- Program Neighborhood Agent usability improvements

ICA Win32 Client Features

Universal Print Driver Support

The Citrix Universal Print Driver is a standard Windows 2000 or Windows NT print driver that encapsulates print jobs in PCL4 format. A client-based interpreter renders the print job using the client device's local print driver and printing services. The Universal Print Driver renders print jobs in monochrome at up to 300dpi on a standard set of forms.

The Universal Print Driver generates smaller print jobs, which can significantly improve performance when printing over WAN or dial-up connections.

Note   If the MetaFrame server is configured to automatically create both native driver and Universal Print Driver printers on the client device, you must explicitly select a printer to process print jobs. If the print job requires color or advanced printing options such as duplex printing, select the standard printer, which uses the printer's native driver.

Auto Client Reconnect

ICA sessions can be dropped for a variety of reasons, including unreliable networks, highly variable network latency, or range limitations of wireless devices.

Auto Client Reconnect is triggered when the ICA Client detects a disconnected session. When this feature is enabled on a MetaFrame XP server, users can transparently reconnect to their session without having to re-enter their logon credentials.

Automatic client reconnection does not occur if users exit applications without logging off.

Windows Installer Packages for ICA Win32 Clients

The ICA Win32 Program Neighborhood Agent and the ICA Win32 Program Neighborhood Client are available as Microsoft Windows Installer (.Msi) packages. If your network is based on Windows 2000 or later, you can take advantage of Microsoft Systems Management Server or Active Directory Services to deploy updated versions of these clients using the Windows Installer packages.

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

The ICA Win32 Web Client is not currently available as a Windows Installer package. Use the self-extracting executable to install the ICA Web Client.

Note   Use Microsoft Systems Management Server or Active Directory to update ICA Clients previously installed with Windows Installer packages, and use Client Auto Update to update ICA Clients previously installed with the self-extracting executable. The two technologies are not interchangeable.

Note   To install the ICA 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 ICA 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/.

Client Auto Update

Client Auto Update allows you to update ICA Client installations using the self-extracting executable (.Exe) files from a central location, rather than having to manually install new client versions on each client device. This feature does not require Microsoft Systems Management Server or Active Directory Services.

Client Auto Update stores self-extracting executables of new ICA Client versions in the Client Update Database. It downloads the latest versions of the ICA Client software to client devices when users connect to the MetaFrame server. MetaFrame for UNIX does not support the Client Update Database. To use the Client Update Database, you must have a MetaFrame or MetaFrame XP for Windows server in your server farm.

ICA Client Auto Update works with all transport types supported by ICA (TCP/IP, IPX, NetBIOS, and serial).

ICA Client Auto Update supports the following features:

- Automatically detects older client files

- Transparently copies new files over any ICA connection

- Provides full administrative control of client update options for each client

- Updates clients from a single database on a network share point

- Safely restores older client versions when needed

Note   Use Client Auto Update to update ICA Clients previously installed with the self-extracting executable. Use Microsoft Systems Management Server or Active Directory to update ICA Clients previously installed with Windows Installer packages. The two technologies are not interchangeable.

Published Content Support

In addition to applications, which are executable (.Exe) files, you can also publish content files, which are non-executable. Content files include documents, Web pages, and audio/video files. You publish content on the MetaFrame server in the same manner as applications. The server then "pushes" the content to the client device.

Users open published content with the associated content viewer or player on the client device or the server, depending on published application settings.

Novell Directory Services Support

When launching ICA Win32 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.

DNS Name Resolution

You can configure ICA Win32 Clients that use the XML Service to connect to the MetaFrame farm to request a Domain Name System (DNS) name instead of a server's IP address.

Extended Parameter Passing

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

Seamless Windows

Citrix ICA Win32 Clients support the seamless integration of local and remote applications on the local desktop. By selecting the Seamless Windows option when configuring a connection, you can, in a single session, access multiple applications, have fully functional local keyboard controls (such as ALT+TAB), switch between local and remote applications on the local taskbar, and define remote application icons on the local desktop.

Client Device Mapping

ICA Win32 Clients support client device mapping. As a result, published resources running in an ICA session can access printers, disk drives, and COM port devices attached to the local client device.

Client Drive Mapping

You can map server drives that are available to the client within an ICA session to the client drive. For example, you can map server drive H to drive C on the local device running the ICA Client. Applications, Windows Explorer, or File Manager use these mappings just like any other network mappings. Drive letters used for drive mapping are configurable, and long file names are supported.

Client Printer Mapping

Client printer mapping allows published resources to access local client printers. You browse for a client printer the same way as for network printers. You can transparently access your local printers from within the ICA session.

Client COM Port Mapping

Just as with client printer mapping, the ICA Client COM port redirector allows published applications to access virtually any peripheral attached to the COM ports of the client device.

Sound Support

ICA Client sound support allows a client device to execute sound files on the MetaFrame server and play them back on the sound system of the local client device if:

- The associated published application renders sound in waveform-audio format

- The client device is equipped with a sound card that supports the waveform-audio format

Waveform-audio is the standard sound format of Windows operating systems. ICA supports output in 8- and 16-bit mono and stereo at 8, 11.025, 22.05, and 44.1KHz. You can configure audio support to use one of three different sound compression schemes: high, medium, and low. Each scheme provides different sound quality and bandwidth usage.

This feature is not available from MetaFrame for UNIX 1.0 and 1.1 servers.

Per-User Time Zone Support

This feature allows the ICA session to reflect the time zone of the client device, regardless of the time zone of the MetaFrame server hosting the session.

For example: A user in Los Angeles, which is in the Pacific time zone, logs on to a MetaFrame server in New York City, which is in the Eastern time zone, and launches Microsoft Outlook as a published application. Microsoft Outlook stamps emails sent during this ICA session with the user's Pacific time zone information.

TAPI Support

ICA Win32 Clients provide TAPI modem support for dial-up connections to MetaFrame servers. TAPI support allows the Win32 Client to detect the presence of TAPI Version 1.4 or later modems on the client device. Users do not need to manage separate modem entries for their local communications programs.

When the ICA Client detects a TAPI modem, it uses the modem installation and configuration utilities built into Windows to manage the modem. If the client device is not TAPI-capable, the ICA Client uses its own modem installation and configuration utilities.

Dialing Prefixes

ICA Clients support dialing prefixes. Dialing prefixes allow you to add special dialing codes, as required by different telephone systems, for dialing out and accessing a remote MetaFrame server.

The most common use of dialing prefixes is defining different dialing methods for different telephone systems. For example, a user with a laptop computer may need to dial 9 to get an outside line at the office, and may need to dial 1 plus the area code when working on the road or at home. In this case, the user can define a dialing prefix named Office for use when dialing out from the office, and a prefix called Remote for use when dialing in from the road or at home.

Windows Clipboard Integration

Users can copy, cut, and paste data between ICA sessions and local applications using the Windows clipboard.

Low Bandwidth Requirements

The Citrix ICA protocol typically uses 20K of bandwidth for each session.

Disk Caching and Data Compression

These features increase performance over low speed asynchronous and WAN connections. Disk caching stores commonly used portions of your screen (such as icons and bitmaps) locally, increasing performance by avoiding retransmission of locally cached data. Data compression reduces the amount of data sent over the communications link to the client device.

SpeedScreen Latency Reduction

SpeedScreen Latency Reduction is a collective term used to describe functionality that enhances the user's experience on slower network connections.

SpeedScreen Latency Reduction is not available from MetaFrame for UNIX 1.0 and 1.1 servers.

The elements of SpeedScreen Latency Reduction are:

Local Text Echo

This ICA Client and MetaFrame server option accelerates the display of text input on the client device.

Mouse-Click Feedback

This ICA Client and MetaFrame server option provides visual feedback for mouse clicks to show that the user's input is being processed.

Business Recovery

ICA Clients support multiple server sites (such as a primary and hot backup server) with different addresses for the same published resource name.

This feature provides uninterrupted connections to published resources in the event of a primary server disruption.

TCP/IP+HTTP Server Location

TCP/IP+HTTP server location allows you to retrieve MetaFrame server and published resource information across network configurations that restrict UDP broadcasts.

Configuring the ICA Win32 Clients to use the TCP/IP+HTTP network protocol has several advantages for most server farms:

- The protocol uses XML data encapsulated in HTTP packets, and uses TCP port 80 by default. Most firewalls are set to allow HTTP packets to pass on port 80.

- The protocol does not rely on UDP or broadcasts to locate servers in the server farm.

- The Citrix XML Service works in a server farm that contains MetaFrame XP servers alone or in combination with NFuse Classic, which allows users to connect to application portals with their Web browsers.

Routers pass TCP/IP packets between subnets, allowing client devices to locate servers that are not on the same subnet.

Wheel Mouse Support

If you run applications that offer wheel mouse support, the ICA Win32 Client transmits the wheel mouse movements in the same manner it transmits other mouse data. ICA Win32 Client wheel mouse support requires MetaFrame 1.8 Service Pack 1 or MetaFrame XP or later, and a local client device that supports wheel mouse functionality.

Multiple-Monitor Support

ICA Win32 Clients support multiple monitors connected to a single computer. Multiple-monitor support is available only when connecting to MetaFrame 1.8, Feature Release 1, and MetaFrame XP servers.

Pass-Through Authentication

Pass-through authentication provides the ability to pass the user's desktop password to the server, eliminating the need for multiple system and application authentications.

Panning and Scaling

Panning allows you to scroll an ICA session image configured at a screen resolution that is higher than that of the client device. Scaling allows you to shrink an ICA session image to fit your screen.

Note   You can shrink a Win32 Client session window to a minimum of 64 pixels in width. However, the Windows operating system may enforce a greater limit based on the prevailing desktop scheme, which overrides the ICA Client scaling limit.

Preconfiguring and Deploying the ICA Win32 Clients

You can deliver ICA Clients to your users and install the software using the following methods:

- Using Microsoft Systems Management Server (SMS) or Active Directory Services in Windows 2000

- Creating an ICA Client download Web site on a Web server

- Copying the ICA Win32 files to a network share point

- Creating ICA Client installation disks

- Using the Components CD included in your MetaFrame XP, Feature Release 2 media pack

For a detailed discussion of the latest deployment methods available, see the *MetaFrame XP Administrator's Guide*. If you are using MetaFrame XP in conjunction with NFuse Classic, see the *NFuse Classic Administrator's Guide* for information about deploying ICA Clients with NFuse Classic.

Using Microsoft Systems Management Server or Active Directory Services

The ICA Win32 Program Neighborhood Client and the Program Neighborhood Agent are available as Windows Installer (.Msi) packages for use with Microsoft Systems Management Server or Active Directory.

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

The installer package files for all three ICA Win32 Clients are located in the following directories (substitute *language* with the language of the ICA Client software) of the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\ica32

where *language* is one of:

- En (English)

- Fr (French)

- De (German)

- Ja (Japanese)

- Es (Spanish)

Identical installer package files for the Program Neighborhood Agent are also located in the directory:

Icainst*language*\ica32\pnagent

Note   The Windows Installer service (Instmsi.exe) is present by default on computers running the Windows 2000 operating system. If the client device is running Windows NT 4.0 or Windows 9x, you must install Windows Installer Version 1.1 or higher.

Creating an ICA Client Download Web Site on a Web Server

If you are not using NFuse Classic, Citrix offers an installation method that uses a Web browser on the client device as the interface for downloading the ICA Client. You can create an ICA Client download Web site on a Web server. Users access a setup page that contains a link to the ICA Win32 Client setup program.

You can download the elements required to create an ICA Client download Web site and the corresponding documentation from the Citrix Web site at http://www.citrix.com/download, or you can get the required files from the Components CD included in your MetaFrame XP, Feature Release 2 media pack. See the *MetaFrame XP Administrator's Guide* for more information about using this CD to create an ICA Client download Web site.

Deploying ICA Clients Over a Network

To deploy the ICA Win32 Client software over a network, follow the instructions below.

▶ **To deploy ICA Win32 Client software from a network share point**

1. Create a share point on a file server that is accessible to your users.

2. Copy the desired ICA Win32 Client executable (Ica32.exe for the Program Neighborhood Client; Ica32a.exe for the Program Neighborhood Agent; or Ica32t.exe for the Web Client) from the Components CD to the share point. These executables are located in the following directory (substitute *language* with the language of your server software):

   Icaweb*language*\ica32

   where *language* is one of:

   • En (English)

   • Fr (French)

- De (German)
- Ja (Japanese)
- Es (Spanish)

3. Supply your users with the path to the executable.

4. Users double-click the executable to begin the installation process.

Creating ICA Client Installation Disks

Use the ICA Client Creator to create client installation disks for the ICA Win32 Program Neighborhood Client. You will need three to four 3.5-inch floppy disks to create the client installation disks.

▶ **To create Citrix ICA Client installation disks**

1. From a MetaFrame XP server: Click **Start**> **Programs**> **Citrix> MetaFrame XP**> **ICA Client Creator**. The **Make Installation Disk Set** dialog box appears.

   From a MetaFrame 1.8 server: Click **Start**> **Programs**> **MetaFrame Tools**> **ICA Client Creator**. The **Make Installation Disk Set** dialog box appears.

2. In the Network Client or Service list, click the desired Citrix ICA Client. Select the **Format Disks** check box to format the disks when creating the installation media. Click **OK**.

3. Follow the on-screen instructions.

Using the Components CD

The Components CD included in your MetaFrame XP, Feature Release 2 media pack contains setup and installation files for all ICA Clients. You can use the Components CD to directly install ICA Client software on client devices that have CD-ROM drives, or copy the CD image to a network share point on a file server.

Notice of Change of Location of Client Name Information

This section contains information about a change introduced in Version 6.30 of the ICA Win32 Clients. The change affects the location in which ICA Win32 Clients store the name of the client device on which they are installed. This section is provided for informational purposes only. It is intended for system administrators who are used to modifying client names from within the ICA Client's Wfcname.ini file.

The MetaFrame server uses the client name to uniquely identify resources (such as mapped printers and disk drives) associated with a given client device. Therefore, each client device running an ICA Client must have a unique client name.

Prior to Version 6.30, ICA Win32 Clients stored the client name in the **ClientName=**<Client_Name> parameter of the **[WFClient]** section of the Wfcname.ini file, located in the %SystemRoot% directory of the client device.

As of Version 6.30, the client name is stored in the Windows Registry instead:

Registry key: **"HKLM\Software\Citrix\ICA Client"**

Key value of type **REG_SZ:** <**ClientName**>

This change affects Version 6.30 of all ICA Win32 Clients (ICA Win32 Program Neighborhood Agent, ICA Win32 Program Neighborhood Client, and ICA Win32 Web Client), and all supported installation mechanisms (Windows Installer package, self-extracting executable, and Auto Client Update).

As result of this change, all ICA Win32 Clients, Version 6.30 or later, retrieve the client name from the Windows Registry. ICA Win32 Clients prior to Version 6.30, if any, on the same client device are not affected by this change, and continue to retrieve the client name from the Wfcname.ini file.

When you install an ICA Win32 Client, Version 6.30 or later, using any of the above-mentioned installation mechanisms, the installation mechanism queries the Windows Registry for the presence of the associated registry key.

If the key is present:

• The Windows Installer package and the self-extracting executable present the client name, as retrieved from the Windows Registry, in the installation dialog box

• Auto Client Update transparently inherits the client name as retrieved from the Windows Registry

If the key is not present, but the Wfcname.ini file is present:

• The Windows Installer package and the self-extracting executable present the client name, as retrieved from Wfcname.ini, in the installation dialog box

• Auto Client Update transparently migrates the client name from the Wfcname.ini file to the Windows Registry

If neither the key nor the Wfcname.ini file is present:

• The Windows Installer package and the self-extracting executable present the machine name in the installation dialog box

• Auto Client Update transparently writes the machine name to the <ClientName> Windows Registry value
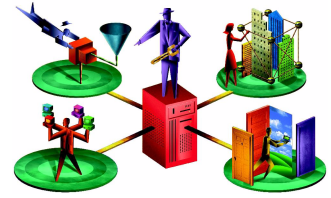
Note   When you install an ICA Win32 Client, Version 6.30 or later, the Wfcname.ini file is no longer created or updated. When you uninstall an ICA Win32 Client, Version 6.30 or later, neither the associated registry key nor the Wfcname.ini file, if applicable, is removed from the client drive.

Installing and Configuring the ICA Win32 Program Neighborhood Agent

This chapter explains how to install and configure the ICA Win32 Program Neighborhood Agent. The following topics are covered:

- Overview of the ICA Win32 Program Neighborhood Agent
- System Requirements
- Installing the ICA Win32 Program Neighborhood Agent
- Securing the Program Neighborhood Agent With SSL/TLS
- Configuring the ICA Win32 Program Neighborhood Agent (Server-Side)
- Customizing the ICA Win32 Program Neighborhood Agent (Client-Side)

Overview of the ICA Win32 Program Neighborhood Agent

Using the ICA Win32 Program Neighborhood Agent in conjunction with NFuse Classic, you can transparently 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.

Among the functions the Program Neighborhood Agent handles are:

- **User authentication.** The ICA Client presents user credentials to the MetaFrame server at startup and every time users launch published resources.
- **User direction.** The ICA Client directs users to the appropriate MetaFrame server for application and content enumeration and launching.

- **Application and content enumeration.** The ICA Client presents users with their individual set of published resources.

- **Application launching.** The ICA Client is the local engine used to launch published applications.

- **Desktop integration**. The ICA Client integrates a user's set of published resources with the user's desktop.

- **User preferences.** The ICA Client validates and implements local user preferences.

The configuration data for all Program Neighborhood Agents on your network is defined in a single configuration file on the NFuse Classic server. The client downloads the configuration data from the server using standard HTTP or HTTPS protocols. As a result, you can implement the client in combination with firewalls using port 80 (for HTTP) or 443 (for HTTPS).

The configuration file, Config.xml, resides in the Inetpub\wwwroot\Citrix\ PNAgent directory of the NFuse Classic server. It is preconfigured with functional and practical default settings to get you started. For information about modifying the presets in Config.xml to implement custom settings for your network, see "Configuring the ICA Win32 Program Neighborhood Agent (Server-Side)" on page 49.

ICA Win32 Program Neighborhood Agent Features

The ICA Win32 Program Neighborhood Agent offers the following features:

- Windows desktop integration

- User-to-user shadowing

- Smart card support

- Content redirection

- Enhanced content publishing support

- Roaming user reconnect

- Content redirection

- Support for SSL/TLS encryption of ICA session data

- Support for Citrix NFuse Classic 1.7 and Enterprise Services for NFuse

- Support for Citrix Secure Gateway 1.1

- Enhanced Internet proxy support

- Auto Client Reconnect

- Novell Directory Services support

- DNS name resolution support

- Seamless windows

- Client device mapping

- Sound support

- Windows Installer Packages

- Client Auto Update

- Windows clipboard integration

- Low bandwidth requirements

- SpeedScreen Latency Reduction

- Disk caching and data compression

- TCP/IP+HTTP server location

- Wheel mouse support

- Multiple monitor support

- Pass-through authentication

- Panning and scaling

- Per-user time zone support

System Requirements

To run the ICA Win32 Program Neighborhood Agent, client devices must meet the following requirements:

- Standard PC architecture, 80386 processor or greater as required for the operating system.

- Windows 95 (OSR2 or later), Windows 98, Windows Me, Windows 2000, Windows XP, or Windows NT 4 or later.

- 8MB RAM or greater for Windows 9x, 16MB RAM or greater for Windows NT 4.0, 32MB or greater for Windows Me and Windows 2000, and 128MB RAM for Windows XP.

- Internet Explorer Version 5.0 or later, or Netscape Navigator or Communicator Version 5.0 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 disk space.

- Windows-compatible sound card for sound support (optional).

- For network connections to the MetaFrame server, a network interface card (NIC) and the appropriate network transport software are required. Supported connection methods and network transports are:

  - TCP/IP+HTTP

  - SSL/TLS+HTTPS

For information about configuring the ICA Win32 Clients to use SSL or TLS to secure communications, see "Configuring and Enabling the Clients for SSL and TLS" on page 148.

Installing the ICA Win32 Program Neighborhood Agent

You can install the ICA Win32 Program Neighborhood Agent using one of the following packages:

- Ica32a.msi – a Windows Installer package for use with Windows 2000 Active Directory Services or Microsoft Systems Management Server, approximately 4.8MB in size

- Ica32a.exe – a self-extracting executable, approximately 2.75MB in size

---

Note   For a discussion of methods for deploying ICA Win32 Client software to users, see "Preconfiguring and Deploying the ICA Win32 Clients" on page 31 of this guide, and the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

Installing the ICA Win32 Program Neighborhood Agent with the Windows Installer Package

You can distribute the Program Neighborhood Agent Windows Installer package (Ica32a.msi) with Microsoft Systems Management Server or Windows 2000 Active Directory Services.

This package is located in the following directories (substitute *language* with the language of the ICA Client software) of the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\\ica32

Icainst*language*\\ica32\\pnagent

where *language* is one of:

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

Note   To install the ICA 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 ICA 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 an ICA Win32 Client that was installed with a Windows Installer package, run the **Add/Remove Programs** utility from the Control Panel, or run the installer package again, selecting the **Remove** option.

Configuring the Windows Installer Package for Silent User Installation

You can configure the Program Neighborhood Agent Windows Installer package for "silent" user installation. Windows Installer informs the user when the client software is successfully installed. The user must clear the Windows Installer message box.

▸ **To configure the Program Neighborhood Agent Window Installer package for silent user installation**

1. At a command prompt, type:

   msiexec /I <MSI_Package> /qn+ [Key=Value]…

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

2. You can set the following keys:

   INSTALLDIR=<Installation_Path>, where <Installation_Path> is the path to the directory where the ICA Client software is installed. The Program Neighborhood Agent software is installed in the Program Files\Citrix\PNAgent directory by default.

   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 Program Neighborhood Agent**.

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

INSTALLDIR=<Installation_Path>, where <Installation_Path> is the path to the directory where the ICA Client software is installed. The Program Neighborhood Agent software is installed in the directory Program Files\Citrix\PNAgent by default.

SERVER_LOCATION=<NFuse Classic_server_URL>. The default value is **PNAgent**. Enter the URL of the NFuse Classic server hosting the Config.xml file in the format http://*<servername>* or https://*<servername>*.

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

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.

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

DEFAULT_NDSCONTEXT=<Context1 [,…]>. Include this parameter if you want to set a default context for 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
```

Installing the ICA Win32 Program Neighborhood Agent with the Self-Extracting Executable

You can distribute the self-extracting executable (Ica32a.exe) to your users for direct installation on client devices.

This executable is located in the following directories (substitute *language* with the language of the ICA Client software) of the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\\ica32

Icainst*language*\\ica32\\pnagent

where *language* is one of:

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

Configuring the Self-Extracting Executable for Silent User 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 Program Neighborhood Agent to your users.

Important   You can use any standard compression utility to extract the client files from the packaged executable. However, you must use commercially available software to repackage the client files for distribution to your users.

▶ **To configure the self-extracting executable for silent user installation**

1. Extract the ICA 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 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, the setup program dialog boxes do not appear on the user's screen.

   ServerURL=NFuse Classic server URL. The default value is **PNAgent**. Enter the URL of the NFuse Classic server hosting the Config.xml file in the format **http://***servername* or **https://***servername* for SSL-secured communications.

   SetMachineNameClientName=Enter **On** 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 **On** to enable pass-through authentication.

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

   Save the file and exit the text editor.

3. (Optional; not required unless you are specifying a default NDS context.)

   Locate and open the file in a text editor.

   Locate the section named **[WFClient]**.

   Add the following line to the list of parameters and values in the **[WFClient]** section:

   DEFAULT_NDSCONTEXT=<Context1 [,…]>.

   If you are including more than one context, separate the contexts by a comma.

   Save the file and exit the text editor.

4. Repackage the client files for distribution to your users.

▸ **To install the Program Neighborhood Agent with the self-extracting executable**

1. To install the client from the Components CD included in your MetaFrame XP, Feature Release 2 media pack, run Ica32a.exe, from the following directory (substitute *language* with the language of your server software):

   Icaweb*language*\\ica32

   where *language* is one of:

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

2. The **Welcome** screen appears. Read the information on this screen and click **Next**.

3. The **License Agreement** screen appears. Read the license agreement and click **Yes** if you agree to the terms of the agreement.

   Setup searches your client device for previously installed versions of the ICA Win32 Program Neighborhood Agent.

4. If Setup does not detect a previous installation of the ICA Win32 Program Neighborhood Agent, go to Step 6.

    If Setup detects a previous installation of the ICA Win32 Program Neighborhood Agent, the **Choose Installation Type** dialog box appears:
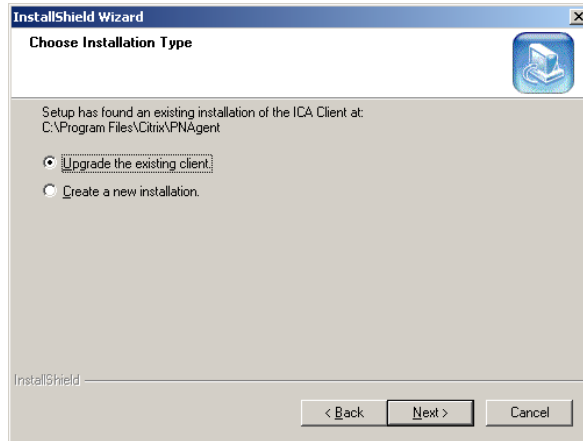
    The **Choose Installation Type** dialog box lets you choose to upgrade the existing client or to create a new and separate installation of the Program Neighborhood Agent in a new location. The default value is **Upgrade the existing client**.

    Click **Next**.

5. The **Choose Destination Location** dialog box appears:
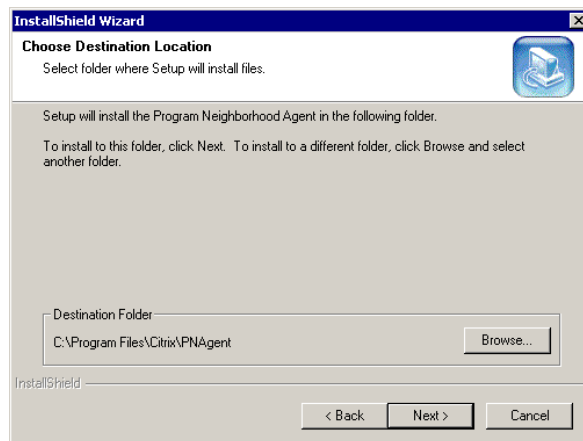
    Click **Browse** to select a different installation location. Click **Next** to accept the default path and continue.

6. The **Select Program Folder** dialog box appears:
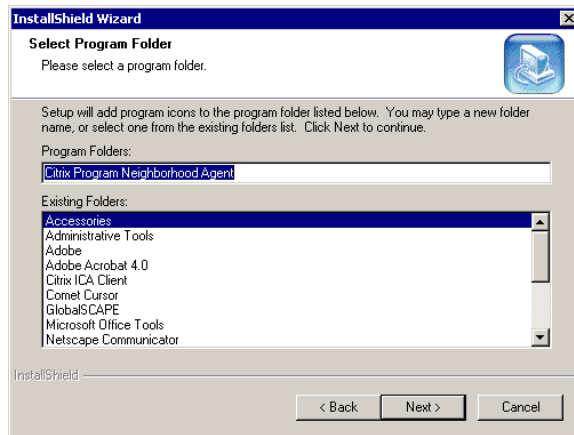
You can choose to use the default Citrix Program Neighborhood Agent folder, specify the name of a new program folder, or add the Program Neighborhood Agent icon to an existing folder. The program folder you specify is created if it does not already exist. Click **Next** to continue.

7. The **Server Address** dialog box appears.
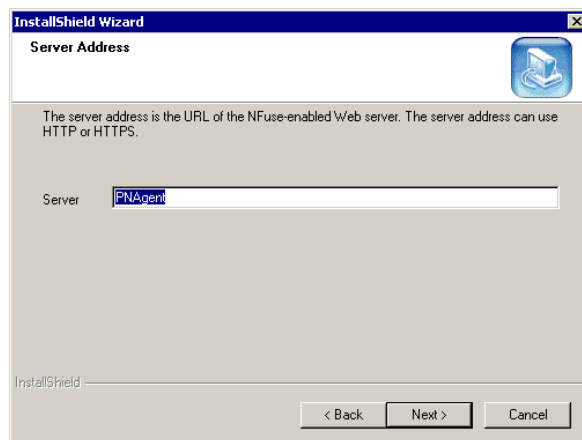
Enter the URL of the NFuse Classic server to connect to in the format http://*servername* (for non-secure connections), or https://*servername* (for secure connections).

Click **Next**.

8. The **Choose Installation Type** dialog box appears.
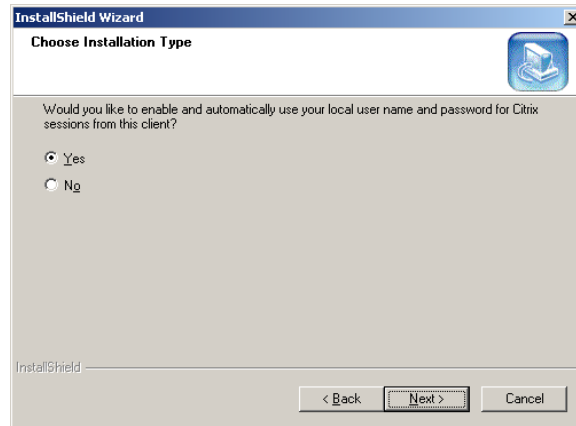
Select **Yes** to enable the pass-through authentication logon mode. This feature allows the ICA Client to access a user's local Windows user name, password, and domain information, and pass it to the server. Users are not prompted to log on to the Program Neighborhood Agent separately.

You must enable this logon mode in the Config.xml file on the NFuse Classic server to make it available to users. See "Configuring the ICA Win32 Program Neighborhood Agent (Server-Side)" on page 49 for instructions. If you enabled this logon mode in Config.xml, and you want the ICA Client to use this logon mode, select **Yes** on this screen, and then select **Pass-through authentication** as the logon mode in the client's **Properties** dialog box.

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

Click **Next**.

9. The **ICA Client Name** dialog box appears.
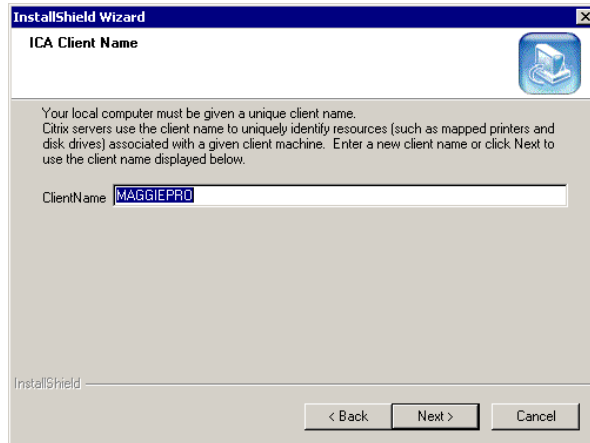
Specify a unique client name for your client device. MetaFrame servers use the client name to manage client printers and other system resources. If you do not assign a unique machine name to each client device, device mapping and application publishing may not operate correctly. You can use the client name displayed in the **Client Name** field or specify a new one.

When you are done, click **Next** to continue. A progress window appears, displaying the file names as they are copied to your hard drive.

10. You are informed that the Program Neighborhood Agent was successfully installed on the computer. Click **Finish**.

Configuring the ICA Win32 Program Neighborhood Agent (Server-Side)

The configuration data for all Program Neighborhood Agents on your network is defined in a single configuration file on your NFuse Classic server. The clients download the configuration data from the server based on the "client pull" concept:

Each client connects to the NFuse Classic server at specified intervals, queries the configuration file for the latest configuration data, and dynamically updates its properties in keeping with the data it receives.

Overview of the Configuration File, Config.xml

The configuration file, Config.xml, is placed in the Inetpub\wwwroot\Citrix\ PNAgent directory of the NFuse Classic server when you install NFuse Classic. The data it contains serves two purposes:

- To point the clients to the files that present users with their individual sets of published resources
- To control the client properties on the local desktop

Config.xml holds a range of parameters that appear as options in the client's **Properties** dialog box. Users choose from these options to set preferences for their ICA sessions, including logon mode, screen size, audio quality, and the locations of links to published resources on their desktop.

Config.xml is preconfigured with functional and practical default settings. It is ready for use without modification in most network environments. However, you can customize client behavior across the network by editing the file in a text editor. For example, you add or remove a particular logon mode option on all clients by adding or removing the corresponding element in Config.xml. With this powerful administrative tool, you can manage and control your client population from a single location and in real time.

Important   The text editor you are using must support UTF8 encoding.

Note   SSL/TLS-secured communications between client and NFuse Classic server, and smart card logon are not enabled by default. You must activate these features in Config.xml if you wish to use them. Instructions are provided on page 50.

Before deploying the Program Neighborhood Agent network-wide, you may want to install a copy of the client on a single client device. You can then evaluate the factory settings and determine whether you want to make adjustments to fit your particular network needs. Comparing between the configuration file and the client, you can monitor the effects of your edits on the client behavior.

CAUTION   The settings in Config.xml are global: all Program Neighborhood Agents on your network draw from the file's values. Changes you make to Config.xml affect all clients served by it.

Citrix recommends that you make a backup copy of Config.xml before modifying the file. Save the active copy of the file by the correct name (Config.xml) in the correct location (Inetpub\wwwroot\Citrix\PNAgent).

Securing the Program Neighborhood Agent With SSL/TLS

Make sure the client device meets all system requirements outlined on page 39 and page 147 of this guide.

To use SSL/TLS encryption for all communications between the Program Neighborhood Agent, the MetaFrame server, and the NFuse Classic server, the following configuration is necessary.

What You Need to do on the NFuse Classic Server

▸ **To use SSL/TLS to secure the communications between the Program Neighborhood Agent and the NFuse Classic server**

Change the URLs in the following elements of Config.xml to use HTTPS. Not all parameters may be specified in your version of the file.

- FolderDisplay/DesktopDisplay/Icon/Location

- ConfigurationFile/Location

- Request/Enumeration/Location

- Request/Resource/Location

Example: If the line specifying Request/Resource/Location is

```
<Location>http://server3.my.company.com/Citrix/PNAgent/
launch.asp</Location>,
```

change it to

```
<Location>https://server3.my.company.com/Citrix/PNAgent/
launch.asp</Location>
```

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.

What You Need to do on the MetaFrame Server

‣ **To use SSL/TLS to secure the communications between the Program Neighborhood Agent and the MetaFrame server**

Make sure to select the **Enable SSL** option on the **ICA Client Options** tab of the **Application Properties** dialog box in the Citrix Management Console for every application you want to secure. For more information, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

‣ **To use Citrix SSL Relay to secure communications between the MetaFrame server and the NFuse Classic server**

Make sure to specify in Config.xml the machine name of the server hosting the SSL certificate. See the *NFuse Classic Administrator's Guide* for more information about using SSL/TLS to secure communications between the MetaFrame server and the NFuse Classic 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 the section "Installing Root Certificates on the ICA Win32 Clients" on page 151 for more information.

‣ **To use SSL/TLS to secure the communications between the Program Neighborhood Agent and the NFuse Classic server**

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 client browser's online Help.

Enabling Smart Card Logon

This section assumes that smart card support is enabled on the MetaFrame 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 readying your network for using smart cards.

The smart card removal policy set on the MetaFrame 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.

▶ **To enable smart card logon with pass-through authentication**

This logon mode requires a smart card to be present or inserted in the smart card reader at logon time. With this logon mode selected, the Program Neighborhood Agent 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 re-enter a PIN to access published resources.

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.

1. Locate the **Logon** section of Config.xml.

2. Add the following element to the **Logon** section:
   <LogonMethod>smartcard_sson</LogonMethod>

3. Save your changes.

▶ **To enable smart card logon without pass-through authentication**

This logon mode requires a smart card to be present or inserted in the smart card reader at logon time. With this logon mode selected, the Program Neighborhood Agent prompts the user for a smart card PIN (Personal Identification Number) when it starts up and every time the user requests a published resource.

1. Locate the **Logon** section of Config.xml.

2. Add the following element to the **Logon** section:
   <LogonMethod>smartcard_prompt</LogonMethod>

3. Remove the **<LogonMethod>smartcard_sson</LogonMethod>** element if it is present.

4. Save your changes.

Enabling NDS Logon Support

See "Novell Directory Services Support" on page 116 for additional information about enabling NDS support.

1. Locate the **Logon** section of Config.xml.

2. Set the text between the **SupportNDS** tags to **true**:

   <SupportNDS>**true**</SupportNDS>

3. Enter the default tree name between the **DefaultTree** tags:

   <DefaultTree>**DefaultTreeName**</DefaultTree>

4. Save your changes.

Configuring Contents of the Properties Dialog Box

This section describes how to modify the contents of the Program Neighborhood Agent **Properties** dialog box. The **Properties** dialog box is available from the Windows System Tray of the client device.

Note   Modifying these parameters directly affects the contents of the **Properties** dialog box of all Program Neighborhood Agents on your network.

Hiding and Showing Tabs of the Properties Dialog Box

By default, the **Properties** dialog box features three tabs: **Server**, **Application Display**, and **ICA Options**. Depending on your network needs, you may not want certain options to be available to users. In this case, you can modify or disable a particular option, or hide a tab altogether.

This section provides an overview of the contents of each tab and instructions for disabling and re-enabling tabs. To enable, disable, or modify particular options on particular tabs, see "Enabling and Disabling User-Customizable Options" on page 58.

*Program Neighborhood Agent default **Properties** dialog box*
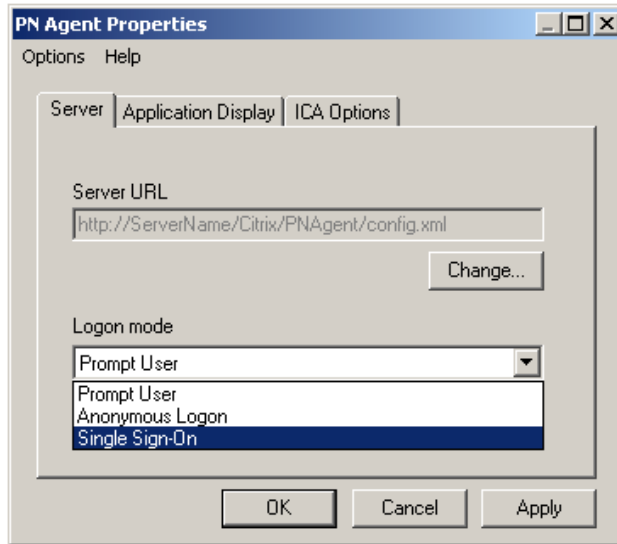
*Default contents of Program Neighborhood Agent **Properties** dialog box*

Server Tab	Application Display Tab	ICA Options Tab	Application Refresh Tab (optional)
Server URL	Show applications in Start menu	Display Window size	Refresh list when Program Neighborhood Agent starts
Logon Mode	Place application shortcuts in a folder of Start menu	Display Color Depth	Refresh list when remote application is launched
	Place folder in Programs submenu	Audio quality	Refresh list at hourly interval
	Show applications in desktop folder		
	Display applications in System Tray		

The Server Tab and its Contents

The options available on the **Server** tab let users modify the URL to the NFuse Classic server that hosts the configuration file, and select from among several logon modes.

The benefit of allowing users to modify the server URL lies in the resulting ease of migrating a segment of users to a different NFuse Classic server. To migrate a segment of users to a different NFuse Classic server, you instruct the members of that segment, by email or other means, to modify the server URL from the **Server** tab. A built-in safeguard ensures that user input of valid URLs only is accepted.

Providing a choice of multiple logon modes may be necessary in environments where multiple users use the same client device but different logon modes.

If you do not want users to have access to any of these options, you can hide the **Server** tab altogether. You can revert to showing the tab at any time.

To enable or disable particular options on the **Server** tab, see "Enabling and Disabling User-Customizable Options" on page 58.

▸   **To hide the Server tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **ServerSettings** tags to **false**:
    <ServerSettings>**false**</ServerSettings>

3. Save your changes.

▸   **To show the Server tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **ServerSettings** tags to **true**:
    <ServerSettings>**true**</ServerSettings>

3. Save your changes.

The Application Display Tab and its Contents

The options available on the **Application Display** tab let users place links to published resources in various locations of the client device, including the Windows desktop, the **Start** menu, the Windows System Tray, and any combination thereof. These options are provided as a convenience to users. Their impact is limited to the way remote content is visually represented on the client device.

Handling of user preferences set on this tab:

The Program Neighborhood Agent validates each user preference at startup by querying Config.xml. If a particular preference is within the range of permitted settings, the ICA Client applies the preference. Otherwise, it reverts to the server default for that option.

If you do not want users to have access to any of these options, you can hide the **Application Display** tab altogether. You can revert to showing the tab at any time.

If you hide the **Application Display** tab, published resources integrate with the desktop according to the settings for desktop integration configured on the MetaFrame server.

To enable or disable particular options on the **Application Display** tab, see "Enabling and Disabling User-Customizable Options" on page 58.

‣ **To hide the Application Display tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **FolderDisplaySettings** tags to **false**:
    <FolderDisplaySettings>**false**</FolderDisplaySettings>

3. Save your changes.

‣ **To show the Application Display tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **FolderDisplaySettings** tags to **true**:
    <FolderDisplaySettings>**true**</FolderDisplaySettings>

3. Save your changes.

The ICA Options Tab and its Contents

The options available on the **ICA Options** tab let users set preferences for the window size, color depth, and sound quality of ICA sessions. Users select from a number of settings available from a pull-down menu for each option.

The ability to customize ICA sessions is provided as a convenience to users. You cannot hide this tab. However, 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, and the settings configured on the MetaFrame server apply.

To force the server default for some or all of the options on this tab, see "Enabling and Disabling User-Customizable Options" on page 58.

Handling of user preferences set on this tab:

The Program Neighborhood Agent validates each user preference at startup by querying Config.xml. If a particular preference is within the range of permitted settings, the ICA Client applies the preference. Otherwise, it reverts to the server default for that option.

The Application Refresh Tab and its Contents

The options available on the **Application Refresh** tab let users customize the rate at which the ICA Client queries the enumeration file for updates. The enumeration file presents users with their particular sets of published resources. Because you may not want users to have control over the refresh rate, this tab is hidden from the **Properties** dialog box by default.

To enable or disable particular options on the **Application Refresh** tab, see "Enabling and Disabling User-Customizable Options" on page 58.

▶ **To show the Application Refresh tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **RefreshSettings** tags to **true**:
   <RefreshSettings>**true**</RefreshSettings>

3. Save your changes.

▶ **To hide the Application Refresh tab**

1. Locate the **UserInterface** section of Config.xml.

2. Set the text between the **RefreshSettings** tags to **false**:
   <RefreshSettings>**false**</RefreshSettings>

3. Save your changes.

Enabling and Disabling User-Customizable Options

This section contains step-by-step instructions for enabling and disabling particular options available from the **Properties** dialog box. The instructions are presented in the order of the tabs on which each option appears.

Important   Before modifying any elements of Config.xml that control user-customizable options, read the following section, "Understanding the Modifiable and Forcedefault Attributes." It conveys information and instructions that are crucial to editing all user-customizable options. This information is not repeated separately in the instructions for editing each option.

Understanding the Modifiable and Forcedefault Attributes

Most of the XML elements that control user-customizable options contain two attributes, **modifiable** and **forcedefault**. The values assigned to these two attributes determine if and how users can set preferences for the associated option.

The value of the **modifiable** attribute determines whether the associated option is enabled or disabled:

- Set the value of the **modifiable** attribute to **"true"** (include the quotation marks) to enable the associated option

- Set the value of the **modifiable** attribute to **"false"** (include the quotation marks) to disable the associated option

The value of the **forcedefault** attribute determines whether the ICA Client applies the user preference for the associated option persistently, or if it reverts to the server default the next time the client starts up or settings are refreshed.

- Set the value of the **forcedefault** attribute to **"true"** (include the quotation marks) to force the server default for the associated option.

- Set the value of the **forcedefault** attribute to **"false"** (include the quotation marks) to persistently apply the user preference for the associated option.

Modifying Server Tab Options

The options available on the **Server** tab let users modify the URL to the NFuse Classic server that hosts the configuration file and select from among several logon modes.

The benefit of allowing users to modify the server URL lies in the resulting ease of migrating a segment of users to a different NFuse Classic server. To migrate a segment of users to a different NFuse Classic server, you instruct the members of that segment, by email or other means, to modify the server URL from the **Server** tab. A built-in safeguard ensures that user input of valid URLs only is accepted.

Providing a choice of multiple logon modes may be necessary in environments where multiple users use the same client device but different logon modes.

If you do not want users to have access to any of these options, you can hide the **Server** tab altogether. You can revert to showing the tab at any time. For instructions about hiding and showing this tab, see "Hiding and Showing Tabs of the Properties Dialog Box" on page 53.

▶ **To disable the Server URL option**

1. Locate the **Location** element of the **ConfigurationFile** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **Location** element to **false**:
   <Location modifiable=**"false"** ...></Location>

3. Save your changes.

▶ **To enable the Server URL option**

1. Locate the **ConfigurationFile** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **Location** element to **true**:
   <Location modifiable=**"true"** ...></Location>

3. Save your changes.

▶ **To delete memorized server URLs**

The Program Neighborhood Agent memorizes the server URLs you specify in the **Server URL** text box. If you specify more than one server URL, the tab presents you with a drop-down list of memorized URLs for quick access. The most recently used URL appears at the top of the list.

See the section "Customizing the ICA Win32 Program Neighborhood Agent (Client-Side)" on page 70 for instructions about deleting memorized server URLs on individual client devices.

▶ **To enable a logon mode**

Providing a choice of multiple logon modes may be necessary in environments where multiple users use the same client device but different logon modes. Each logon mode is listed on a separate line in the **Logon** section of Config.xml. The logon mode defined at the top of the list is the logon mode the ICA Client applies the first time it is launched after installation or reinstallation.

Note   The **LogonMethod** elements do not have **modifiable** and **forcedefault** attributes.

Important   If you did not enable the pass-through authentication feature when you first installed the Program Neighborhood Agent, you must reinstall the client software before you can use the pass-through authentication logon mode.

The following variables are recognized and supported:

Variable	Description
anonymous	Allows anonymous user logon. No credentials are passed. Users can launch resources published for anonymous users only.
prompt	The Program Neighborhood Agent prompts users for a user name, password, and domain.
smartcard_prompt	Prompts for smart card pin each time users access a published resource.
smartcard_sson	Authenticates users each time they start the client, but not for subsequent access to published resources.
sson	Authenticates users based on cached Windows credentials. Users are not prompted to log on to the Program Neighborhood Agent separately.

1. Locate the **Logon** section of Config.xml.

2. To enable anonymous logon, add the following element to the **Logon** section:

    <LogonMethod>anonymous</LogonMethod>

    To enable pass-through authentication, add the following element to the **Logon** section:

    <LogonMethod>sson</LogonMethod>

    To enable smart card logon with pass-through authentication, add the following element to the Logon section:

    <LogonMethod>smartcard_sson</LogonMethod>

    To enable smart card logon without pass-through authentication, add the following element to the Logon section:

    <LogonMethod>smartcard_prompt</LogonMethod>

3. Save your changes.

Note   In conjunction with logon modes that prompt users for a password, you can allow users to save their password. This option is disabled by default. See the section "Enabling and Disabling the Save Password Option on the Logon Screen" on page 68 for instructions about enabling this option.

▶ **To disable a particular logon mode**

1. Locate the **Logon** section of Config.xml.

2. Locate and delete the **LogonMethod** element that specifies the logon mode you want to disable.

3. Save your changes.

Modifying Application Display Tab Options

The options available on the **Application Display** tab let users place links to published resources in various locations of the client device, including the Windows desktop, the **Start** menu, the Windows System Tray, and any combination thereof.

If you do not want to allow users to set preferences for these options, you can hide the **Application Display** tab altogether. You can revert to showing the tab at any time. For instructions about hiding and showing this tab, see "Hiding and Showing Tabs of the Properties Dialog Box" on page 53 for more information.

Handling of user preferences:

You control which settings users are allowed to customize and if they can do so on a per-session basis or persistently. The client queries Config.xml at startup to validate each user preference against its controlling element in Config.xml.

▶ **To disable the Show applications in Start menu check box**

1. Locate the **FolderDisplay/StartMenuDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **false**:
    <Enabled modifiable=**"false"** ...></Enabled>

3. Save your changes.

▶ **To enable the Show applications in Start menu check box**

1. Locate the **FolderDisplay/StartMenuDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **true**:
    <Enabled modifiable=**"true"** ...></Enabled>

3. Save your changes.

▶ **To force a folder for links to published resources**

This creates a folder in the **Start** menu with shortcuts to published resources. Users cannot disable or rename the folder.

1. Locate the **StartMenuDisplay** section of Config.xml.

2. Clear the value of the **root** attribute of the **RootFolder** opening tag:
    <RootFolder root=**""** ...></RootFolder>

3. Set the value of the **modifiable** attribute of the **RootFolder** opening tag to **false**:
   <RootFolder modifiable=**"false"** ...></RootFolder>

4. Set the value of the **forcedefault** attribute of the **RootFolder** opening tag to **true**:
   <RootFolder forcedefault=**"true"** ...></RootFolder>

5. Enter a name for the folder you want to create between the **RootFolder** tags:
   <RootFolder...>*NewFolderName*</RootFolder>

6. Save your changes.

▶ **To disable the Start menu folder text box**

1. Locate the **StartMenuDisplay** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **RootFolder** element to **false**:
   <RootFolder modifiable=**"false"** ...></RootFolder>

3. Save your changes.

▶ **To enable the Start menu folder text box**

1. Locate the **StartMenuDisplay** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **RootFolder** element to **true**:
   <RootFolder modifiable=**"true"** ...></RootFolder>

3. Save your changes.

▶ **To disable the Show applications in desktop folder check box**

1. Locate the **FolderDisplay/DesktopDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **false**:
   <Enabled modifiable=**"false"** ...></Enabled>

3. Save your changes.

▶ **To enable the Show applications in desktop folder check box**

1. Locate the **FolderDisplay/DesktopDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **true**:
   <Enabled modifiable=**"true"** ...></Enabled>

3. Save your changes.

▸ **To change the default name of the optional desktop folder**

The default name of the folder for published resources that users can opt to put on their desktop is **My Program Neighborhood Applications**. Follow these steps to change the folder default name:

1. Locate the **FolderDisplay/DesktopDisplay/Icon** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **Name** opening tag to **true**:

   <Name modifiable=**"true"** ...></Name>

3. Replace the folder default name, **My Program Neighborhood Applications,** with the name of your choice.

4. Save your changes.

▸ **To disable the Display applications in System Tray check box**

1. Locate the **FolderDisplay/SystemTrayMenuDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **false**:

   <Enabled modifiable=**"false"** ...></Enabled>

3. Save your changes.

▸ **To enable the Display applications in System Tray check box**

1. Locate the **FolderDisplay/SystemTrayMenuDisplay/Enabled** element.

2. Set the value of the **modifiable** attribute of the **Enabled** opening tag to **true**:

   <Enabled modifiable=**"true"** ...></Enabled>

3. Save your changes.

Modifying ICA Options Tab Options

The options available on the **ICA Options** tab let users set preferences for the window size, color depth, and sound quality of ICA sessions. Users select from a number of settings available from a pull-down menu for each option.

The ability to customize ICA sessions is provided as a convenience to users. You cannot hide this tab. However, 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, and the settings configured on the MetaFrame server apply.

▶ **To force the server defaults for all ICA options**

1. Locate the **ICA_Options** section of Config.xml.

2. Delete all lines between the **ICA_Options** tags:
   <ICA_Options>
   </ICA_Options>

3. Save your changes.

Window size options.   The window size options specify the dimensions of the ICA session on the client screen. Config.xml is preconfigured with the following settings:

• Default - applies the server-side window size configuration.

• Full Screen - Published applications and desktops are contained within an ICA session window that occupies the entire client screen, hiding other desktop content. Users cannot switch back and forth between published applications and the local desktop.

• Seamless - Published applications and desktops are not contained within an ICA session window. Each published application and desktop is resizable, as if it were physically installed on the client device. Users can switch back and forth between published applications and the local desktop.

• Specific screen dimensions, in pixels - Published applications and desktops are contained within an ICA session window that occupies the selected client screen area. Users can switch back and forth between published applications and the local desktop.

Citrix recommends that you do not modify these settings.

▶ **To remove all window size settings other than Default**

1. Locate the **ICA_Options/DisplaySize** section of Config.xml.

2. Locate and delete all elements inside the **DisplaySize** tags:
   <DisplaySize>
   </DisplaySize>

3. Save your changes.

Color depth options.   Greater color depth requires higher bandwidth. Config.xml is preconfigured with the following settings:

• Default - applies the server-side color depth configuration

• 16 Color

• 256 Color

- High Color (16 bit)

- True Color (24 bit)

Citrix recommends that you do not modify these settings.

▶ **To remove all color depth settings other than Default**

1. Locate the **ICA_Options/ColorDepth** section of Config.xml.

2. Locate and delete all elements inside the **ColorDepth** tags:
   <ColorDepth>
   </ColorDepth>

3. Save your changes.

Audio quality options. Higher sound quality requires more bandwidth. Config.xml is preconfigured with the following settings:

Default - applies the server-side audio quality configuration

High Quality

Low Quality

Medium Quality

Off

▶ **To remove a particular audio quality setting**

1. Locate the **ICA_Options/Audio** section of Config.xml.

2. Locate and delete the **Options** element that specifies the sound quality level you want to remove.

3. Save your changes.

Modifying Application Refresh Tab Options

The options available on the **Application Refresh** tab let users customize the rate at which the ICA Client queries the enumeration file for updates. The enumeration file presents users with their particular sets of published resources.

Because you may not want users to have control over the refresh rate, this 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. See "To show the Application Refresh tab" on page 57 for instructions about enabling the tab.

Config.xml is preconfigured in such a way that the Program Neighborhood Agent automatically refreshes settings when it starts up, and again at six-hour intervals. Enabling the tab makes all options on it user-customizable, unless you modify them as follows:

▸ **To disable the Refresh list when Program Neighborhood Agent starts check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **OnApplicationStart** opening tag to **false**:

   <OnApplicationStart modifiable=**"false"** ...></OnApplicationStart>

3. Save your changes.

▸ **To enable the Refresh list when Program Neighborhood Agent starts check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **OnApplicationStart** opening tag to **true**:

   <OnApplicationStart modifiable=**"true"** ...></OnApplicationStart>

3. Save your changes.

▸ **To clear the Refresh list when Program Neighborhood Agent starts check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the text between the **OnApplicationStart** tags to **false**:
   <OnApplicationStart ...>**false**</OnApplicationStart>

3. Save your changes.

▸ **To select the Refresh list when Program Neighborhood Agent starts check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the text between the **OnApplicationStart** tags to **true**:
   <OnApplicationStart ...>**true**</OnApplicationStart>

3. Save your changes.

▸ **To disable the Refresh list when remote application launched check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **OnResourceRequest** opening tag to **false**:
   <OnResourceRequest modifiable=**"false"** ...></OnResourceRequest>

3. Save your changes.

‣ **To enable the Refresh list when remote application launched check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **OnResourceRequest** opening tag to **true**:
   <OnResourceRequest modifiable=**"true"** ...></OnResourceRequest>

3. Save your changes.

‣ **To select the Refresh list when remote application launched check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the text between the **OnResourceRequest** tags to **true**:
   <OnResourceRequest ...>**true**</OnResourceRequest>

3. Save your changes.

‣ **To clear the Refresh list when remote application launched check box**

1. Locate the Request/Enumeration/Refresh section of Config.xml.

2. Set the text between the **OnResourceRequest** tags to **false**:
   <OnResourceRequest ...>**false**</OnResourceRequest>

3. Save your changes.

‣ **To disable the Refresh list on hourly interval check box**

1. Locate the **Request/Enumeration/Refresh** section of Config.xml.

2. Set the value of the **modifiable** attribute of the **Poll** opening tag to **false**:
   <Poll modifiable=**"false"** ...></Poll>

3. Save your changes.

‣ **To enable the Refresh list on hourly interval check box**

1. Locate the Request/Enumeration/Refresh section of Config.xml.

2. Set the value of the **modifiable** attribute of the **Poll** opening tag to **true**:
   <Poll modifiable=**"true"** ...></Poll>

3. Save your changes.

▶ **To clear the Refresh list on hourly interval check box**

1. Locate the **Request/Enumeration/Refresh/Poll** section of Config.xml.

2. Set the text between the **Enabled** tags to **false**:
   <Enabled>**false**</Enabled>

3. Save your changes.

▶ **To select the Refresh list on hourly interval check box**

1. Locate the **Request/Enumeration/Refresh/Poll** section of Config.xml.

2. Set the text between the **Enabled** tags to **true**:
   <Enabled>**true**</Enabled>

3. Save your changes.

▶ **To modify the Refresh frequency**

1. Locate the **Request/Enumeration/Refresh/Poll** section of Config.xml.

2. Set the text between the **Period** tags to the desired number of hours:
   <Period>*your number here*</Period>

3. Save your changes.

Enabling and Disabling the Save Password Option on the Logon Screen

If you enable the **Save Password** option and if users select a logon mode that prompts them for a password, the logon screen presents them with a check box that allows them to save the password. If they choose to save the password, it is automatically entered the next time they log on. This option is disabled by default.

▶ **To enable the Save Password option on the logon screen**

1. Locate the **EnableSavePassword** element of the **Logon** section of Config.xml.

2. Set the text between the **EnableSavePassword** tags to **true**:
   <EnableSavePassword>**true**</EnableSavePassword>

3. Save your changes.

▶ **To disable the Save Password option on the logon screen**

1. Locate the **EnableSavePassword** element of the **Logon** section of Config.xml.

2. Set the text between the **EnableSavePassword** tags to **false**:
   <EnableSavePassword>**false**</EnableSavePassword>

3. Save your changes.

Forcing and Preventing File Cleanup

By default, the Program Neighborhood Agent does not remove links to published resources when the user logs off or exits the client. This is convenient because the user can establish another session simply by double-clicking one of the links, rather than having to start the ICA Client first. However, not removing the links also leaves the associated resources accessible to unauthorized users of the client device if the authorized user selected the **Pass-through authentication** logon mode in combination with the **Save Password** option.

▸ **To force file cleanup upon logoff**

1. Locate the **FileCleanup** section of Config.xml.

2. Set the text between the **Logoff** tags to **true**:
   <Logoff>**true**</Logoff>

3. Save your changes.

Note   If a user moves, copies, or creates shortcuts to links, file cleanup does not remove the "clones."

▸ **To prevent file cleanup upon logoff**

1. Locate the **FileCleanup** section of Config.xml.

2. Set the text between the **Logoff** tags to **false**:
   <Logoff>**false**</Logoff>

3. Save your changes.

▸ **To force file cleanup upon exit**

1. Locate the **FileCleanup** section of Config.xml.

2. Set the text between the **Exit** tags to **true**:
   <Exit>**true**</Exit>

3. Save your changes.

▸ **To prevent file cleanup upon exit**

1. Locate the **FileCleanup** section of Config.xml.

2. Set the text between the **Exit** tags to **false**:
   <Exit>**false**</Exit>

3. Save your changes.

Customizing the ICA Win32 Program Neighborhood Agent (Client-Side)

This section presents general information about setting user preferences for the Program Neighborhood Agent. You set user preferences from the **Properties** dialog box, which is available from the **Program Neighborhood Agent** menu in the Windows System Tray. For more detailed information, see the online Help for the Program Neighborhood Agent.

Configuring the Server URL

The Program Neighborhood Agent requires the URL to a configuration file (Config.xml) on the NFuse Classic server. This configuration file contains the information the Program Neighborhood Agent needs so you can run remote content on your local computer.

Your system administrator may prompt you at times to change the server URL. Do not attempt to change the server URL unless you are directed to do so.

▶ **To change the server URL**

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 as directed in the dialog box that appears. Enter the 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, or click **Cancel** to cancel the operation.

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.

3. Click **Change**.

4. Click the down arrow to view the entire list of memorized server URLs.

5. Right-click the URL you want to delete and select **Delete** from the menu that appears.

6. Click **Update**.

7. Click **OK**.

Selecting a Logon Mode

Your installation of the Program Neighborhood Agent may offer multiple logon modes. Consult with your system administrator to find out which logon mode to use.

▸ **To select a logon mode**

1. In the Windows System Tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.

2. On the **Server** tab, select a logon from the **Logon mode** menu.

3. Click **OK** to apply your selection and close the Program Neighborhood Agent **Properties** dialog box.

Creating Shortcuts to Published Resources

You can place links to published resources in your **Start** menu, on your Windows desktop, and in the **Program Neighborhood Agent** menu in the Windows System Tray.

▸ **To configure shortcuts to published resources**

1. In the Windows System Tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.

2. On the **Application Display** tab, choose from the following options:

   Show applications in Start menu – Select this option to place links to published resources in the Windows **Start** menu. If you want to group the links in a folder, enter the name of the folder in the text box below this option.

   Show this folder in Programs submenu – Select this option to place the customized folder in the Programs submenu of the **Start** menu.

   Show applications in desktop folder – Select this option to place links to published resources in a folder on the desktop. Enter a name for the folder in the text box below this option.

   Display applications in System Tray – Select this option to place links to published resources in the **Applications** submenu of the **Program Neighborhood Agent** menu in your Windows System Tray.

3. Click **OK** to apply the selections and close the Program Neighborhood Agent **Properties** dialog box.

Installing and Configuring the ICA Win32 Web Client
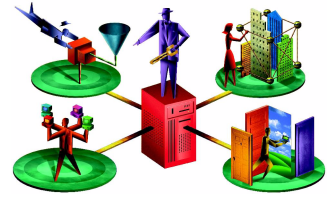
This chapter explains how to install and configure the ICA Win32 Web Client. The following topics are covered:

- Overview of the ICA Win32 Web Client

- System Requirements

- Configuring the ICA Win32 Web Client for Silent User Installation

- Installing the ICA Win32 Web Client

Overview of the ICA Win32 Web Client

Use the ICA Win32 Web Client if you want users to access your published resources from within their familiar Web browser. This ICA Client requires the use of NFuse Classic or ALE (Application Launching and Embedding); see "Application Launching and Embedding" on page 131.

The Web Client is available as a self-extracting executable. At approximately 1.8MB in size, this package is significantly smaller than the other ICA Win32 Clients. The smaller size allows quicker downloads and installation. You can configure the ICA Win32 Web Client for silent user installation.

ICA Win32 Web Client Features

The ICA Win32 Web Client supports the following features:

- User-to-user shadowing

- Smart card support

- Content redirection

- Enhanced content publishing support

- Roaming User Reconnect

- Support for SSL/TLS encryption of ICA session data

- Support for Citrix NFuse Classic 1.7 and Enterprise Services for NFuse

- Support for Citrix Secure Gateway 1.1

- Enhanced Internet proxy support

- Auto Client Reconnect

- Novell Directory Services support

- DNS name resolution support

- Extended parameter passing

- Seamless windows

- Client device mapping

- Sound support

- TCP/IP+HTTP server location

- Wheel mouse support

- Multiple monitor support

- Panning and scaling

- Per-user time zone support

- Windows clipboard integration

- Low bandwidth requirements

- SpeedScreen Latency Reduction

- Disk caching and data compression

System Requirements

To run the ICA Win32 Web Client, client devices must meet the following
requirements:

- Standard PC architecture, 80386 processor or greater as required for the
  operating system and Web browser.

- Windows 9x, Windows Me, Windows 2000, Windows XP, or Windows NT
  3.51 or later.

- 8MB RAM or greater for Windows 9x, 16MB RAM or greater for Windows NT
  3.51 or 4.0, 32MB or greater for Windows Me and Windows 2000, and 128MB
  RAM or greater for Windows XP.

- Internet Explorer Version 5.0 or later, or Netscape Navigator or Communicator Version 5.0 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 disk space.

- Windows-compatible sound card for sound support (optional).

- For network connections to the MetaFrame server, a network interface card (NIC) and the appropriate network transport software are required. The ICA Win32 Web Client supports the TCP/IP network transport only.

Configuring the ICA Win32 Web Client for Silent User Installation

Installing the ICA Win32 Web Client requires minimal user interaction. A typical installation presents the user with the following:

1. An initial prompt informs the user that the Citrix ICA Win32 Web Client is about to be installed. The user clicks **Yes** to continue with Setup or **No** to stop Setup.

2. A Citrix License Agreement. The user clicks **Yes** to accept the agreement or **No** to reject the agreement. If the user clicks **No**, Setup stops.

3. An indication that Setup is copying files to the client device. By default, the ICA Win32 Web Client is installed in the Program Files\Citrix\Icaweb32 directory.

4. A message box informs the user that the Citrix ICA Web Client was installed successfully. The user must click **OK** to clear the message.

5. If the user is running Netscape Navigator, the user must restart the browser.

You can further limit user interaction with Setup by suppressing the appearance of the initial user prompt (Step 1 above) and of the Citrix License Agreement (Step 2 above). To do this, you need to extract the ICA Client files from Ica32t.exe, edit the Ctxsetup.ini file, and repackage the files for distribution. See the following section for instructions.

Important   You can extract the client files from Ica32t.exe with any standard compression utility. However, you must use commercially available software to repackage the client files for distribution to your users.

▶ **To configure the ICA Win32 Web Client for silent user installation**

1. Extract the ICA Client files from Ica32t.exe using your preferred compression utility. This installer package is located in the following directory (substitute *language* with the language of the ICA Client software) of the Component CD included in your MetaFrame XP, Feature Release 2 media pack:

   Icaweb*language*\\ica32

   where *language* is one of:

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

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. Change the value of the setting from 1 to **0**.

5. Save the file and exit the text editor.

6. Repackage the client files for distribution to your users.

Installing the ICA Win32 Web Client

The ICA Win32 Web Client self-extracting executable, Ica32t.exe, is located in the following directory (substitute *language* with the language of the ICA Client software) of the Component CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\\ica32

where *language* is one of:

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

If you are using the ICA Win32 Web Client with NFuse Classic, see the *NFuse Classic Administrator's Guide* for information about deploying this ICA Client with NFuse Classic.

▸   **To install the ICA Win32 Web Client**

1. Run Ica32t.exe.

2. The initial prompt informs you that the Citrix ICA Win32 Web Client is about to be installed. Click **Yes** to continue with Setup or **No** to stop Setup.

3. The Citrix License Agreement appears. Click **Yes** to accept the agreement or **No** to reject the agreement. If you click **No**, Setup stops.

4. You are informed that Setup is copying files to the client device. The default file location for the ICA Win32 Web Client is Program Files\Citrix\icaweb32.

5. You are informed that the Citrix ICA Web Client was installed successfully. Click **OK** to clear the message.

6. If you are running Netscape Navigator, you must restart the browser.

Installing and Configuring the ICA Win32 Program Neighborhood Client

This chapter explains how to install and configure the ICA Win32 Program Neighborhood Client. The following topics are covered:

- Overview of the ICA Win32 Program Neighborhood Client

- System Requirements

- Installing the ICA Win32 Program Neighborhood Client

- Starting the ICA Win32 Program Neighborhood Client

- Configuring the ICA Win32 Program Neighborhood Client

Overview of the ICA Win32 Program Neighborhood Client

Use the ICA Win32 Program Neighborhood Client if you do not want to use NFuse Classic to deliver published resources. The Program Neighborhood Client has its own user interface — the Program Neighborhood window — from which users browse for application sets or create custom ICA connections to MetaFrame servers or published applications.

This ICA Client can also be used with NFuse Classic and ALE (Application Launching and Embedding); see "Application Launching and Embedding" on page 131. If you are planning to use NFuse Classic and have not yet deployed any ICA Clients, use the ICA Win32 Program Neighborhood Agent or the ICA Win32 Web Client.

ICA Win32 Program Neighborhood Client Features

The ICA Win32 Program Neighborhood Client provides the following features:

- Program Neighborhood user interface
- User-to-user shadowing
- Smart card support
- Content redirection
- Roaming User Reconnect
- Support for SSL/TLS encryption of ICA session data
- Support for Citrix NFuse Classic 1.7 and Enterprise Services for NFuse
- Support for Citrix Secure Gateway 1.1
- Enhanced Internet proxy support
- Auto Client Reconnect
- Published content support
- Novell Directory Services support
- DNS name resolution support
- Extended parameter passing
- TAPI support
- Seamless windows
- Client device mapping
- Sound support
- Dialing prefixes
- Windows Installer Packages
- Client Auto Update
- Windows clipboard integration
- Low bandwidth requirements
- SpeedScreen Latency Reduction
- Disk caching and data compression
- Business recovery
- TCP/IP+HTTP server location
- Wheel mouse support

- Multiple monitor support
- Pass-through authentication
- Panning and scaling
- Per-user time zone support

System Requirements

To run the ICA Win32 Program Neighborhood Client, 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 2000, Windows XP, or Windows NT 3.51 or later
- 8MB RAM or greater for Windows 9x, 16MB RAM or greater for Windows NT 3.51 or 4.0, 32MB or greater for Windows Me and Windows 2000, and 128MB RAM or greater for Windows XP
- To use the ICA Win32 Program Neighborhood Client with a Web browser, Internet Explorer Version 5.0 or later, or Netscape Navigator or Communicator Version 5.0 or later
- Microsoft mouse or 100% compatible pointing device
- VGA or SVGA video adapter with color monitor
- High-density 3.5-inch disk drive (optional) and available hard disk space
- Windows-compatible sound card for sound support (optional)
- For serial (dial-up) connections to the MetaFrame server, an internal modem or serial port and external modem using a 16550 Universal Asynchronous Receiver/Transmitter (UART) is recommended
- For network connections to the MetaFrame server, a network interface card (NIC) and the appropriate network transport software are required. Supported connection methods and network transports are:
  - TCP/IP+HTTP
  - SSL/TLS+HTTPS
  - TCP/IP
  - NetBIOS
  - IPX
  - SPX

For information about configuring the ICA Win32 Clients to use SSL or TLS to secure communications, see "Configuring and Enabling the Clients for SSL and TLS" on page 148.

Installing the ICA Win32 Program Neighborhood Client

You can install the ICA Win32 Program Neighborhood Client using one of the following packages:

- Ica32.msi – a Windows Installer package for use with Windows 2000 Active Directory Services or Microsoft Systems Management Server, approximately 5.2MB in size

- Ica32.exe – a self-extracting executable, approximately 3.7MB in size

---

Note   For a discussion of methods for deploying ICA Win32 Client software to users, see "Preconfiguring and Deploying the ICA Win32 Clients" on page 31 of this guide, and the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

Installing the ICA Win32 Program Neighborhood Client with the Windows Installer Package

You can distribute the Program Neighborhood Client Windows Installer package (Ica32.msi) with Microsoft Systems Management Server or Windows 2000 Active Directory Services.

This package is located in the following directories (substitute *language* with the language of the ICA Client software) on the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\\ica32

Icainst*language*\\ica32

where *language* is one of:

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

> **Note**   To install the ICA 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 ICA 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 an ICA Win32 Client that was installed with a Windows Installer
package, run the **Add/Remove Programs** utility from the Control Panel, or run the
installer package again, selecting the **Remove** option.

Configuring the Windows Installer Package for Silent User Installation

You can configure the Program Neighborhood Client Windows Installer package
for "silent" user installation. Windows Installer informs the user when the client
software is successfully installed. The user must clear the Windows Installer
message box.

▸ **To configure the Program Neighborhood Client Windows Installer package**
  for silent user installation

1. Enter the command line

   msiexec /I <MSI_Package> /qn+ [Key=Value]…

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

2. You can set the following keys:

   INSTALLDIR=<Installation_Path>, where <Installation_Path> is the path to
   the directory where the ICA Client software is installed. By default, the ICA
   Client software is installed in the Program Files\Citrix\ICA Client directory.

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

   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 ICA Client software. The default value
   is **Citrix ICA Client**.

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

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

   SSOnCredentialType=<**NT**>. Include this parameter if you want to use
   Windows NT credentials with pass-through authentication on client devices that
   have the Novell Client installed.

DEFAULT_NDSCONTEXT=<Context1 [,…]>.Include this parameter if you want to set a default context for 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
```

Installing the ICA Win32 Program Neighborhood Client with the Self-Extracting Executable

You can distribute the self-extracting executable (Ica32.exe) to your users for direct installation on client devices.

This executable is located in the following directory (substitute *language* with the language of the ICA Client software) on the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

Icaweb*language*\\ica32

where *language* is one of:

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

Configuring the Self-Extracting Executable for Silent User Installation

You can configure numerous settings before you deploy the Program Neighborhood Client software to your users. This allows users to install the client and begin using it immediately, without having to configure settings.

You can customize many Program Neighborhood Client 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 ICA Win32 Clients* on the Citrix Web site at http://www.citrix.com/support; select **Product Documentation**.

Important   You can use any standard compression utility to extract the client files from the packaged executable. However, you must use commercially available software to repackage the client files for distribution to your users.

When the ICA Win32 Program Neighborhood Client 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 the ICA Win32 Program Neighborhood Client 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 user installation**

1. Extract the client file set from Ica32.exe using your preferred compression utility software. These files are located in the following directory (substitute *language* with the language of the ICA Client software) on the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

   Icaweb*language*\\ica32

   where *language* is one of:

   - En (English)

   - Fr (French)

   - De (German)

   - Ja (Japanese)

   - Es (Spanish)

2. Open the files containing the customizable settings in a text editor. These files are:

   Appsrv.src

   Module.src

   Pn.src

   Wfclient.src

3. Change the parameters to reflect your desired settings. For definitions of parameters in the client .Ini files, see the *Configuration Guide for the ICA Win32 Clients* on the Citrix Web site at http://www.citrix.com/support; select **Product Documentation**.

4. Repackage the client files for distribution to your users.

▸   **To install the Program Neighborhood Client with the self-extracting executable**

1. Make sure the client device is properly configured and cabled. Make sure any previous installations of the Citrix ICA Client (including the ICA Connection Center, whose icon appears in the Windows system tray when active) are not running.

2. If you are installing from disks:

   Insert ICA Win32 Client Setup disk number 1 in drive A (or other appropriate drive) of the client device. For Windows 9x, Windows Me, Windows 2000, and Windows NT 4.0 client devices, click **Start** > **Run a:\setup**. For Windows NT 3.5x client devices, on the **File** pull-down menu of Program Manager run **a:\setup**.

   For information about creating ICA Client installation disks, see "Creating ICA Client Installation Disks" on page 33.

   If you are installing from a MetaFrame server:

   Run **Setup.exe**, located in the following directory on your MetaFrame server:

   %SystemRoot%\System32\Clients\Ica\Ica32\Disks\Disk1

   If you are installing the client from the Components CD included in your MetaFrame XP, Feature Release 2 media pack:

   Run **Ica32.exe**, located in the following directory (substitute *language* with the language of your ICA Client software):

   Icaweb*language*\ica32

   where *language* is one of:

   En (English)

   Fr (French)

   De (German)

   Ja (Japanese)

   Es (Spanish)

3. The **Welcome** screens appear. Read the information on these screens and click **Next**.

4. The Citrix License Agreement appears. Read the license agreement and click **Yes** if you agree to the terms of the agreement.

   Setup searches your client device for previously installed versions of the ICA Win32 Program Neighborhood Client.

5. If Setup does not detect a previous installation of the ICA Win32 Program Neighborhood Client, go to Step 6.

If Setup detects a previous installation of the ICA Win32 Program Neighborhood Client, the **Choose Installation Type** dialog box appears:
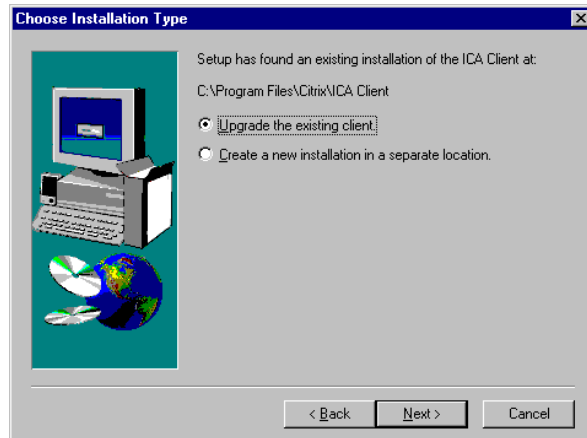
The **Choose Installation Type** dialog box lets you choose to upgrade the existing client or to create a new and separate installation of the Program Neighborhood Client in a new location. The default value is **Upgrade the existing client**.

Click **Next**.

6. The **Choose Destination Location** dialog box appears:
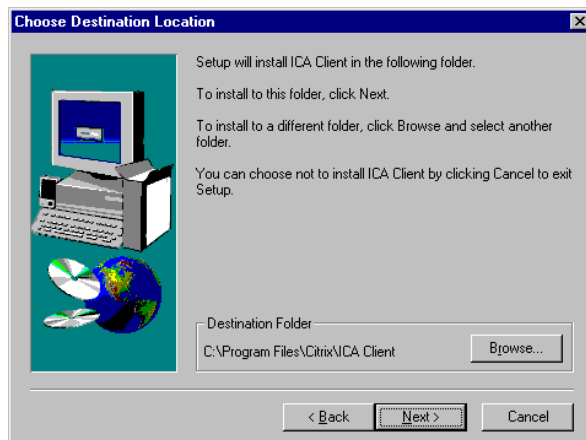
You can change the displayed path if desired by clicking **Browse**.

Click **Next** to accept the displayed path and continue installation.

7. The **Select Program Folder** dialog box appears:

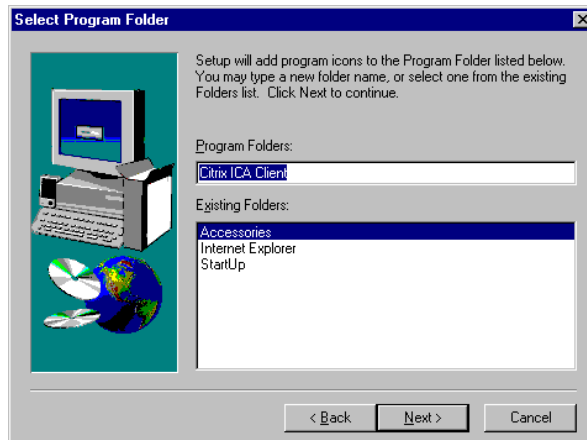You can choose to use the default Citrix ICA Client folder, specify the name of a new program folder, or add the ICA Win32 Client icons to an existing folder. The program folder you specify is created if it does not already exist.

Click **Next** to continue.

8. The **ICA Client Name** dialog box appears:
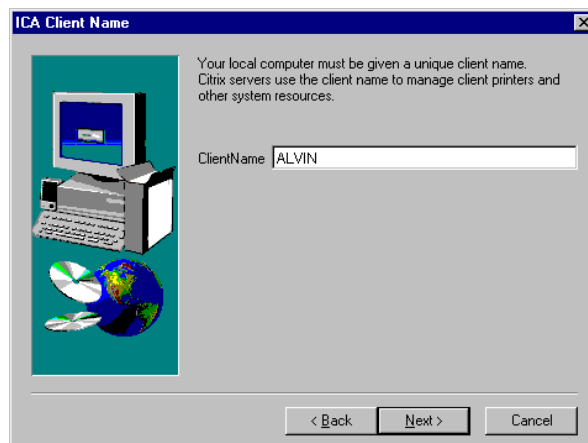
Specify a unique client name for your client device. MetaFrame servers use the client name to manage client printers and other system resources. If you do not assign a unique machine name to each client device, device mapping and application publishing may not operate correctly.

Click **Next** to continue.

9. The **Select Desired Features** screen appears.

   Select **Yes** if you want the Program Neighborhood Client to access your local Windows user name, password, and domain information. If you select **Ye**s, you are not prompted to log on to the Program Neighborhood Client separately.

   When you are done, click **Next** to continue. A progress window appears, displaying the file names as they are copied to your hard drive.

10. If you are installing from a floppy disk, go to Step 11.

   If you are not installing from a floppy disk, a dialog box informs you that the Program Neighborhood Agent was successfully installed on the computer. Click **Finish**.

11. If you are installing from a floppy disk, the **Setup Needs the Next Disk** dialog box appears. Remove the first ICA Win32 Client disk from drive A (or other appropriate drive) and insert the second disk. Click **OK**.

12. When the Citrix ICA Client finishes copying the program files, the **Information** dialog box appears. Click **OK** to exit this window.

   The Citrix ICA Client program group appears on the desktop.

Starting the ICA Win32 Program Neighborhood Client

▶   **To start the Program Neighborhood Client**

1. Double-click the Program Neighborhood Client icon on the desktop to open the **Program Neighborhood** window. If you specified a default application set for this user, this window contains all the resources the user can run. If no default is specified, a list of application sets appears.

2. Double-click the application set to access. A logon dialog box appears.

3. Enter a valid user name, domain, and password.

Configuring the ICA Win32 Program Neighborhood Client

This section explains how to configure the Program Neighborhood Client using the Program Neighborhood user interface.

Configuring Network Protocol and Server Location

With the Citrix ICA Win32 Program Neighborhood Client, users can connect to a MetaFrame server in the following ways:

- Establishing a custom ICA connection by dialing into a MetaFrame server using a modem installed on the client device. This method uses a serial connection to a MetaFrame server.

- Establishing a custom ICA connection over a direct serial cable. This method uses a serial connection to a MetaFrame server.

- Over the local or wide-area network connection between the client device and the MetaFrame server. This method uses one of the following network protocols:

    - TCP/IP+HTTP

    - SSL/TLS+HTTPS

    - TCP/IP

    - IPX

    - SPX

    - NetBIOS

You can also use Microsoft's Remote Access Service (RAS) or Dial-Up Networking (DUN) in combination with the Citrix ICA Client to connect to a MetaFrame server. For this type of connection, the client device must meet the following requirements in addition to those specified at the beginning of this chapter:

- 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 MetaFrame server

Specifying the Network Protocol for ICA Browsing

The network protocol setting allows you to control the way the ICA Client searches for MetaFrame servers and how it communicates with them.

This section discusses the TCP/IP+HTTP, SSL/TLS+HTTPS, and TCP/IP protocols and their implementation in a MetaFrame server farm. For additional information about configuring MetaFrame XP server farms for ICA browsing, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

Using TCP/IP+HTTP Network Protocol for ICA Browsing

If you select **TCP/IP+HTTP** as the network protocol from the Program Neighborhood Client's **Settings** menu, the ICA Client uses the HTTP protocol to search for MetaFrame servers. Select this protocol when using the ICA Client over the Internet or through a firewall or proxy server. Select **SSL/TLS+HTTPS** to use SSL/TLS-secured communications.

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, which 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 ICA Clients to locate servers that are not on the same subnet.

If you select **TCP/IP+HTTP** as the network protocol, specify servers to contact for ICA browsing by entering IP addresses or DNS names of MetaFrame servers in the **Address List** box in the Program Neighborhood Client.

If you select **TCP/IP+HTTP** as the network protocol and specify MetaFrame servers in the **Address List** box, the ICA Client communicates with the Citrix XML Service on a specified server for ICA browsing.

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 ICA Clients.

If you select **TCP/IP+HTTP** as the network protocol, Auto-Locate does not use a broadcast to determine the nearest farm server, but rather attempts to resolve the host name "ica" to a MetaFrame server. Therefore, if you want Auto-Locate functionality, the TCP/IP+HTTP protocol does not increase network traffic with ICA Client broadcasts.

Tip   You can configure the ICA Clients' DNS server 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 ICA Clients.

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

If you select **TCP/IP+HTTP** as the network protocol, communication between the client and XML Service consists of XML-formatted data in HTTP packets.

Communicating with the Citrix XML Service

Citrix XML Service is installed by default on all MetaFrame XP servers. It is also installed with Feature Release 1 for MetaFrame 1.8.

If you select **TCP/IP+HTTP** as the network protocol, the XML Service communicates published application information to clients using HTTP protocol and XML data. The XML Service also communicates published application information to NFuse Classic servers.

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

With NFuse Classic, for example, a user connects to an NFuse Classic Web page using a Web browser. The XML Service provides a list of available applications to the NFuse Classic server. The NFuse Classic server displays the available applications on the user's personalized application Web page.

Using SSL/TLS+HTTPS Network Protocol for ICA Browsing

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

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 MetaFrame servers. See the *MetaFrame Administrator's Guide* for Windows or UNIX for information about configuring the MetaFrame server to use SSL/TLS.

Using TCP/IP Network Protocol for ICA Browsing

If you select **TCP/IP** as the network protocol and **(Auto-Locate)** appears in the **Address List** box of the **Settings** menu, ICA Clients send UDP broadcasts to the ICA Browser service on port 1604 to locate MetaFrame servers and published applications. Select this option if all of the MetaFrame servers and clients are located on the same network.

By default, MetaFrame XP server farms operating in native mode do not respond to ICA 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 MetaFrame XP servers or published applications in the server farm.

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 ICA 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 MetaFrame servers in the **Address List** box in the Program Neighborhood Client. You must do this if the ICA Client is not on the same subnet as a data collector.

Configuring Connections to MetaFrame Servers and Published Applications

This section describes how to configure connections to MetaFrame servers and published applications. The Program Neighborhood Client offers two connection methods:

- Connecting to MetaFrame servers and published applications using application sets

- Connecting to MetaFrame servers and published applications using custom ICA connections

Configuring TCP/IP+HTTP Server Location

You can retrieve MetaFrame server and published application information across a firewall that does not allow UDP broadcasts by selecting **TCP/IP+HTTP** or **SSL/ TLS+ HTTPS** server location. The ICA Win32 Program Neighborhood Client uses TCP/IP+HTTP as the default network protocol.

▸ **To configure TCP/IP+HTTP server location**

1. From the **Network Protocol** drop-down list, select **TCP/IP+HTTP**.

2. Click **Add** to display the **Add Server Location Address** dialog box.

3. Enter the name or IP address of a MetaFrame server and a recognized port number (the default is port 80) and click **OK**.

   Note   If you do not enter an IP address, you must have a MetaFrame server on your network mapped to the default name of "ica." TCP/IP+HTTP server location does not support the **[Auto-Locate]** function.

4. The specified server responds with a list of all servers and published resources in its server farm.

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

Configuring SSL/TLS+HTTPS Server Location

If you configured your MetaFrame servers to accept SSL/TLS-secured connections, you can enable SSL/TLS on the Program Neighborhood Client.

▸ **To configure SSL/TLS+HTTPS server location**

1. From the **Network Protocol** drop-down list, select **SSL/TLS+HTTPS**.

2. Click **Add** to display the **Add Server Location Address** dialog box.

3. Enter the fully qualified domain name of the MetaFrame server with the digital server certificate and verify that port 443 is listed as the default port.

4. Click **OK**.

> **Important**   SSL/TLS+HTTPS server location retrieves information only on a per-server farm basis. To retrieve information from more than one server farm, you must configure SSL/TLS+HTTPS 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.

For more information about configuring connections, see "Configuring Connection Properties" on page 97. For more information about using SSL to secure client to server communication, see "Configuring and Enabling the Clients for SSL and TLS" on page 148.

Configuring Server Location and Business Recovery

Server location (also called *server browsing*) provides a method for a user at a network-connected ICA Client to view a list of all MetaFrame servers on the network that have ICA connections configured for that network protocol, and a list of all published applications. You can specify a separate server location for each network protocol.

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

1. The ICA Client broadcasts a "Get Nearest MetaFrame server" packet. The first MetaFrame server to respond returns the address of the master ICA Browser, which is used in the next step.

2. The ICA 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 MetaFrame servers on the network 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 MetaFrame server that functions as the master browser.

Business recovery provides consistent connections to published applications in the event of a master ICA Browser server disruption. You can define up to three groups of MetaFrame servers to which you want to connect: a primary and two backups. Each group can contain from one to five servers. When you specify a server group for your client, the client attempts to contact all the servers within that group simultaneously and the first server to respond is the one to which you connect. The client broadcasts only if you select **(Auto-locate)** from the address list.

Using Application Sets and 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. Applications 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 a MetaFrame for UNIX server. To connect to an application published on a MetaFrame for UNIX server, you *must* use a custom ICA connection.

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

• An existing MetaFrame 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 MetaFrame for UNIX Operating Systems server

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

Adding Application Sets and Custom ICA Connections

To locate additional application sets you can access, or to add a custom ICA connection, use the **Find New Application Set** or the **Add New ICA Connection** wizards, respectively.

▶ **To find a new application set**

1. Double-click the Find New Application Set icon in the Program Neighborhood window.

2. Follow the instructions in the Find New Application Set wizard.

▸ **To add a custom ICA connection**

1. Double-click the Custom ICA Connections icon. The **Custom ICA Connections** window opens.

2. Double-click the Add ICA Connection icon.

3. Follow the instructions in the Add New ICA Connection wizard.

For details about the settings in the Find New Application Set and Add New ICA Connection wizards, see the wizards' application help.

Configuring Application Sets and Custom ICA Connections

The following procedures describe how to configure the properties and settings of application sets and custom ICA connections:

- Configuring connection properties

- Configuring default options

- Configuring a logon mode

- Configuring general settings

- Configuring bitmap caching

- Configuring hotkeys

- Configuring event logging

Configuring Connection Properties

▸ **To configure connection properties**

1. Start the Program Neighborhood Client.

2. If you are configuring an application set:

   Select the application set and click **Settings** on the Program Neighborhood Client toolbar.

   If you are configuring a custom ICA connection:

   Select the custom ICA connection you want to configure and click the **Properties** button in the Program Neighborhood Client toolbar to display the **Properties** dialog box.

3. Click the **Connection** tab to display the **Connection** page.

   From the **Connection** page, you can configure the following:

Connection Type. Choose a connection type. Select **Local Area Network** to connect to the MetaFrame server over a local network that covers a confined geographical area (such as an office building or complex). Select **Wide Area Network** to connect to the MetaFrame server over a network that covers a wide geographical area.

If you are configuring a custom ICA connection, you can select either **Server** or **Published Application.** When you select **Server**, this field specifies the MetaFrame server to run the published application.

For more information about the options on this tab, click **Help**.

Configuring Default Options

▶ **To configure default options**

1. Start the Program Neighborhood Client.

2. If you are configuring an application set:

    Select the application set and click **Settings** in the Program Neighborhood Client toolbar.

    If you are configuring a custom ICA connection:

    Right-click in the custom ICA connection window and select **Custom Connections Settings**.

3. Click the **Default Options** tab to display the **Default Options** page. For custom ICA connections: Any options configured in this dialog box are applied to *all* custom ICA connections. To override these default options on an individual custom ICA connection, select the ICA connection and click **Properties** on the Program Neighborhood Client toolbar. Select the **Options** tab.

4. From the **Options** and **Default Options** pages, you can configure the following:

    Use data compression. Data compression reduces the amount of data that needs to be transferred but requires additional processor resources to compress and decompress the data. If your connection is bandwidth-limited, enabling data compression increases performance.

    Use disk cache for bitmaps. Bitmap caching to disk stores commonly-used graphical objects such as bitmaps in a local cache on the client's hard disk space. If your connection is bandwidth-limited, enabling disk caching increases performance. If your client is on a high-speed LAN, you do not need disk caching. Dial-in connections have disk caching enabled by default.

    Queue mouse movements and keystrokes. Queuing causes the client to send mouse and keyboard updates less frequently to the MetaFrame server. Select this option to reduce the number of network packets sent from the ICA Client to the MetaFrame server. If you do not select this option, the session is more responsive to keyboard and mouse movements. Selecting this option improves performance if you dial in to RAS and then use a network to connect.

Turn off desktop integration for this application set. You can configure the Program Neighborhood Client to create desktop shortcuts and add items to the **Start** menu for published applications. If users do not want published applications sent directly to the desktop, they must select this check box.

Enable sound. Select this check box to enable sound support. The client device must have a compatible sound card installed. Published applications can then play sounds on the client.

Select one of the following values for **Quality**:

- **High**. This setting is recommended only for connections where bandwidth is plentiful and sound quality is important. This setting allows clients to play a sound file at its native data rate. Sounds at the highest quality level require about 1.3Mbps of bandwidth to play clearly. Transmitting this amount of data can result in increased CPU utilization and network congestion.

- **Medium**. This setting is recommended for most LAN-based connections. This setting causes any sounds sent to the client to be compressed to a maximum of 64Kbps. This compression results in a moderate decrease in the quality of the sound played on the client device. The host CPU utilization will decrease compared with the uncompressed version due to the reduction in the amount of data being sent across the wire.

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

Encryption Level. Select the level of encryption for the ICA connection. The default level is **Basic**. Select **128-bit for Login Only** to use encryption during authentication.

The MetaFrame server must be configured to allow the selected encryption level or greater. To enable encryption levels higher than **Basic**, the MetaFrame server must support RC5 encryption. This support is included with SecureICA Services, MetaFrame 1.8 Feature Release 1, and MetaFrame XP.

Note   Selecting an encryption level higher than **Basic** disables automatic logon to the MetaFrame server.

SpeedScreen Latency Reduction. SpeedScreen Latency Reduction is a collective term used to describe the functionality that helps enhance user experience on slower network connections. Latency reduction is available only if you are connecting to a server that is configured and licensed for latency reduction.

For slower connections (for example, if you are connecting over a WAN or a dial-up connection), select **On** to decrease the delay between user input and screen display. Choose either **Mouse Click Feedback** or **Local Text Echo**.

For faster connections (for example, if you are connecting over a LAN), select **Off**.

If you are not certain of the connection speed, set mode to **Auto** to turn latency reduction on or off depending on the speed of the connection. You can override Auto mode using the **Toggle Latency Reduction** hotkey.

Window Properties. Specify the number of colors displayed in the application's window in the **Windows Colors** field.

Use Server Default (for application sets). To use the server-configured default settings for the properties, make sure this check box is selected. To change the settings, clear this check box and choose new settings.

Use Custom Default (for custom ICA connections). To override the default options, clear this check box.

Specify the window size that a published application runs in the **Window Size** field.

If you are connecting to a published application, you can select **Seamless Windows** to run the application on your local desktop in a separate, seamless window.

Configuring a Logon Mode

▶ **To configure a logon mode for application sets and custom ICA connections**

1. Start the Program Neighborhood Client.

2. If you are configuring an application set:

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

    If you are configuring a custom ICA connection:

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

3. On the **Logon Information** tab, select the logon mode you want to configure for this client:

- **Local user** prompts users for authentication when they establish a connection.

  Local user / Pass-through authentication enables pass-through authentication based on the user's Windows desktop credentials.

- **Smart card** authenticates users based on a smart card and Personal Identification Number (PIN).

  Smart card / Pass-through authentication enables pass-through authentication based on the smart card and the smart card PIN. This logon mode requires a smart card to be present or inserted in the smart card reader at logon time.

- **User-specified credentials** authenticates users based on credentials (user name, password, and domain information) to be specified here.

  User-specified credentials / Save password. This check box is selected by default. Clear the check box if you do not want users to save their passwords in clear text. For instructions about removing the check box from the interface altogether, see "Preventing Users From Saving Passwords" on page 103.

Note   To connect to a MetaFrame server or published application that is configured to use Novell Netware Directory Services (NDS), enter the user's NDS distinguished name in the **User Name** field and the password in the **Password** field. Leave the **Domain** field blank.

---

4. Click **OK**.

Reducing the Number of Required Logons

By default, users are required to enter their credentials each time they launch a published application or connect to a MetaFrame server desktop. You can configure the Program Neighborhood Client to reduce the number of times users have to enter credentials using one of the following methods:

- Enabling pass-through authentication
- Caching credentials with application set information

You can enable pass-through authentication to pass the user's local Windows desktop credentials to the MetaFrame server. This eliminates the need for multiple authentications.

If you enable pass-through authentication, all existing application sets are automatically configured to use pass-through authentication. However, existing custom ICA connections are automatically configured not to use pass-through authentication. You must, therefore, enable pass-through authentication for each custom ICA connection for which you want to use this logon mode.

Enabling Pass-Through Authentication

To enable pass-through authentication, you must complete the following tasks:

- Enable the feature at the machine level
- Enable the feature at the user level

▶ **To enable pass-through authentication at the machine level**

1. Log on as an administrator.
2. Start the Program Neighborhood Client.
3. Go to **Tools** > **ICA Settings**.
4. Click the **General** tab and select the **Pass-Through Authentication** check box.
5. Click **OK**.

When this feature is turned on at the machine level, each user can enable it at the user level in the following manner.

▶ **To enable pass-through authentication at the user level**

1. Log on as a user.
2. Start the Program Neighborhood Client.
3. Go to **Tools** > **ICA Settings**.
4. Click the **General** tab and select the **Use local credentials to log on** check box.
5. Click **OK**.

Enabling pass-through authentication at the user level makes this the default configuration for all existing and new application sets. Follow the instructions below for enabling new and existing custom ICA connections to use this feature.

▶ **To enable pass-through authentication for new custom ICA connections**

When creating a new custom ICA connection, enable pass-through authentication by selecting **Use local User name and Password** in the Add ICA Connection wizard.

If you want to prevent users from caching their passwords in clear text, see "Preventing Users From Saving Passwords" on page 103.

‣ **To enable pass-through authentication for existing custom ICA connections**

1. Start the Program Neighborhood Client.

2. Select a custom ICA connection for which you want to enable pass-through authentication.

3. On the Program Neighborhood Client toolbar, click the **Properties** button.

4. Click the **Logon Information** tab.

5. Select **Local user** to authenticate using the local Windows desktop credentials; select **Pass-through authentication** to enable pass-through authentication based on the same local Windows desktop credentials.

    -Or-

    Select **Smart card** to authenticate using a smart card and Personal Identification Number (PIN); select **Pass-through authentication** to enable pass-through authentication based on the smart card and the smart card PIN.

    -Or-

    Select **User-specified credentials** to authenticate using credentials other than local user or smart card credentials.

    The **Save password** check box is selected by default. Clear the **Save password** check box if you do not want to save the password in clear text.

6. Click **OK**.

7. Repeat Steps 1 through 6 for each ICA connection for which you want to enable pass-through authentication.

Preventing Users From Saving Passwords

You can remove the **Save Password** check box from an application set's logon screen and from the **Login Information** tab of the **Settings** dialog box. Removing the check box prevents users from saving their passwords for application sets. You can prevent users from saving their passwords for all application sets or specific application sets.

Use the following table to determine which files you need to edit, then follow the instructions below.

To disable password saving for all application sets	To disable password saving for particular application sets
%User Profile%\Application Data\ICAClient\Appsrv.ini	%User Profile%\Application Data\ICAClient\Pn.ini

▶ **To prevent users from saving their passwords for *all* application sets**

1. Exit the Program Neighborhood Client 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 section named **[WFClient]**.

4. Add the following line to the list of parameters and values in the **[WFClient]** section:

    NoSavePwordOption=On

    If the parameter already exists, make sure its value is set to **On**.

5. Save the file and exit the text editor.

6. Repeat Steps 1 through 5 for additional users.

7. Start the Program Neighborhood Client.

Adding this parameter and setting it to **On** prevents users from saving passwords for any application set. Any existing cached passwords are deleted.

▶ **To prevent users from saving their passwords for *particular* application sets**

1. Exit the Program Neighborhood Client if it is running. Make sure all Program Neighborhood components, including the Connection Center, are closed.

2. Open the individual's user-level Pn.ini file (default directory: %User Profile%\Application Data\ICAClient) in a text editor.

3. Locate the section name that corresponds to the application set for which you want to disable password saving; for example, [MyAppSet].

4. Add the following line to the list of parameters and values in that section:

    NoSavePwordOption=On

    If the parameter already exists, make sure its value is set to **On**.

5. Add the parameter and value to each application set section as desired.

6. Save the file and exit the text editor.

7. Repeat Steps 1 through 6 for additional users.

8. Start the Program Neighborhood Client.

Adding this parameter and setting it to **On** prevents users from saving passwords for the specified application set or sets. Any existing cached passwords are deleted.

Configuring General Settings

▶ **To configure the general settings**

1. Start the Program Neighborhood Client.

2. From the **Tools** menu, click **ICA Settings** to display the **ICA Settings** dialog box.

3. Click the **General** tab.

   From the **General** page, you can configure the following settings:

   - **Client Name**. This field allows you to change the name of the client device. The MetaFrame server uses the client name to uniquely identify resources (such as mapped printers and disk drives) associated with a given client device. The client name must be unique for each device running the Citrix ICA Client.

   - **Serial Number**. This is the serial number of the ICA Client software. This field is necessary only when you are using the Citrix ICA Client with a Citrix Terminal product such as *WINFRAME* for Terminals and MetaFrame for Terminals, which require each client to have a Citrix *PC Client Pack* serial number to connect to the server. If a serial number is required, you must enter it exactly as it appears on the Serial Number card.

   - **Keyboard Layout**. Allows you to specify the keyboard layout of your client device. The MetaFrame server uses the keyboard layout information to configure the user session for your keyboard layout. The default value of **(User Profile)** uses the keyboard layout specified in your user profile.

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

   - **Display Connect To screen when making Dial-in Connections**. Select this check box to display the **Connect To** screen when you make a dial-in connection.

   - **Display terminal window when making Dial-in Connections**. Select this check box if your dial-in configuration includes third-party products, such as security devices and X.25 PADs, that require an ASCII dialog before connecting to the MetaFrame server.

- **Allow automatic client updates**. Select this check box to allow the MetaFrame server to update your Citrix ICA Client software when newer versions become available. When the MetaFrame server detects an outdated client version, it notifies you that a newer version is available and replaces the ICA Client files.

- **Pass-Through Authentication.** If you are logged on as an administrator of the client machine, select this option to enable pass-through authentication at the machine level. Each user must turn this feature on by selecting **Use local credentials to log on** described below. See "Enabling Pass-Through Authentication" on page 102 for more information.

- **Use local credentials to log on**. Select this check box to enable pass-through authentication at the user level. Pass-through authentication must also be enabled at the machine level. See "Enabling Pass-Through Authentication" on page 102 for more information.

Configuring Bitmap Caching

▸ **To configure bitmap caching**

1. Start the Program Neighborhood Client.

2. From the **Tools** menu, click **ICA Settings** to display the **ICA Settings** dialog box.

3. Click the **Bitmap Cache** tab.

   From the **Bitmap Cache** page, you can configure the following settings:

   - **Bitmap Cache Size**. Specify the size of the bitmap cache in kilobytes.

   - **Bitmap cache directory**. The default directory where the cached data is stored is displayed in this field.

   - **Change Directory**. If you want to specify a new directory for cached data, click the **Change Directory** button.

   - **Minimum size bitmap to be cached**. The size of the smallest bitmap to be cached to disk.

   - **Clear cache Now**. Click this button to remove all cached data from the directory.

Tip   Do not clear the cache if any ICA connections are open. Before clearing the cache, verify that all ICA connections are closed.

- Click **OK** to close the dialog box.

Configuring Hotkeys

▸ **To configure the hotkeys**

1. Start the Program Neighborhood Client.

2. From the **Tools** menu, click **ICA Settings** to display the **ICA Settings** dialog box.

3. Click the **Hotkeys** tab.

4. For each hotkey in the list, select a shift state and a key.

5. You can disable the hotkey by selecting **(none)** for the key.

   Hotkeys are used to control the behavior of the Program Neighborhood Client, and as substitutes for the standard Windows hotkeys for a published application.

   The fields on the **Hotkeys** page are:

   • **Task List**. The Task List hotkey displays the local **Start** menu (**Windows Task List** on Windows NT 3.51.)

   • **Close Remote Application**.The Close Remote Application hotkey disconnects the published application from the MetaFrame server and closes the Citrix ICA Client window. The behavior of this hotkey is the same as choosing **Close** from the system menu of the ICA Client window.

     Closing the published application in this manner either leaves the associated application in a disconnected state on the MetaFrame server or exits the application on the MetaFrame server, depending on how the server is configured.

   • **Toggle Title Bar**. This hotkey causes the ICA Client window to alternately display and hide its title bar. When the title bar is displayed, the ICA Client window can be moved or closed.

   Note   This hotkey must be used to return to a seamless window after accessing the Windows NT Security dialog box using the **CTRL+ALT+DEL** hotkey.

   • **CTRL-ALT-DEL**. This hotkey causes the CTRL-ALT-DEL key sequence to be sent to the server that is running the published resource. In Windows NT, the CTRL-ALT-DEL key sequence causes a Windows NT session to switch to the Windows NT Security desktop.

   • **CTRL-ESC**. This hotkey causes the CTRL-ESC key sequence to be sent to the server that is running the published application. CTRL-ESC is a standard Windows hotkey. See your Windows documentation for more information about the CTRL-ESC hotkey.

- **ALT-ESC**. This hotkey causes the ALT-ESC key sequence to be sent to the server that is running the published application. ALT-ESC is a standard Windows hotkey. See your Windows documentation for more information about the ALT-ESC hotkey.

- **ALT-TAB**. This hotkey causes the ALT-TAB key sequence to be sent to the server that is running the published application. ALT-TAB is a standard Windows hotkey. See your Windows documentation for more information about the ALT-TAB hotkey.

- **ALT-BACKTAB**. This hotkey causes the ALT-SHIFT-TAB key sequence to be sent to the server that is running the published application. ALT-SHIFT-TAB is a standard Windows hotkey. See your Windows documentation for more information about the ALT-SHIFT-TAB hotkey.

- **CTRL-SHIFT-ESC**. This hotkey causes the CTRL-SHIFT-ESC key sequence to be sent to the server that is running the published application. CTRL-SHIFT-ESC is a standard Windows NT hotkey. See your Windows NT documentation for more information about the CTRL-SHIFT-ESC hotkey.

- **Toggle Latency Reduction**. This hotkey turns SpeedScreen Latency Reduction on or off. Latency reduction reduces the time between your keyboard or mouse input and a visible response on the screen. If a published resource on the MetaFrame XP server is configured to use latency reduction, the ICA Client also uses latency reduction by default. If the latency reduction feature causes problems when running the application, you can turn it off using this hotkey.

Configuring Event Logging

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

▶ **To configure event logging**

1. Start the Program Neighborhood Client.

2. From the **Tools** menu, click **ICA Settings** to display the **ICA Settings** dialog box.

3. Click the **Event Logging** tab.

   From the **Event Logging** page, you can configure the following settings:

   Event Log File. In the **Name** field, enter the name and directory path of the file in which to log Citrix ICA Client events.

- Select **Overwrite existing event log** to cause the event log file to be overwritten with new events when a published application is run.

- Select **Append to existing event log** to keep old events and add new ones to the end of the file.

Log Events. Select the event categories that you want to log from the types listed below. If no events are selected, no logging takes place.

- **Connections and Disconnections**. Logs an event whenever the Citrix ICA Client connects and disconnects from a MetaFrame server. This category is selected by default.

- **Errors**. Logs an event whenever an error is encountered by the Citrix ICA Client. This category is selected by default.

- **Data Transmitted**. Logs an event for each packet of information sent by the Citrix ICA Client to the MetaFrame server. This is intended primarily for technical support purposes.

- **Data Received**. Logs an event for each packet of information received by the Citrix ICA Client from the MetaFrame server. This category is intended primarily for technical support purposes.

- **Keyboard and Mouse Data**. Logs an event whenever you press a key on the keyboard or move the mouse. This category is intended for technical support purposes.

Improving ICA Performance Over Low-Bandwidth Connections

If you are using ICA over 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:

- **Change your ICA Client configuration.** Changing your client configuration can reduce the bandwidth that ICA requires, and improve performance.

- **Change the way you use the client.** Changing the way you use the client can also reduce the bandwidth required for a high-performance connection.

- **Use the latest MetaFrame server and ICA Clients.** Citrix continually enhances and improves ICA performance with each release. Many performance features require the latest client and server software to function.

Changing Your ICA 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 ICA 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 MetaFrame 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.

▸ **To change the client configuration**

1. In the Windows System Tray, right-click the Program Neighborhood Agent icon and choose **Properties** from the menu that appears.

2. Select the **ICA Options** tab.

3. Make the desired configuration changes according to the guidelines below.

4. Click **OK** to save your changes.

Changing the Way You Use the Client
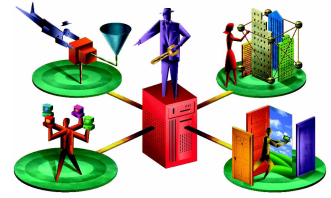
ICA technology is highly optimized and typically does not have high CPU and bandwidth requirements. However, if you are using a very low-bandwidth connection, you should consider the following to preserve performance:

- **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 ICA Win32 Clients

This chapter explains how to configure and use features common to the ICA Win32 Clients. The following topics are covered:

- Configuring New Features of Version 6.30 of the ICA Win32 Clients

- Configuring Existing Features Common to the ICA Win32 Clients

- Configuring Multiple Monitors

- Using Applications Published on MetaFrame for UNIX

Configuring New Features of Version 6.30 of the ICA Win32 Clients

Configure the following new features in the same manner for all of the ICA Win32 Clients.

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 on the MetaFrame server, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

Smart Card Support

MetaFrame XP's smart card support is based on Microsoft's PC/SC standard specifications. MetaFrame XP 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 logon pass-through authentication**.

   With **Smart card logon** selected, the ICA 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 logon pass-through authentication** selected, the ICA Client prompts the user for a smart card PIN 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 re-enter a PIN to access published resources.

4. Click **OK** to close the **Properties** dialog box.

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 XP does not handle smart card PIN management. PIN management is handled by your cards' Cryptographic Service Provider.

Configuring Existing Features Common to the ICA Win32 Clients

This section explains how to configure existing features that are common to the ICA Win32 Clients. For configuration instructions specific to each ICA Win32 Client, see the appropriate chapter about the client you plan to use.

The following topics are discussed in this section:

- 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

Auto Client Reconnect

ICA sessions can be dropped because of unreliable networks, highly variable network latency, or range limitations of wireless devices.

The *auto client reconnect* feature is triggered when the ICA Client detects a disconnected session. When this feature is enabled on a MetaFrame server, users do not have to reconnect manually or re-enter logon credentials to continue working. Automatic reconnection does not occur if users exit applications without logging off.

When a reconnection sequence begins, the user is informed that the client will reconnect after a set interval. Reconnection requires no action by users, although they can choose to cancel the process or reconnect immediately. Because session drops may be caused by network instability, wait a few moments before reconnecting to give the network time to recover from the problem that caused the disconnection.

Auto client reconnect incorporates a re-authentication mechanism based on encrypted user credentials. When a user initially logs on to a server to use an application, MetaFrame XP encrypts and stores the user credentials in memory and sends the encryption key to the ICA Client. For reconnection, the client submits the key to the server. The server decrypts the credentials and submits them to Windows logon for authentication.

Changing ICA Client Default Auto Reconnect Settings

Auto client reconnect is enabled on the ICA Win32 Clients by default. When the ICA Client detects that its connection to the server is broken, it waits 30 seconds before beginning the reconnection sequence. By default, the ICA Client attempts to reconnect three times and then stops.

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

TransportReconnectEnabled=0 to disable auto client reconnect.

TransportReconnectDelay=n to configure the number of seconds to wait before attempting to reconnect.

TransportReconnectRetries=n to configure the number of reconnection attempts.

Novell Directory Services Support

When launching ICA Win32 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 MetaFrame XP farm, NDS users enter their credentials on an NDS tab on the ICA Client logon screen. If users have the Novell Client (Version 4.8) installed, they can browse the NDS tree to choose their context. See "Enabling NDS Logon Support" on page 53 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 MetaFrame XP 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.

- **NFuse Classic.** NDS users enter their credentials on an NFuse Classic NDS logon screen. See the *NFuse Classic Administrator's Guide* for information about configuring NFuse Classic for NDS.

Note   To use NDS logon information with earlier versions of ICA Win32 Clients, enter the NDS tree name in the **Domain** field and a distinguished name in the **User** field on the ICA Win32 Client logon screen.

Setting a Default Context for NDS

You can set a default context for NDS for the Program Neighborhood Agent and for the Program Neighborhood Client. To set a default context for NDS, you must configure the Windows Installer package or the self-extracting executable, depending on which one you are using to deploy the ICA Clients.

Program Neighborhood Agent

Windows Installer Package.   For instructions about setting a default context for NDS in the Windows Installer package of the Program Neighborhood Agent, see "Configuring the Windows Installer Package for Silent User Installation" on page 41.

Self-Extracting Executable.   For instructions about setting a default context for NDS in the self-extracting executable of the Program Neighborhood Agent, see "Configuring the Self-Extracting Executable for Silent User Installation" on page 43.

Program Neighborhood Client

Windows Installer Package.   For instructions about setting a default context for NDS in the Windows Installer package of the Program Neighborhood Client, see "Configuring the Windows Installer Package for Silent User Installation" on page 83.

Self-Extracting Executable.   For general information about configuring the self-extracting executable for the Program Neighborhood Client, see "Configuring the Self-Extracting Executable for Silent User Installation" on page 84.

▸ **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 Executable for Silent User Installation" on page 84.

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

3. Add the following parameter to the file:

    DEFAULT_NDSCONTEXT=<Context1 [,…]>.

    If you are including more than one context, separate the contexts by a comma.

4. Save and close the file.

Using NT Credentials with Novell Client and Pass-Through Authentication

If the Program Neighborhood Client is configured to use pass-through authentication on a client device that has the Novell Client installed, the Program Neighborhood Client, by default, uses the NDS credentials to authenticate the user to the MetaFrame server. If you want the ICA Client to use the user's Windows NT credentials with pass-through authentication instead, you must add a parameter to the ICA Client's Appsrv.ini file. 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 the section "Configuring the Windows Installer Package for Silent User Installation" on page 83.

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 ICA Win32 Clients that use the XML Service to connect to the MetaFrame farm to request a Domain Name System (DNS) name instead of a server's 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.

The ICA Win32 Program Neighborhood Client is configured to use TCP/IP+HTTP (the XML Service) browsing by default. ICA Clients connecting to published applications through NFuse Classic also use the XML Service. For ICA Clients connecting through NFuse Classic, the NFuse Classic server resolves the DNS name on behalf of the client.

DNS name resolution is disabled by default in the MetaFrame farm and enabled by default on the ICA Win32 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 MetaFrame 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 Win32 ICA Clients**

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 MetaFrame server. When a user double-clicks a locally saved file, the file is opened by the application associated with it on the MetaFrame XP server.

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

Note   Version 6.30 of the Program Neighborhood Agent supports client-to-server content redirection, a feature introduced in Feature Release 2 of MetaFrame XP 1.0 for Windows. Functionally equivalent to extended parameter passing, client-to-server content redirection allows you to enforce all underlying file type association from the MetaFrame server, eliminating the need to configure extended parameter passing on individual client devices.

If your entire ICA Client population consists only of Program Neighborhood Agents, and if you want to take advantage of the administrative ease of client-to-server content redirection, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack 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 the MetaFrame XP server to support parameter passing, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

On the client side, you must replace the file type's **open** command with a command line that passes the file name and path to the MetaFrame 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 file type's **open** command with a command line that passes the file name and path to the application published on the MetaFrame server.

Important   MetaFrame XP 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 ICA Win32 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 ICA Win32 Client executable to include in the command line.

Determining the ICA Win32 Client Executable

Users can connect to published applications using the following methods:

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

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 NFuse Classic)	wfica32.exe
Applications in application sets (using the Program Neighborhood Client)	pn.exe

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

Identifying Published Applications

Each ICA Win32 Client executable uses different command line syntax to specify configuration data when launching published applications. When creating your command line, you must use the executable's command line syntax to correctly identify the published application.

Note   To view an executable's required command line syntax, from a command prompt, change directories to the ICA Win32 Client's installation directory and then type the executable's name 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 ICA Win32 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 set 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 that contain remapped server drives. If your MetaFrame server drives are not remapped, you must add the following text to the argument: **client**\; for example: **/param:"\\client\%1"** .

Mapping Client Devices

The Citrix ICA 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 ICA Client informs the MetaFrame 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 Windows ICA Client printers so they appear to be directly connected to the MetaFrame 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 server documentation for information about the change client command.

Turning off Client Device Mappings

On the MetaFrame server, specify client device mapping options in the **Client Settings** dialog box in Citrix Connection Configuration.
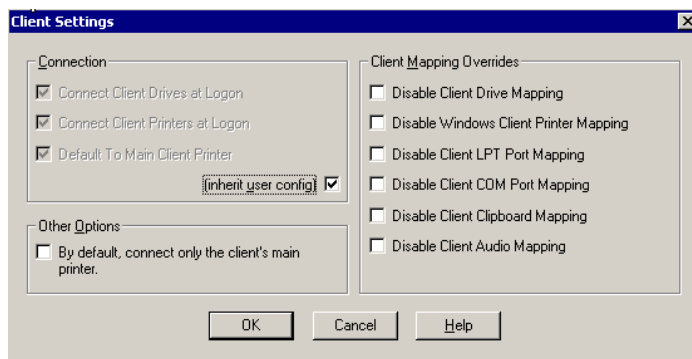
The **Connection** options control whether drives and printers are mapped to client drives and printers. If these options are cleared, the devices are still available but must be mapped to drive letters and port names manually.

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's drives are automatically mapped at logon.
Connect Client Printers at Logon	If this option is selected, the client device's printers are automatically mapped 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 MetaFrame 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 Citrix ICA 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 for UNIX 1.0 servers.

The MetaFrame 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 MetaFrame server as:
A	A
B	B
C	V
D	U

The MetaFrame server can be configured so that the server drive letters do not conflict with the client drive letters; in this case the MetaFrame server drive letters are changed to higher drive letters. For example, changing MetaFrame 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 MetaFrame server as:
A	A
B	B
C	C
D	D

The drive letter used to replace the MetaFrame 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 MetaFrame server drive letter, the network drive mapping is not valid.

When an ICA Client device connects to a MetaFrame 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 Citrix ICA Win32 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 how to configure ICA Client printing for MetaFrame for UNIX connections, see the *MetaFrame for UNIX Operating Systems 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 users do not want the automatically created printers deleted when they log off, use Print Manager in the ICA session to view the **Properties** dialog box for the client printer. This dialog box contains a **Comment** field with the string **Auto Created Client Printer** for automatically created client printers. 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 automatically be set. 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 Print Manager to connect to a client printer. These printers are not automatically deleted when you log off.

▶ **To view mapped client printers when connected to a MetaFrame server**

While connected to the MetaFrame server, double-click **My Computer** on the remote desktop and then double-click **Printers**. The **Printers** window opens:
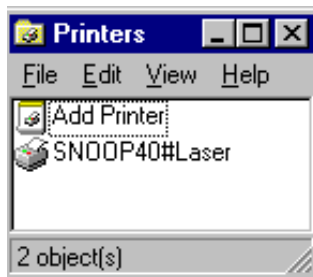
The **Printers** screen displays the local printers mapped to the ICA session. The name of the printer takes the form *clientname#printername,* where *clientname* is the unique name given to the client device during ICA Client Setup and *printername* is the Windows printer name. In this example ICA session, a client machine called "Snoop40" 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 client device's COM ports to be used during ICA sessions on a MetaFrame server. These mappings can be used like any other network mappings.

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

▶ **To map a client COM port**

1. Start the ICA Client and log on to the MetaFrame 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 MetaFrame 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 Sound Support

Client sound support mapping enables applications running on the MetaFrame server to play sounds through a Windows-compatible sound device installed on the client device. You can control the amount of bandwidth used by client audio mapping.

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

▶ **To configure ICA Client sound support on a MetaFrame server running on Windows NT 4.0, Terminal Server Edition**

1. Go to **Programs** > **Administrative Tools (Common)** > **Terminal Server Connection Configuration**.

2. Double-click a connection.

3. Click the **ICA Settings** button.

4. Select an option from the **Client Audio Quality** drop-down list.

▶ **To configure ICA Client sound support on a MetaFrame server running on Windows 2000 Server**

1. Go to **Programs** > **Administrative Tools** > **Terminal Services Configuration**.

2. Click **Connections** in the left pane.

3. Double-click a connection in the right pane.

4. Click the **ICA Settings** tab.

5. Select an option from the **Client Audio Quality** drop-down list.

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

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

The **Client Audio Quality** options are:

- **High**. This setting is recommended only for connections where bandwidth is plentiful and sound quality is important. This setting allows clients to play a sound file at its native data rate. Sounds at the highest quality level require about 1.3Mbps of bandwidth to play clearly. Transmitting this amount of data can result in increased CPU utilization and network congestion.

- **Medium**. This setting is recommended for most LAN-based connections. This setting causes any sounds sent to the client to be compressed to a maximum of 64Kbps. This compression results in a moderate decrease in the quality of the sound played on the client device. The host CPU utilization will decrease compared with the uncompressed version due to the reduction in the amount of data being sent across the wire.

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

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 Citrix ICA Client on the appropriate Windows platform

  -Or-

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

The following hardware configurations have been tested with multiple monitor support.

Important   It is highly recommended 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

Using the ICA Win32 Program Neighborhood and Web Clients with Application Launching and Embedding

If you are not planning to use NFuse Classic but still want to deliver applications over the Web, you can use Application Launching and Embedding (ALE) in conjunction with the Web-based ICA Client Installation feature. The ICA Win32 Program Neighborhood and Web Clients can be used with launched and embedded applications.

The ICA Win32 Web Client replaces the ActiveX control and Netscape Plug-In Clients and is available as a self-extracting executable. At approximately 1.8MB in size, this package is significantly smaller than the other ICA Win32 Clients. The smaller size allows quicker downloads and installation. You can configure the ICA Win32 Web Client for silent user installation.

Application Launching and Embedding

Using Application Launching and Embedding, you can provide access to full-function Windows-based applications from HTML pages over intranets or the Internet, without rewriting application code.The applications look and feel as if they are running locally, even though they are executing on the MetaFrame server.

There are two ways to access an application from a Web page: launching and embedding.

- *Launching* an application from a Web page means you click a hyperlink that references an ICA file. Clicking the hyperlink causes the application to start and appear in a separate window on the local desktop. You can then use this application as if it is installed and running on your local computer.

- *Embedding* an application places the window in which the program runs within the Web browser window.

Launched Applications

The Citrix Management Console (MetaFrame XP) and the Published Application Manager (MetaFrame 1.8) include wizards that allow you to create ICA files and HTML pages. The HTML pages are saved on your Web server for users to visit and launch ICA sessions. HTML pages that launch ICA sessions contain a hyperlink to an ICA file that is located in a public HTML directory. When clicked, the hyperlink downloads the ICA file to the client device. The ICA Client software uses the parameters in the ICA file to launch the application on the user's desktop. If the ICA file does not detect ICA Client software installed on the client device, the client software is presented to the user to download and install.

▸ **To set up a Web page so users can launch an application**

1. Copy Ica32t.exe (for the ICA Win32 Web Client) or Ica32.exe (for the ICA Win32 Program Neighborhood Client) to your Web server. These files are located in the following directories (substitute *language* with the language of the ICA Client software) of the Component CD included in your MetaFrame XP, Feature Release 2 media pack:

   Icaweb*language*\ica32

   where *language* is one of:

   • En (English)

   • Fr (French)

   • De (German)

   • Ja (Japanese)

   • Es (Spanish)

2. Publish an application. See the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack for information about publishing applications.

3. Use the Create HTML File wizard (MetaFrame XP) or the Write HTML File wizard (MetaFrame 1.8) to create an HTML page on your server. You can also create an ICA file. For more information about creating HTML files that contain published applications, see your MetaFrame server documentation and the online Help for the Citrix Management Console (MetaFrame XP) or the Published Application Manager (MetaFrame 1.8).

4. Open the HTML file in a text editor and edit the client type parameter to include the full path to the Ica32.exe or Ica32t.exe file, respectively. This parameter calls the ICA Win32 Client to run the published application.

Embedded Applications

The Citrix ICA Win32 Program Neighborhood and Web Clients allow you to embed applications into Internet Explorer and Netscape Navigator.

Rather than creating separate Web pages for Microsoft Internet Explorer and Netscape Navigator users, you can create a single Web page that contains two types of HTML tags to embed applications for Internet Explorer and Netscape Navigator.

Complete the following steps to use one of the ICA Win32 Clients to embed applications into Internet Explorer or Netscape Navigator:

1. Create an ICA Client download Web site using the Web-based ICA Client Installation feature. The elements required for the download Web site are included in your MetaFrame XP, Feature release 2 media pack. You can also download them from the Download area of the Citrix Web site at http://www.citrix.com/download. Click the "Download Web-based ICA Clients Install Components" link.

   For more information about constructing an ICA Client download Web site using your MetaFrame XP, Feature Release 2 media pack, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack.

   For more information about constructing an ICA Client download Web site using the packages downloaded from the Citrix Web site, see the corresponding Readme.htm file located on the same Web page as the packages.

2. Create an HTML page using the appropriate wizard on your MetaFrame server.

3. Set the **cabLoc** parameter for Internet Explorer users.

4. Set the **Pluginspage** parameter for Netscape Navigator users.

Detailed instructions for Steps 2, 3, and 4 are below.

▶ **To create an HTML page using MetaFrame server software**

1. On MetaFrame XP, open the Create HTML File wizard from the Citrix Management Console.

   On MetaFrame 1.8, open the Write HTML File wizard from the Published Application Manager.

2. Create the HTML file and save it to your Web server. For detailed instructions, follow the online Help for the appropriate wizard.

3. Open the HTML file in a text editor.

▶ **To set the cabLoc parameter for Internet Explorer users**

1. In the HTML file you created in the preceding Step 2, locate the line that begins with: **var cabLoc =**.

   Replace the value after the equal sign (=) with the URL of the appropriate Win32 Client: "http://*Webserver*/*directory*/ica32.exe or ica32t.exe";

   Make sure you include the quotation marks and the semicolon.

2. Save the file and make sure it is stored on your Web server along with the ICA file for the embedded session. When users visit the HTML page, Internet Explorer automatically downloads and installs the ActiveX control.

3. Keep the HTML file open.

▶ **To set the Pluginspage parameter for Netscape Navigator users**

1. Locate the line that begins with: **var plugRefLoc =**.

   Replace the value after the equal sign (=) with the URL of the default Web page of the ICA Client download Web site you created with the Web-based ICA Client Installation feature.

   ```
   Example:
   var plugRefloc = "http://mywebserver/ica60/setup.htm";
   ```
   Make sure you include the quotation marks and the semicolon.

2. Save the file and make sure it is stored on your Web server along with the ICA file for the embedded session.

3. Publish a link to the HTML page. When users visit the HTML page, Netscape Navigator refers users who do not have the plug-in to the appropriate ICA Client download page.

Using Applications Published on MetaFrame for UNIX

For connections to applications published on a MetaFrame for UNIX server, two additional utilities provide functionality for configuring session display and cutting and pasting objects between the ICA session and the client device. This section describes how to use these utilities.

Using the Window Manager

If you are connecting to an application published on a MetaFrame for UNIX server, use the Citrix window manager to minimize, resize, position, and close windows, and access seamless "full screen" mode. This section describes how to use the window manager.

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.

Accessing Seamless "Full Screen" Mode

▸ **To switch between seamless and seamless "full screen" modes**

Press SHIFT+F2.

Minimizing, Resizing, Positioning, and Closing Windows

When you connect to a published application on a MetaFrame server, buttons to minimize, resize, position, and close windows are provided by the ctxwm window manager.

▶ **To minimize, resize, position, and close windows**

Use the left mouse button to click the following buttons:

To	Click	Note
Minimize published application windows on your desktop		Seamless windows are minimized as buttons on the desktop's taskbar. Non-seamless and seamless "full screen" windows are minimized as icons on the desktop.
Open a minimized window	oclo...	Click its button on the taskbar or its icon on the desktop.
Adjust the size of published application windows		Click and hold down the mouse button, then move the pointer to the edge of the window and drag it in the direction you want to scale it. The window dimensions are displayed in the top left-hand corner. Release the mouse button to apply the resizing. To resize the window proportionately, move the mouse pointer to a corner of the window and drag it.
Reposition published application windows	oclock	Click and hold down the mouse button, drag the window to the required position on the desktop, and release the mouse button.
Close and exit a published application	X	When you close the last application in a session, the session disconnects automatically after 20 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 is displayed.

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 MetaFrame 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 MetaFrame for UNIX server, 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 MetaFrame for UNIX server.

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 **screengrab** option.

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

2. When ctxgrab is started, a dialog box is displayed.

▶ **To copy from an application in an ICA 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 ICA 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 server 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 an ICA 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 application in the ICA Window to paste the information.

▶ **To copy from an application in an ICA Client window to a local application**

1. From the application in the ICA 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 an ICA 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 ICA Client window to paste the information.

▶ **To copy from an application in an ICA Client window to xv**

1. From the application in the ICA 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.

Implementing Security Measures for the ICA Win32 Clients

This chapter discusses measures you can take to secure the communication between your MetaFrame server farm and the ICA Win32 Clients. The following topics are covered:

*   Connecting to a Server Through a Proxy Server

*   Integrating the ICA Win32 Clients with Citrix Secure Gateway or SSL Relay

*   Connecting to a Server Through a Firewall

Integrating the ICA Win32 Clients with Your Security Solutions

You can integrate the ICA Win32 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 ICA Win32 Clients with Citrix Secure Gateway or SSL Relay solutions with Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols

*   Connecting to a server through a firewall

Connecting to a Server Through a Proxy Server

Proxy servers are used to limit access into and out of your network, and to handle connections between ICA Clients and MetaFrame servers. The ICA Win32 Clients support SOCKS and Secure proxy protocols.

ICA Win32 Program Neighborhood Agent and Web Client

In communicating with the MetaFrame server, the Win32 Program Neighborhood Agent and the Win32 Web Client use proxy server settings that are configured remotely on the NFuse Classic server. See the *NFuse Classic Administrator's Guide* for information about configuring proxy server settings for these ICA Clients.

In communicating with the NFuse Classic server, the Win32 Program Neighborhood Agent and the Win32 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.

ICA Win32 Program Neighborhood Client

The Win32 Program Neighborhood Client uses proxy server settings you configure locally from the ICA 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 ICA Win32 Client in an organization with many 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 4.0 or later, or Netscape for Windows 4.76 or later.

▸ **To enable auto-client proxy detection**

1. Start the Program Neighborhood Client.

2. If you are configuring an application set:

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

     If you are configuring a custom ICA connection:

     Right-click the custom ICA connection you want to configure and select **Custom Connection 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 the Program Neighborhood Client.

2. If you are configuring an application set:

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

     If you are configuring a custom ICA connection:

     Right-click the custom ICA connection you want to configure and select **Custom Connection 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**.

6. Specify the port number of the proxy server (if other than 1080 for SOCKS or 8080 for Secure proxy).

7. 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 ICA Client to pass the credentials without user intervention. To that end, you create a default setting that applies:

•   To one or several existing custom ICA connections

    -or-

•   To all future custom ICA connections to be created using the Add ICA Connection wizard

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

▸   **To create a default for one or several existing custom ICA connections**

1. Exit the Program Neighborhood Client 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 for which you want to create the default.

4. Add the following lines to the list of parameters and values in the **[*ServerLocation*]** section:

    ProxyUsername=<**user name**>

    where user name is the user name recognized by the proxy server.

    ProxyPassword=<**password**>

    where 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.

▸   **To create a default for all future custom ICA connections to be created using the Add ICA Connection wizard**

1. Exit the Program Neighborhood Client 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 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.

Integrating the ICA Win32 Clients with Citrix Secure Gateway or SSL Relay

You can integrate the ICA Win32 Clients with a Citrix 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.

Citrix Secure Gateway

You can use Citrix Secure Gateway in either *Normal* mode or *Relay* mode. No ICA Client configuration is required if you are using Citrix Secure Gateway in Normal mode.

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

*   The fully qualified domain name (FQDN) of the Secure Gateway server

*   The port number of the Secure Gateway server

ICA Win32 Program Neighborhood Agent and Web Client

To connect to a Citrix Secure Gateway server, the Win32 Program Neighborhood Agent and the Win32 Web Client use settings that are configured remotely on the NFuse Classic server. See the *NFuse Classic Administrator's Guide* for information about configuring proxy server settings for these ICA Clients.

ICA Win32 Program Neighborhood Client

▸ **To configure the details of your Secure Gateway server**

1. Make sure the client device meets all system requirements outlined on page 81 and page 147 of this guide.

2. Start the Program Neighborhood Client.

3. If you are configuring an application set:

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

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

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

Citrix SSL Relay uses TCP port 443 by default to listen for SSL/TLS-secured connections. When the SSL Relay receives an SSL/TLS connection, it decrypts the data before redirecting it to the MetaFrame 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 ICA Client and a MetaFrame server. Connections using SSL/TLS encryption are marked with a padlock icon in Citrix Connection Center.

• In an NFuse Classic deployment, between the MetaFrame server and the NFuse Classic server.

For information about configuring and using SSL Relay to secure your MetaFrame installation, see the *MetaFrame XP Administrator's Guide* included in your MetaFrame XP, Feature Release 2 media pack. For information about configuring the NFuse Classic server to use SSL/TLS encryption, see the *NFuse Classic Administrator's Guide*.

System Requirements

In addition to the system requirements listed for each ICA Win32 Client in its respective chapter, you also must ensure that the client device:

• Supports 128-bit encryption

• Has a root certificate installed that can verify the signature of the Certificate Authority on the server certificate

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.

About Root Certificates

See the section "Installing Root Certificates on the ICA Win32 Clients" on page 151 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.

Configuring and Enabling the 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.

Note   The ICA Win32 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 ICA Clients support, or connection may fail.

Forcing TLS Connections for all ICA Win32 Clients

To force the ICA Win32 Clients (including the ICA Win32 Web Client) to connect with TLS, you must specify TLS on your Citrix Secure Gateway server or SSL Relay service. See the *Citrix Secure Gateway Administrator's Guide* or SSL Relay service documentation for more information.

Important   By default, Citrix SSL Relay uses TCP port 443 on the MetaFrame server for SSL/TLS-secured communication. If you configure SSL Relay to listen on a port other than 443, you must change the **SSLProxyHost** setting to reflect the new port number in the following situations:

- If the user is connecting to a published application accessed through an application set, change the **SSLProxyHost** setting in the user's local Appsrv.ini and Pn.ini files.

- If the user is connecting to a published application or MetaFrame server through a custom ICA connection, change the **SSLProxyHost** setting in the user's Appsrv.ini file.

▸   **To configure the ICA Win32 Program Neighborhood Client to use SSL/TLS**

1. Make sure the client device meets all system requirements outlined on page 81 and page 147 of this guide.

2. Open the Program Neighborhood Client.

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**. The **Application Set** dialog box 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 **Connection Properties** dialog box 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 Connection 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 MetaFrame server(s) to the Address List.

6. Click **OK**.

▶ **To configure the ICA Win32 Program Neighborhood Agent to use SSL/TLS**

1. Make sure the client device meets all system requirements outlined on page 39 and page 147 of this guide.

2. To use SSL/TLS to encrypt application enumeration and launch data passed between the Program Neighborhood Agent and the NFuse Classic server, configure the appropriate settings in the Config.xml file on the NFuse Classic server. The Config.xml file must also include the machine name of the MetaFrame server hosting the SSL certificate.

3. To use secure HTTP (HTTPS) to encrypt the configuration information passed between the Program Neighborhood Agent and the NFuse Classic server, enter the URL of the NFuse Classic server hosting the Config.xml file 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 Program Neighborhood Agent 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 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 Program Neighborhood Agent 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 NFuse Classic server, or in the user-level Appsrv.ini file of the local client device. See the *NFuse Classic 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 Program Neighborhood Agent 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 section named **[WFClient]**.

4. Set the values of these two parameters as follows:
   SSLENABLE=On
   SSLCIPHERS=GOV
   SECURECHANNELPROTOCOL=TLS

5. Save your changes.

Installing Root Certificates on the ICA Win32 Clients

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

The Citrix ICA Win32 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 Citrix ICA Win32 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 Profiles. See your Microsoft Windows 2000 documentation for more information.

Otherwise, 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 the Win32 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**.

4. The Certificate Import Wizard starts. 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.

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 ICA Win32 Clients through a network firewall that maps the server's internal network IP address to an external Internet address, do the following:

1. Open the Program Neighborhood Client.

2. If you are configuring an application set:

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

    If you are configuring a custom ICA connection:

    Right-click the custom ICA connection you want to configure and select **Custom Connection Settings**. The **Custom ICA Connections** dialog box appears.

3. Click **Add**. The **Add Server Location Address** window appears.

4. Enter the external Internet address of the MetaFrame server.

5. Click **OK**. The newly added external Internet address of the MetaFrame server appears in the Address List.

6. Click **Firewalls**.

7. Select **Use alternate address for firewall connection**.

8. Click **OK** twice.

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

Updating the ICA Win32 Clients
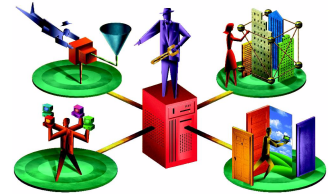
This chapter explains how to deploy ICA Client updates across your network. The following topics are covered:

- The ICA Client Update Process

- Using the Client Update Configuration Utility

- Specifying a Default Client Update Database

- Configuring Default Client Update Options

- Adding ICA Clients to the Client Update Database

- Working with the ICA Win32 Client Downloaded from the Citrix Web Site

- Removing an ICA Client From the Client Update Database

- Changing the Properties of the ICA Win32 Client

About the Client Auto Update Feature

The ICA Win32 Program Neighborhood Agent and the ICA Win32 Program Neighborhood Client are available as Microsoft Windows Installer (.Msi) packages. 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 Windows Installer packages.

To deploy updates to the ICA Win32 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 ICA 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 MetaFrame 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 ICA Win32 Clients on your network. You can also use this feature to revert to a previous version of a client. Visit the Citrix Web site at http://www.citrix.com/download frequently for the latest releases and documentation of the ICA Win32 Clients.

As new versions of ICA Clients become available, you add them to the Client Update Database. When an ICA Client logs on to a MetaFrame 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 ICA Win32 Client installed with Windows Installer (.Msi) packages. You must redeploy an ICA Win32 Client installer package when a new version of the ICA Client is released.

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

- Automatically detects ICA Client versions

- Copies new files over any ICA connection without user intervention

- Provides administrative control of update options for each ICA Client

- Updates ICA Clients from a single database on a network share point

- Safely restores older ICA Client versions when needed

The ICA Client Update Process

ICA Clients are identified by platform with a product and model number. The version number is assigned when new ICA Clients are released.

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

- By default, the MetaFrame 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 and downloading the files in the background while they continue to work. Users connecting to the MetaFrame server with a modem get 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 ICA 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. The user must log on to the MetaFrame server again to continue working.

- When the user disconnects from the server and closes all client programs, the ICA Client update process finishes.

- As a safeguard, the existing ICA 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 disk.

Configuring the Client Update Database

You can configure a Client Update Database on each MetaFrame server in a server farm, or configure one database to update the ICA Clients for multiple MetaFrame servers.

The Client Update Database contains several ICA Clients. As Citrix releases new versions of the ICA Clients, you add them to the Client Update Database.

Using the Client Update Configuration Utility

Use the 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 ICA Clients to the database
- Remove outdated or unnecessary ICA Clients
- Change the properties of an ICA Client in the database
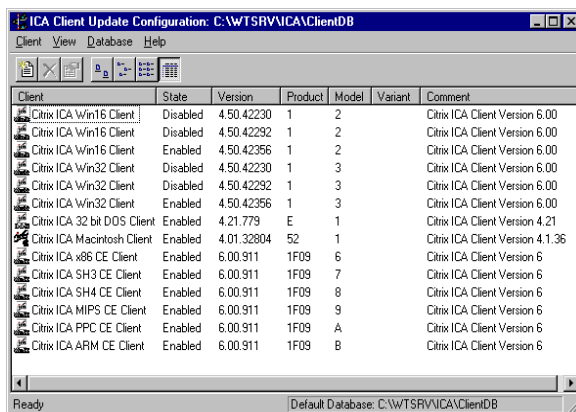
The following sections give an overview of the Client Update Configuration utility. For details, see the utility's online help.

▶ **To start the ICA Client Update Configuration utility**

1. 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.

2. The **ICA Client Update Configuration** window appears. The status bar shows
   the location of the current update database, which the MetaFrame server uses to
   update ICA Clients. The window shows the ICA Clients in the database.

Note MetaFrame for UNIX does not use the Client Update Database. To use
the Client Update Database, you must have a MetaFrame or MetaFrame XP for
Windows server in your server farm.

Creating a New Client Update Database

The ICA Client Distribution wizard creates the Client Update Database in the
location %SystemRoot%\Ica\ClientDB. You can create a new update database in
any location on a server disk or on a network share point.

▶ **To create a new update database**

1. From the **Database** menu, choose **New**. The **Path for the new Client Update
   Database** dialog box appears.

2. Enter 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 MetaFrame servers. If the Client Update Database is on a shared network drive, use the ICA Client Update Configuration utility to configure your MetaFrame servers to use the same shared database.

▶   **To set the default database for MetaFrame servers**

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

2. Specify the path to the default database and click **Open**. The database opens.

3. On the **Database** menu, click **Set Default**. The **Set Default Database** dialog box opens:
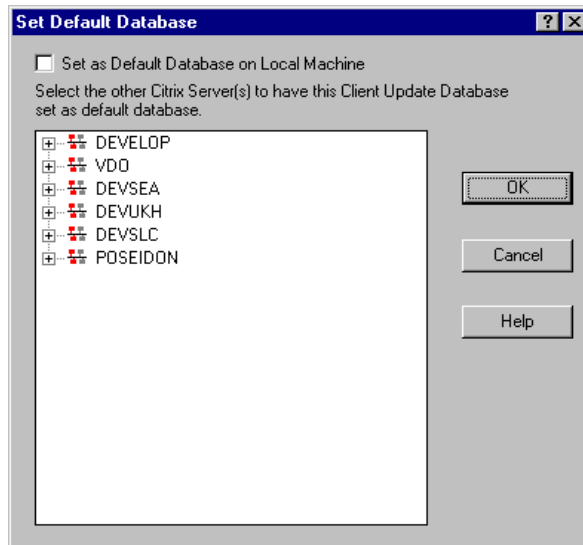
4. Select **Set as Default Database on Local Machine** to make the currently opened database the default database. You can also set other MetaFrame servers to use the currently open database as the default database.

5. Double-click a domain name to view the servers in that domain. Click a server to set its default database to the currently open database. 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. Choose **Properties** from the **Database** menu to display the dialog box.
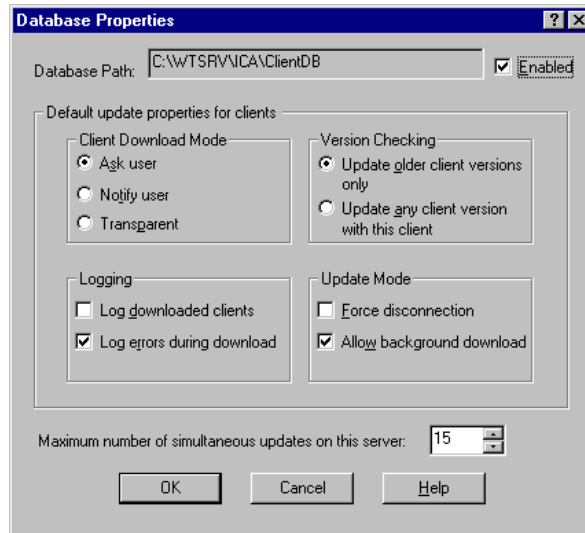
- The **Database Path** box displays the path and file name of the database you are configuring.

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

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

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

- 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.

Click **OK** when you finish configuring the database settings.

Adding ICA Clients to the Client Update Database

To deploy a newer version of the ICA Win32 Client, add it to the Client Update Database. You can download the latest ICA Clients from the Citrix Web site at http://www.citrix.com/download.

Working with the ICA Win32 Client Downloaded from the Citrix Web Site

If you downloaded a new version of the ICA Win32 Program Neighborhood Client, you must first extract the files from the Ica32.exe file before you can add the client to the Client Update Database.

▶ **To extract files from Ica32.exe**

1. Copy Ica32.exe to the root of your MetaFrame server's hard drive.

2. Create a directory to contain the extracted files.

3. At a command prompt, type the following, substituting *samplepath* with the path to the directory you created in Step 2:

   ica32.exe /a /extract /path c:*samplepath*

Note   In the above command, the /path command is optional, but recommended. If you do not specify a path, the files are extracted to %SystemRoot%\Ica\Cltimage by default.

▶ **To add a Citrix ICA Client 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 System32\Clients\Ica. You can also find the Update.ini file on the MetaFrame XP Components CD.
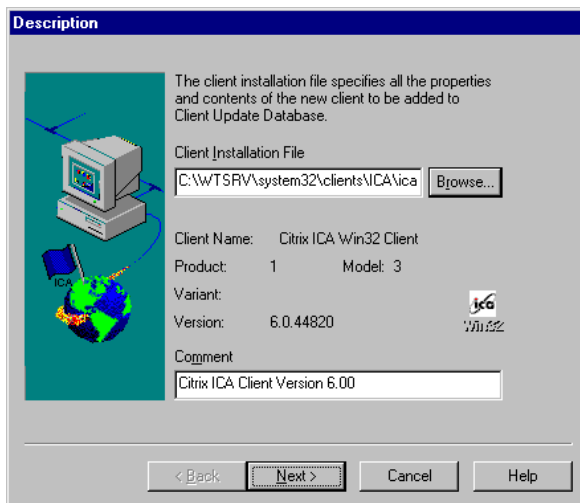
3. The client name, product number, model number, and version number are displayed. The **Comment** text box displays a description of the new client. You can modify this comment.

   Click **Next** to continue.

4. The **Update Options** dialog box appears. The options on 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.
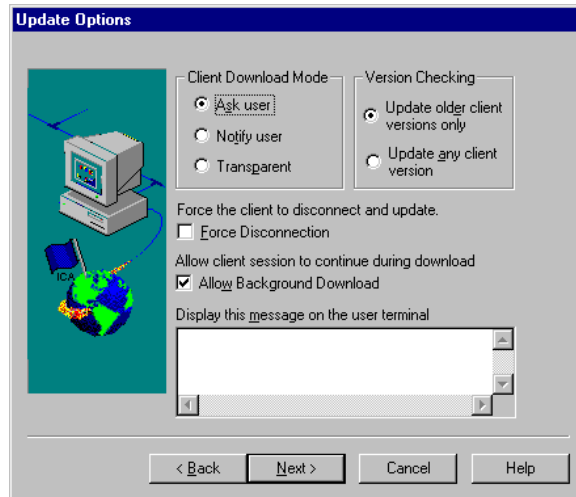
The options available in this dialog box are discussed in the online Help for this dialog box.

Click **Next** when you finish configuring the client update options.

5. The **Event Logging** dialog box appears.
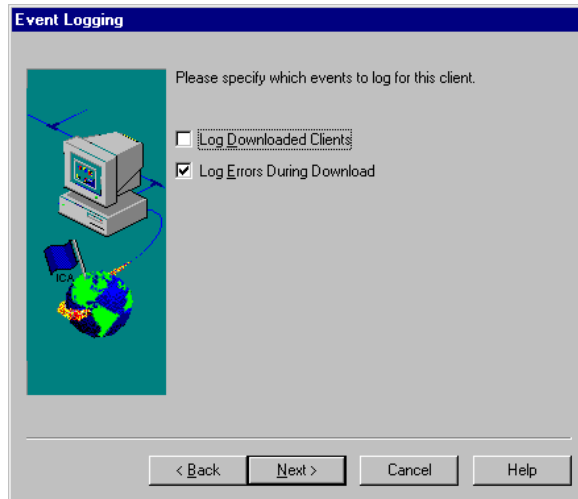
The database-wide logging options are displayed. You can specify different behavior for individual clients.

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**.

6. The **Enable Client** dialog box appears.
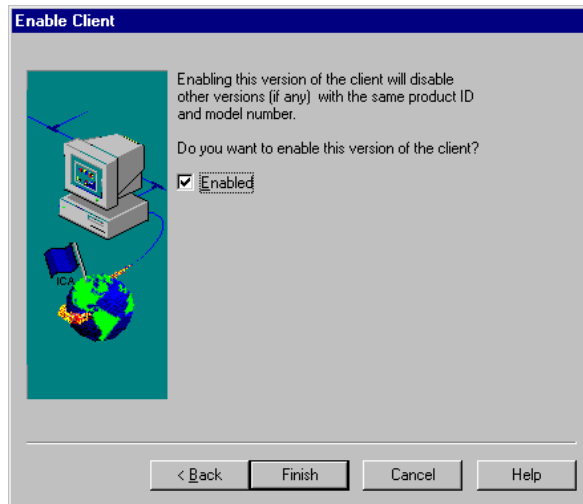
The Client Update Database can contain multiple versions of an ICA Client with the same product and model numbers. However, only one version of the client can be enabled. The enabled client is used for client updating.

Click **Finish** to copy the ICA Client installation files into the Client Update Database.

Removing an ICA Client From the Client Update Database

It is important to delete ICA 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 the ICA Win32 Client from the database**

1. Select the Win32 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. Click **Yes** to remove the client.

Changing the Properties of the ICA Win32 Client

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

▶ **To change the properties of the ICA Win32 Client**

1. Select the Win32 Client.

2. On the **Client** menu, choose **Properties**. The **Properties** dialog box appears, containing tabs labeled **Description**, **Update Options**, **Event Logging**, and **Client Files**.
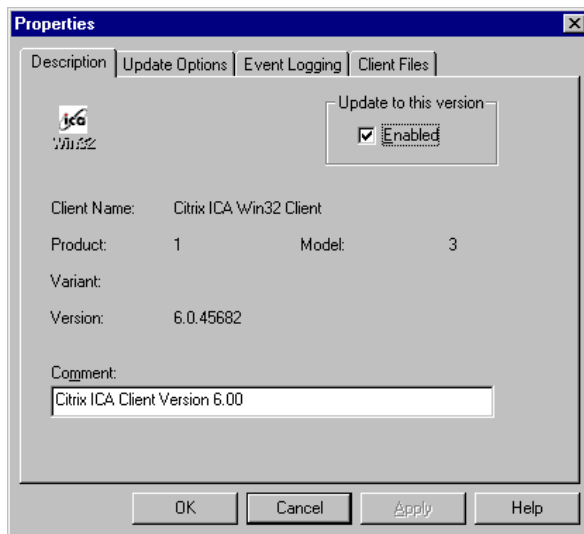
3. The **Description** tab of the **Properties** dialog box lists the client name, product number, model number, and version number.

   Select **Enabled** to update the same platform ICA Client to this version.

   Optionally, enter a new comment in the **Comment** text box.

4. Use the **Update Options** tab to configure update options for the client.
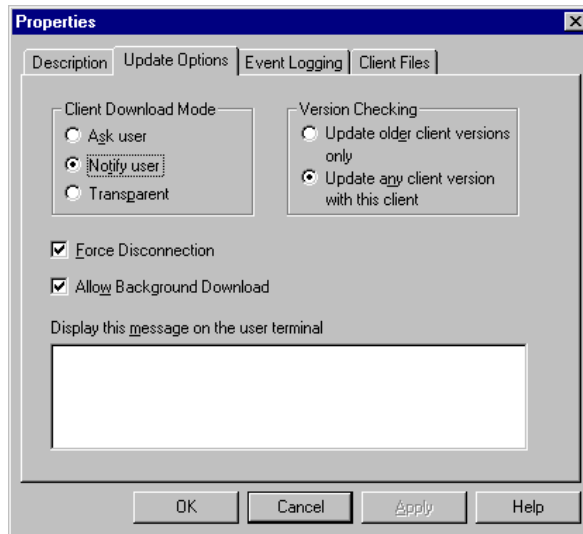
- Under **Client Download Mode**, select **Ask user** to give the user the choice to accept or postpone the update process. Select **Notify user** to notify the user of the update and require the client update. Select **Transparent** to update the user's ICA Client software without notifying or asking 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.
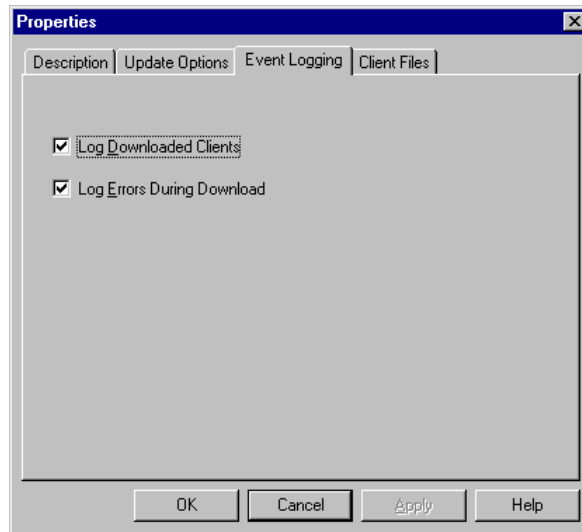
- Select the **Log Downloaded Clients** option to write an event to the event log when a client is updated.
- Select the **Log Errors During Download** option to write errors that occur during a client update to the event log.

6. Use the **Client Files** tab to view the list of files associated with this client.
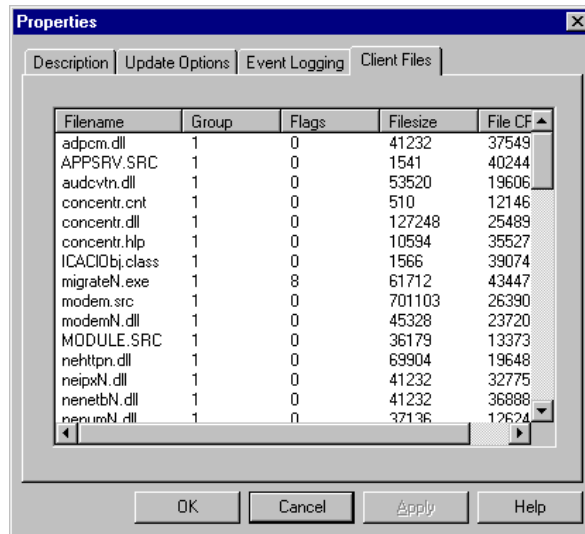
The Client Update Database stores the following information about each client
file: file name, group, flags, file size, and file CRC.

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

Config.xml Elements

Element	Default	Description
FolderDisplay/StartMenuDisplay/Enabled	true	If "true," the check box is enabled. Allows users to place shortcuts to remote resources in the Start menu.
FolderDisplay/StartMenuDisplay/ RootFolder/	none	Creates a shortcut folder in the Start menu. The folder has the name you enter here if the root attribute = "" (blank) and the modifiable attribute = "false" and the forcedefault attribute = "true"; disables the check box.
FolderDisplay/StartMenuDisplay/ RootFolder/root attribute	programs	If "true," the check box and the custom folder name text box are enabled. Users may group shortcuts in a custom folder of the Start menu; rename the custom folder; move the custom folder to the Programs submenu.
FolderDisplay/DesktopDisplay/Enabled	false	If "true," the check box is enabled. Users may group shortcuts in a custom folder of the Windows desktop.
FolderDisplay/DesktopDisplay/Icon/Name	Citrix Program Neighborhood	If "true," the custom folder name text box is enabled. Users can rename the custom desktop folder.
FolderDisplay/DesktopDisplay/Icon/ Location	(URL)	URL to the file to be used as the icon for the desktop folder. If blank, the Program Neighborhood Agent default icon is used.
FolderDisplay/SystemTrayMenuDisplay/ Enabled	true	If "true," users can place shortcuts to remote resources in the System Tray.
ConfigurationFile/Location	http://*servername*/ Citrix/PNAgent/ config.xml	The URL of the configuration file the client accesses to request configuration data.
ConfigurationFile/Refresh/Poll/Enabled	false	An option to have the configuration data updated on a poll mechanism.

Element	Default	Description
ConfigurationFile/Refresh/Poll/Period	8	The interval, in hours, to refresh the configuration information.
Request/Enumeration/Location	http://*servername*/ Citrix/PNAgent/ enum.asp	The URL for the enum.asp or enum.jsp file on the NFuse Classic server.
Request/Enumeration/Smartcard_Location	https://*servername*/ Citrix/PNAgent/ smartcard_enum.asp	The URL for the smartcard_enum.asp or smartcard_enum.jsp file on the NFuse Classic server.
Request/Enumeration/Refresh/ OnApplicationStart	true	Refresh published application data when Program Neighborhood Agent starts.
Request/Enumeration/Refresh/ OnResource Request	false	Refresh published application data when a published application is launched.
Request/Enumeration/Refresh/Poll/ Enabled	true	Refresh enumeration on a timed interval. Note: **forcedefault** and **modifiable** attributes are on Request/Enumeration/Refresh/Poll.
Request/Enumeration/Refresh/Poll/Period	6	If Request/Enumeration/Refresh/Poll/Enabled is "true," this parameter specifies the interval, in hours, at which published application data is refreshed. This parameter corresponds to the **Refresh list on hourly interval** text entry box in the **Application Refresh** tab of the **PN Agent Properties** dialog box of the Program Neighborhood Agent. Note: **forcedefault** and **modifiable** attributes are on Request/Enumeration/Refresh/Poll.
Request/Resource/Location	http://*servername*/ Citrix/PNAgent/ enum.asp http://*servername*/ Citrix/PNAgent/ launch.asp	The URL for the launch.asp or launch.jsp file on the NFuse Classic server.
Request/Resource/Smartcard_Location	https://*servername*/ Citrix/PNAgent/ smartcard_launch.asp	The URL for the smartcard_launch.asp or smartcard_launch.jsp file on the NFuse Classic server.

Element	Default	Description
Logon/LogonMethod	prompt sson smartcard_prompt smartcard_sson anonymous	Specifies what logon options users have available for ICA sessions: If "prompt," the client prompts the user for user name, password, and domain name. "Prompt" is the default if "sson" is selected but not available. If "sson," the client uses pass-through authentication. If "anonymous," the client does not prompt the user for credentials; the user is authenticated as anonymous.
Logon/EnableSavePassword	false	If "true", and if users select a logon mode that prompts them for a password, the logon screen presents them with a check box that allows them to save the password. If they choose to save the password, it is automatically entered the next time they log on.
Logon/SupportNDS	false	Specifies whether to include NDS authentication. If "true," you must also define **Logon/NDS_Settings/DefaultTree.**
Logon/NDS_Settings/DefaultTree	N/A	If NDS authentication is enabled, this parameter must specify the NDS tree to use for user authentication.
UserInterface/ServerSettings	true	Shows/hides the **Server** tab.
UserInterface/FolderDisplaySettings	true	Shows/hides the **Application Display** tab.
UserInterface/RefreshSettings	false	Shows/hides the **Application Refresh** tab.
FileCleanup/Logoff	false	If "true," removes desktop and **Start** menu shortcuts upon logoff.
FileCleanup/Exit	false	If "true," removes desktop and **Start** menu shortcuts upon exit.
ICA_Options/DisplaySize/Options/Percent	(1-100)	Specifies the size for ICA session windows as a percentage of the total client screen size.
ICA_Options/DisplaySize/Options/Dimension/Width	(NUMBER)	Specifies the width for ICA session windows, in pixels.
ICA_Options/DisplaySize/Options/Dimension/Height	(NUMBER)	Specifies the height for ICA session windows, in pixels.

Element	Default	Description
ICA_Options/DisplaySize/Options/Mode	seamless fullscreen	Allows the user to specify the size of the ICA session window. Each value must be specified separately.
ICA_Options/ColorDepth/Options	1 2 4 8	Specifies what color depth options users have available for ICA sessions: 1 - 16 colors 2 - 256 colors 4 - High color (16 bit) 8 - True color (24 bit) Each value must be specified separately.
ICA_Options/Audio/Options	high low medium off	Specifies what audio quality options users have available for ICA sessions. Each value must be specified separately.
ICA_Options/ICA_TemplateFile	template.ica	Specifies the name of the ICA template file.

Index