

Administrator's Guide

Citrix MetaFrame for UNIX[®] Operating Systems

Version 1.1, Solaris Intel / Solaris SPARC[®], HP-UX[®] and AIX[®]
Citrix Systems, Inc.

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Citrix Systems, Inc.

© 2000 Citrix Systems, Inc. All rights reserved.

Citrix, Independent Computing Architecture (ICA), SecureICA, Program Neighborhood, SpeedScreen, ReadyConnect, MetaFrame and WinFrame are registered trademarks or trademarks of Citrix Systems, Inc. in the U.S.A. and other countries.

UNIX is a registered trademark of The Open Group in the U.S.A. and other countries.

Solaris is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

HP-UX is a registered trademark of Hewlett-Packard Company.

AIX is a registered trademark of International Business Machines Corporation.

ctwm source. Copyright 1988 by Evans & Sutherland Computer Corporation, Salt Lake City, Utah

Portions Copyright 1989 by the Massachusetts Institute of Technology, Cambridge, Massachusetts

All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Evans & Sutherland and M.I.T. not be used in advertising in publicity pertaining to distribution of the software without specific, written prior permission.

EVANS & SUTHERLAND AND M.I.T. DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL EVANS & SUTHERLAND OR M.I.T. BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

ctwm source. Copyright 1992 Claude Lecommandeur. Permission to use, copy, modify and distribute this

software [ctwm] and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in

supporting documentation, and that the name of Claude Lecommandeur not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Claude Lecommandeur make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Claude Lecommandeur DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL Claude Lecommandeur BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

X Server source code. Copyright 1996 X Consortium Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

xload. Copyright (c) 1989 X Consortium. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

X Window System is a trademark of X Consortium, Inc.

XFree86. Copyright 1990,91 by Thomas Roell, Dinkelscherben, Germany. Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Thomas Roell not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Thomas Roell makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THOMAS ROELL DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THOMAS ROELL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

FreeType Engine. The software is based in part of the work of the FreeType Team. The FreeType project is copyright (C) 1996-1998 by David Turner, Robert Wilhelm, and Werner Lemberg.

XPM Library (C) 1989-95 GROUPE BULL

xv. Copyright 1994 by John Bradley. All rights reserved.

All other Trade Names referred to are the Servicemark, Trademark, or Registered Trademark of the respective manufacturers.

Part Number 300-0076

Contents

Welcome to Citrix MetaFrame for UNIX Operating Systems	1
Overview	1
What's New in this Release?	1
New Users—Read This First	2
Documentation Conventions	2
UNIX Command-line Conventions	3
A Quick Overview	4
MetaFrame Features	6
Integrating With Other Citrix Servers	9
Getting Started Quickly	9
What To Do Next	10
Getting More Information	11
Finding Resources on the Citrix Web Site	11
Providing Feedback About this Guide	12
Chapter 1 Deploying MetaFrame	13
Overview	13
Before You Begin	14
System Requirements	14
Minimum Machine Specifications	14
On the Solaris Platform	15
On the HP-UX Platform	15
On the AIX Platform	15
UNIX Operating System Requirements	16
On the Solaris Platform	16
On the HP-UX Platform	16
On the AIX Platform	17
Euro Currency Symbol Support	17
Installing MetaFrame	17
Installation Overview	17
Creating the Citrix Server Administrator User and Group	18
Installing MetaFrame on Solaris	18
Installing MetaFrame for the First Time	18
Upgrading from a Previous Release of MetaFrame	19
Performing an Unattended Install	21
Installing MetaFrame on HP-UX	22
Performing an Unattended Install on HP-UX	23
Installing MetaFrame on AIX	24

Performing an Unattended Install on AIX	25
Setting the Paths to MetaFrame Commands	26
Configuring User Access to Commands	26
Configuring ctxsrvr Access to Commands	27
Setting the Path to the man pages	27
Starting and Stopping MetaFrame	28
Starting MetaFrame	28
Stopping MetaFrame	28
About ICA Client Keyboard Support	30
Configuring Non-English Keyboard Support	30
Troubleshooting Non-English Keyboard Support	31
Configuring MetaFrame Event Logging	31
Removing MetaFrame	32
Reinstalling MetaFrame	34
Chapter 2 Citrix Licensing	35
Overview	35
What Is Citrix Licensing?	36
License Types	36
The Grace Period	36
About User Counts	37
Pooling User Counts	37
Client Device Licensing	37
Activating a License	38
Step 1—Adding the License Serial Number	38
Step 2—Getting an Activation Code	39
Step 3—Activating the License	39
Installing an Upgrade License	40
Displaying Information About Licenses	41
Adjusting the Pooled User Count	42
Removing a License	42
Removing an Upgrade License	42
Chapter 3 Publishing Applications and Desktops	43
Overview	43
About Published Applications	44
Why Publish Applications?	44
Administrative Control	44
User Access	44
Efficient Use of Resources	45
Publishing Applications for Explicit or Anonymous Use	45

Publishing Applications for Explicit Use	45
Publishing Applications for Anonymous Use.	45
Configuring Published Applications	46
Publishing an Application, Shell Script, or Desktop.	46
Publishing an Application	46
Publishing a Shell Script	47
Publishing a Desktop	48
Publishing a Java Application	48
Publishing a UNIX Command Line.	48
Publishing an Application on a UNIX Server of Different Architecture.	48
Displaying Details About Published Applications	51
Changing the Settings of a Published Application	52
Enabling and Disabling Published Applications.	53
Creating a New Published Application from Existing Details	53
Renaming a Published Application.	54
Publishing an Application on More than One Server	55
Restricting Connections to Published Applications Only.	56
Deleting a Published Application	56
Configuring an Initial Program	57
Deploying Applications that Require a 3-Button Mouse	58
Publishing an Application With Mouse Mappings.	58
Step 1—Identify the Mouse Buttons	58
Step 2—Decide How to Emulate the Missing Mouse Button	59
Step 3—Publish the Application With Mouse Mappings	59
Publishing Pre-Configured Applications for Anonymous Use	60
Chapter 4 Managing Servers, Users, and Sessions	63
Overview	63
Displaying Information About Users and Sessions	64
Understanding the Display	65
Displaying Information About Servers on the Network	66
Understanding the Display	66
Ending a Session	67
Logging Off a Session.	67
Disconnecting a Session	68
Connecting to a Disconnected Session.	68
Resetting a Session.	69
Reconnecting to Load Balanced Sessions	70
Shadowing a User's Session.	70
Ending Shadowing	72

Configuring a Different Hotkey to End Shadowing.....	72
Sending Messages to Users.....	73
Printing	74
Displaying Client Printers or Printer Ports	74
Printing From the Command Line.....	75
Printing From Applications	76
Troubleshooting Printing.....	76
Chapter 5 Configuring a MetaFrame Server	77
Overview.....	77
Configuring the Server	78
Controlling Logon Settings	78
Setting the Number of Permitted ICA Connections	80
Controlling Behavior for Disconnected or Broken Connections.....	80
Enabling or Disabling Printing for Users	81
Enabling or Disabling Clipboard Mapping	82
Providing Additional Graphics Clipboard Support	83
Providing Users with ctxcapture	83
Enabling or Disabling Shadowing.....	84
Controlling Timeout Behavior.....	86
Generating and Using Server Configuration Details.....	88
Displaying a List of the Current Configuration	88
Creating a Shell Script of the Current Configuration.....	88
Propagating Server Configuration to Multiple Servers	89
Screensaver Setting Recommendations	89
MetaFrame for UNIX and the ICA Browser Service	90
Controlling the Master Browser	90
Locating the Current Master ICA Browser	91
Manipulating Master Browser Elections.....	91
Introducing a New Server	92
Biasing the Results of Elections.....	92
Configuring the ICA Browser	93
Starting and Stopping the ICA Browser.....	94
Configuring ICA Gateways	95
About Customizable Scripts and Files	96
Customizing the Login Screen.....	96
Changing the Window Manager	97
Changing the Window Manager for Every User	97
Changing the Window Manager for a Particular User.....	98

Chapter 6 Advanced Topics	99
Overview	99
Load Balancing Published Applications	100
Load Balancing Using a Group of Servers	100
Troubleshooting Load Balancing	101
Configuring Anonymous Users	102
Displaying Anonymous User Settings	102
Configuring Anonymous User Settings	103
Changing the Number of Anonymous Users	103
Changing the Naming of Anonymous User Accounts	104
Setting an Idle Timeout Period	104
Configuring MetaFrame Security	105
Why Use MetaFrame Security?	105
Security Overview	105
Which Users Are Affected by Security?	105
Which Functions Can ctxsecurity Control?	106
Controlling Security at Different Levels	107
The Security Checking Process	107
Default Security Settings	109
Displaying Security Settings for a Function	110
Configuring Security Settings	110
Changing the Global Security Settings	110
Configuring Security for a User	111
Configuring Security for Groups of Users	111
Using Inherit to Remove Settings	111
Examples	112
Configuring MetaFrame for Non-Standard Shells	113
Using ICA With Network Firewalls	114
ICA Browsing With Network Address Translation	115
Returning External Addresses to ICA Clients	115
Configuring the TCP/IP Port Number	116
Configuring the Operating System for a Large Number of Connections	117
Configuring a Solaris System	117
Changing the Number of Pseudo-Terminals	117
Increasing File Limits	118
Increasing the Number of Concurrent CDE Sessions	118
If the Database Gets Corrupted	119
Configuring an HP-UX System	120
Configuring an AIX System	121
Changing the Number of Pseudo-terminals	121
Increasing the Number of Processes Per User	121

Appendix A MetaFrame Command Reference	123
Overview	123
ctx3bmouse	124
ctxalt	125
ctxanoncfg	126
ctxappcfg	127
ctxbrcfg	129
ctxcapture	130
ctxcfg	131
ctxconnect	133
ctxdisconnect	133
ctxgrab	134
ctxlicense	135
ctxlogout	136
ctxlpr	137
ctxmaster	138
ctxmsg	138
ctxprinters	139
ctxqserver	140
ctxqsession	142
ctxquser	142
ctxreset	143
ctxsecurity	144
ctxshadow	145
ctxshutdown	147
ctxsrv	148
Appendix B Glossary	149
Overview	149
Index	161

Welcome to Citrix MetaFrame for UNIX Operating Systems

Overview

This manual is for system administrators responsible for installing, configuring, and maintaining MetaFrame for UNIX Operating Systems.

This chapter will help you to get started using MetaFrame. It includes:

- A brief overview that describes how server-based computing works
- Descriptions of some of the MetaFrame features that you may want to use in your MetaFrame deployment
- A quick guide to installing MetaFrame, with cross-references to help you get the most from this manual and other Citrix documentation

What's New in this Release?

Release 1.1 of MetaFrame for UNIX includes the following new features:

- Support for the Solaris 8, AIX and HP-UX operating systems.
- The ability to start and stop the browser independently of other MetaFrame processes.
- Support for French, German, Swedish, Spanish, and Italian keyboards.
- The ability to disable a published application without having to delete its configuration, and then enable it at a later time.
- Graphics clipboard support that allows users to copy graphics between applications running in ICA sessions and applications running on the local client device.
- The ability to configure a hotkey to end shadowing. This is useful if the default hotkey combination cannot be used, or another key combination is preferable.

Solaris

HP-UX

Solaris

- Upgrade licensing.
- Support for Pluggable Authentication Modules (PAM) for user name and password validation. AIX uses its own authentication mechanism.
- Improved support for Citrix NFuse—see the Citrix website for more details on this feature.

New Users—Read This First

Documentation Conventions

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

Convention	Meaning
Bold	Where appropriate, this indicates boxes and buttons, column headings, command-line commands and options, icons, dialog box titles, lists, menu names, tabs, menu commands and user input.
<i>Italic</i>	Indicates a placeholder for information or parameters that you must provide. For example, if the procedure asks you to type <i>filename</i> , you must type the actual name of a file. Italic also indicates new terms and the titles of other books.
ALL UPPERCASE	Represents keyboard keys (for example, CTRL, ENTER, F2).
Monospace	Represents text displayed at the command prompt and text file contents.
➤	Indicates a procedure with sequential steps.
■	Indicates a list of related information, not procedural steps.
{braces}	Encloses required items in syntax statements. For example, { yes no } indicates that you must specify yes or no when using the command. Type only the information within the braces, not the braces themselves.
[brackets]	Encloses optional items in syntax statements. For example, [<i>password</i>] indicates that you can choose to type a <i>password</i> with the command. Type only the information within the brackets, not the brackets themselves.
(vertical bar)	Stands for “or” and separates items within braces or brackets. For example, { /hold /release /delete } indicates that you must type /hold or /release or /delete.
... (ellipsis)	Indicates that you can repeat the previous item(s) in syntax statements. For example, /route: <i>devicename</i> [...] indicates that you can specify more than one device, putting commas between the device names.

Information that is specific to a particular UNIX platform is identified using the following symbols:

Symbol	Identifies instructions specific to the
AIX	IBM AIX Operating System
HP-UX	Hewlett-Packard HP-UX Operating System
Solaris	Sun Solaris Operating Environment

A symbol with a line through it indicates information that does not apply to a particular platform. For example, the following symbol is used to indicate information that does not apply to the HP-UX platform: ~~**HP-UX**~~

Note The examples and screens shown throughout the documentation are for the Solaris Operating Environment, unless indicated otherwise.

UNIX Command-line Conventions

MetaFrame for UNIX has a command line interface, so you type the commands to control the server at the command prompt.

If you are not familiar with UNIX command lines, note that:

- All UNIX commands are case sensitive.
- The spacing on the command line is important, and must be followed exactly as described in the instructions.

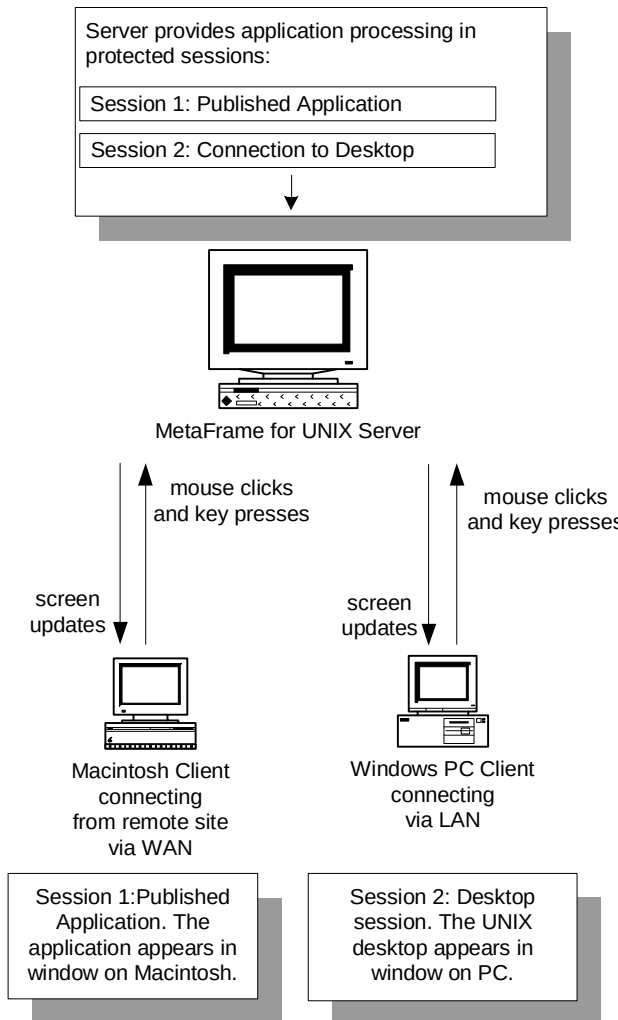
Note Only one instance of some MetaFrame commands should be run at any one time—these are the commands that cause configuration changes (rather than commands that just query and display information). If more than one instance runs simultaneously, you may get unpredictable results.

A Quick Overview

MetaFrame is a server-based software product that you can use to provide the users of your network with access to applications.

- You install MetaFrame on a UNIX machine that will be used as a server. MetaFrame allows multiple users to log on and run applications in separate, protected sessions on the same server. The applications you will provide must also be available on the server. You may want to install Word processors, Web browsers, Java applications, a particular window manager or custom applications.
- You install the ICA (Independent Computing Architecture) Client software on the client device, so users can connect to the MetaFrame server from a client device, such as a Windows PC. The ICA Client software is provided free, and is available for a range of different devices, allowing users to connect from various platforms.
- MetaFrame uses the ICA protocol to send information between the client device and the server. This sends the keystrokes, mouse clicks and screen updates between the server and the client. The application processing remains on the server, which means that processing on the client is kept to a minimum. To the user of the client device, it appears as if the software is running locally on the client.
- Because applications run on the server and not on the client computer, users can connect from any client device. A Macintosh, a Windows PC, or another UNIX machine can be used—the specification of which does not matter—the application looks and feels the same on each client device.

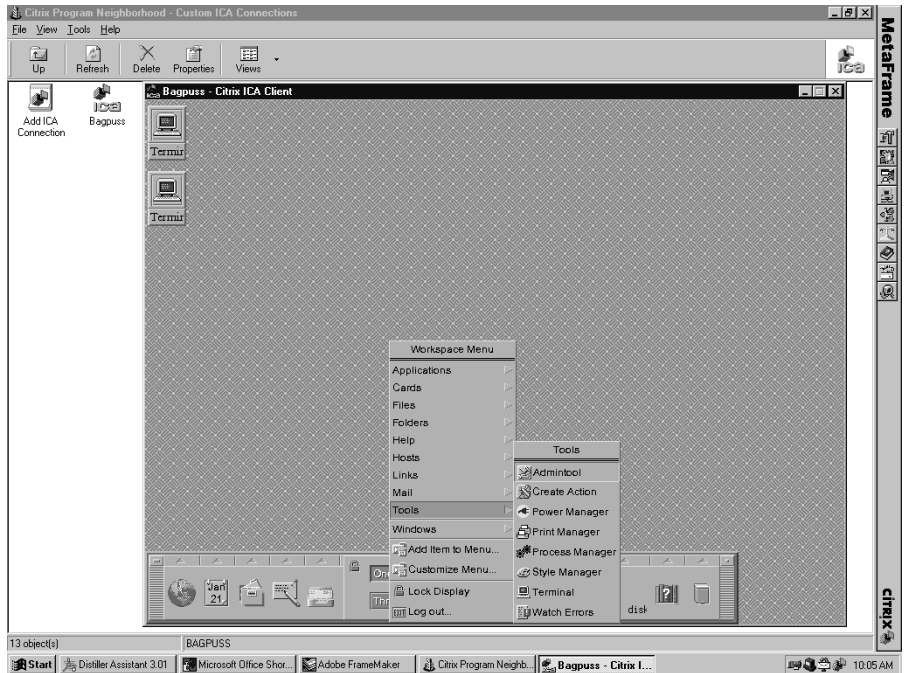
The following diagram shows a MetaFrame server with ICA connections:



MetaFrame Features

Connection from a client device

Connecting to an application or desktop usually simply involves clicking on an icon or application name from the client software on the client device. You can use the same client software on a client device to create connections to MetaFrame for UNIX and MetaFrame for Windows NT Application Servers.



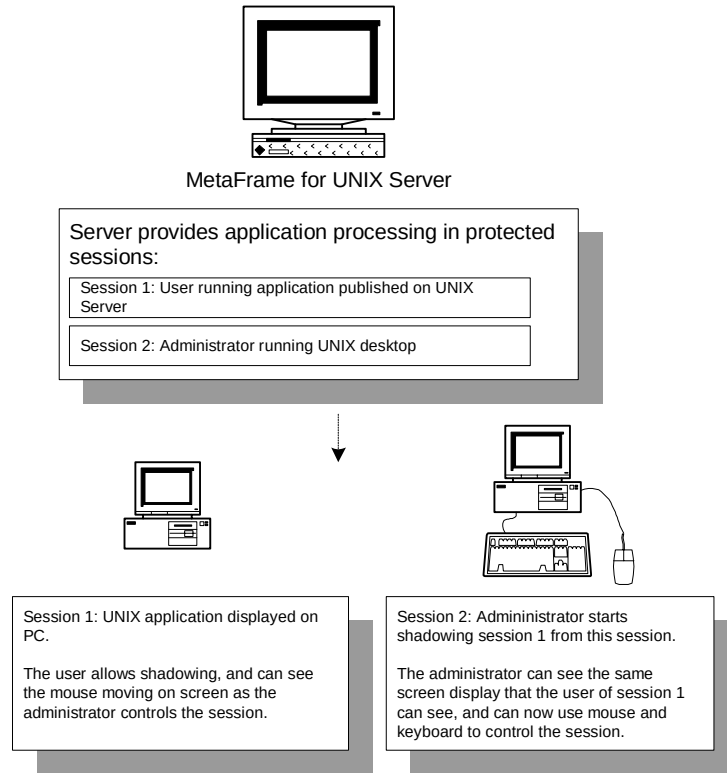
Connection to published applications

You can publish specific applications. A *published application* is a predefined application or shell script and its associated environment. Publishing applications is an efficient way to provide access to specific applications. This means users just connect to an application and the session closes when they exit the application.

Connection to UNIX desktops from client devices

You can provide full access to the UNIX server desktop. The user can run any application available on the desktop, in any order, or simultaneously. The server desktop appears in a window on the client device.

One MetaFrame license consumed per client device	When a user first connects to a server, one MetaFrame license is consumed. If the user of that client device connects to other servers or published applications during the session, additional licenses are not consumed. This means that a user can connect to a number of published applications while still using just the one license. See the “Citrix Licensing” chapter for further information.
Integration with UNIX security and accounts	MetaFrame uses the security set up on the UNIX server. Therefore, you do not need to set up new user accounts for MetaFrame—the user at the client device can log on using their existing UNIX user account and password. Solaris and HP-UX use Pluggable Authentication Modules (PAM) for user name and password validation; AIX uses its own authentication mechanism.
Special group and account for Citrix server administrators	MetaFrame requires you to create a special user group with the authority to run MetaFrame administration commands and start and stop the server. This is the Citrix server administrator group, which is called ctxadm. The user ctxsrvr must be created and added to this group.
Configurable permissions for access to MetaFrame features	You can control which users or groups of users can use particular MetaFrame features, such as logging on, disconnecting and sending messages to other sessions, using the MetaFrame security feature.
Guest user access	MetaFrame includes special anonymous user accounts with limited permissions. You can use these accounts to provide users with guest access to published applications and a temporary working directory for use during the session.
Shadowing user sessions	You can display and interact (using your keyboard and mouse) with another user’s session from your own session. This feature is called <i>shadowing</i>. You can use shadowing to help remote users with training or technical support issues. The following diagram shows the shadowing of one session from another:



Copying text and graphics between applications

Users can copy text and graphics between server-based applications and applications running locally on the client device. The clipboard behaves as if all applications are actually running on the client device itself.

Load balancing between servers

If you have Load Balancing licenses, you can publish the same application on a number of servers. Users connect to the published application and MetaFrame ensures that the connections are distributed among servers so that one particular server does not become overloaded.

Integrating With Other Citrix Servers

- MetaFrame for UNIX Operating Systems will co-exist with other Citrix servers (for example, MetaFrame for Windows 2000) on a network by sharing master browser information.
- Although you cannot add MetaFrame for UNIX Operating Systems servers to Citrix server farms, you can use a *pass-through* ICA Client to access applications on MetaFrame for UNIX using Program Neighborhood. ICA PassThrough technology allows non-Win32 ICA Clients to take advantage of the Citrix Program Neighborhood features. This is done by publishing the pre-installed ICA Client on a MetaFrame for Windows server and having clients “pass through” the server’s Citrix Program Neighborhood client to access a server farm.
- You can use the Citrix XML Service with NFuse to provide users with access to Windows and UNIX applications from one location. Citrix NFuse is an application portal technology that lets you integrate and publish applications into a Web browser from any standard Web server. The Citrix XML Service for MetaFrame for UNIX is available as a separate download package from the Citrix Web site with this release.

Note Cross-server administration between Windows and UNIX versions of MetaFrame is not possible.

Getting Started Quickly...

This section describes the minimum steps required to install and set up a MetaFrame server. For full details of how to install MetaFrame for UNIX, including step-by-step installation and configuration instructions, see Chapter 1 “Deploying MetaFrame”. We recommend that you read this chapter before installing MetaFrame for UNIX Operating Systems for the first time.

- **To install and get a MetaFrame server up and running, you need to**
1. Set up the accounts for the Citrix server administrator. Use the standard UNIX system tools to do this. At minimum, you need to log on as root and create a group called **ctxadm**, and a user called **ctxsrvr** (with **ctxadm** as the primary group).
 2. Install the MetaFrame for UNIX server software on your UNIX server using the appropriate installer, and then start MetaFrame on the server using the command `ctxsrv start`.
 3. Install the client software on each client device you plan to use from the ICA Client CD-ROM included in your MetaFrame for UNIX package, or from the Citrix website.

4. After installing the client software, create ICA connections to your MetaFrame server and test that you can connect from each type of ICA Client.

To find out how to	Read
Install the MetaFrame server software	"Deploying MetaFrame" in Chapter 1.
Install the client software on a particular client device	The installation section in the <i>Client Administrator's Guide</i> for the client you plan to deploy.
Create a connection from a client device to a server	The <i>Client Administrator's Guide</i> for the appropriate client.

Note There is a *Client Administrator's Guide* for each client in the documentation directory on the ICA Client CD-ROM in your MetaFrame for UNIX server package. The filename for the PDF refers to the client, for example ICAUNIX.PDF is the filename for the *Client Administrator's Guide* for the UNIX Client.

What To Do Next

When you can connect to your MetaFrame server from an ICA Client, your server is operational and you must activate the license.

MetaFrame is a configurable system that allows you a great deal of control over application deployment.

To find out how to	Read
Activate the base license on the server	"Citrix Licensing" in Chapter 2.
Publish applications	"Publishing Applications and Desktops" in Chapter 3.
Manage the MetaFrame server during typical use	"Managing Servers, Users and Sessions" in Chapter 4.
Control access to MetaFrame features	"Configuring MetaFrame Security" in Chapter 6.
Start using load balancing	"Load Balancing Published Applications" in Chapter 6.
Change the settings on the MetaFrame server	"Configuring a MetaFrame Server" in Chapter 5.

Getting More Information

Your MetaFrame software includes the following online documentation:

- The online version of the *MetaFrame Administrator's Guide*.
- Man pages, which can be displayed online, for each MetaFrame command line tool. These provide an overview of the command, warnings and important notes, and pointers to related commands.
- The *Citrix ICA Client Administrator's Guides* tell administrators how to install, configure, and deploy the various ICA Clients to end-users.



The documentation for MetaFrame is also available in Adobe PDF format in the documentation directory of the MetaFrame for UNIX and the ICA Client CD-ROM (except on the HP-UX platform). Using the Adobe Acrobat Reader, you can view and search the documentation electronically or print it for easy reference. To download the Adobe Acrobat Reader for free, please go to Adobe's Web site at: <http://www.adobe.com/>.

Important Please consult the **readme** file in the root directory of your CD-ROM, for any last-minute updates, installation instructions, and corrections to the documentation. On HP-UX, the readme file is available from the swinstall tool.

Finding Resources on the Citrix Web Site

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

- Downloadable Citrix ICA Clients, available at <http://download.citrix.com/>
- A Frequently Asked Questions page with answers to the most common technical issues
- An FTP server containing the latest service packs and hotfixes for download
- An Online Knowledge Base containing an extensive collection of technical articles, troubleshooting tips, and white papers
- Interactive online support forums
- The Citrix Developer Network (CDN) available at <http://www.citrix.com/cdn> This new, 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 Citrix server-based computing solutions into their products.
- A product documentation library, available at: <http://www.citrix.com/services/productdocs.asp>

Providing Feedback About this Guide

We strive to provide you with accurate, clear, complete, and usable documentation for Citrix products. If you have any comments, corrections, or suggestions for improving our documentation, we would be happy to hear from you. You can email the authors at:

`documentation@citrix.com`

Please include the name and version number of the product and the title of the document in your email.

CHAPTER 1

Deploying MetaFrame



Overview

This chapter describes how to install, deploy, and remove MetaFrame. Topics covered in this chapter include:

- System requirements
- Setting up the Citrix server administrator user and group
- Installing MetaFrame
- Setting up environment variables
- Starting and stopping MetaFrame
- Removing MetaFrame

Before You Begin

Please make sure that you read the following information before installing MetaFrame:

- “UNIX Operating System Requirements”—the UNIX Operating System must be installed before you install MetaFrame. You must also ensure that your operating system is configured to run MetaFrame, and that you install the required updates, as listed in this section.
- “Creating the Citrix Server Administrator User and Group”—you must set up **ctxadm** and **ctxsrvr** before installing MetaFrame.

Note Make sure that all users who connect to the server have a home directory path that is valid on the server, and that can be written to by the user. If a user has no home directory and tries to connect, the logon fails.

System Requirements

This section lists the minimum machine specifications and operating system requirements for MetaFrame for UNIX Operating Systems. This section also describes how to use ICA Client devices with non-English keyboards with your MetaFrame for UNIX server.

Minimum Machine Specifications

The minimum machine specification depends on how many connections are to be supported. As a general rule, we recommend that each server has between 16 and 24 MB of RAM per ICA connection. However, you may need to increase this amount of RAM depending upon the type of applications your users are running.

The following specifications are guidelines only.

Solaris On the Solaris Platform**1 to 3 connections:****Solaris SPARC**

Sparcstation 20

128MB RAM

Solaris Intel

P133

128MB RAM

More than 3 connections:**Solaris SPARC**

'Ultra-30'

UltraSPARC-II 248MHz

256MB RAM

Solaris Intel

PII 300

256MB RAM

HP-UX On the HP-UX Platform**1 to 3 connections:**

C110 120MHz PA-RISC

128MB RAM

More than 3 connections:

A400 440MHz PA-RISC

256MB RAM.

AIX On the AIX Platform**1 to 3 connections:**

43P Model 150

128MB RAM

More than 3 connections:

44P Model 270

256MB RAM

UNIX Operating System Requirements

Solaris On the Solaris Platform

The Solaris edition of MetaFrame for UNIX Operating Systems requires:

- Solaris 2.6 (also known as SunOS 5.6), SPARC version
- Or-
- Solaris 7 (also known as Solaris 2.7 and SunOS 5.7), SPARC or Intel versions
- Or-
- Solaris 8, SPARC or Intel versions

The server machine must have an X Window system installed with the appropriate window manager for the platform, for example, CDE.

The following operating system packages are required:

- SUNWxwoft X Window System optional fonts
- SUNWuiu8 Iconv modules for UTF-8 Locale

Check if these packages are installed using the **pkginfo** command.

Note On Solaris 7 and 8 these two packages are installed when you do an end-user install. On Solaris 2.6 the packages are not installed in an end-user install.

The Iconv libraries must be installed; they are necessary for MetaFrame to run. Check that the following files exist in the **/usr/lib/iconv** folder:

```
UCS-2*.so
UTF-8*.so
8859-1*.so
```

HP-UX On the HP-UX Platform

The HP-UX edition of MetaFrame for UNIX Operating Systems requires:

- HP-UX version 11.x

The server machine must have an X Window system installed with the appropriate window manager for the platform, for example, CDE.

The following operating system packages are required:

- PHSS_20298 Xserver cumulative patch
- PHSS_21493 X11 cumulative patch

AIX

On the AIX Platform

The AIX edition of MetaFrame for UNIX Operating Systems requires:

- AIX version 4.3.3

The server machine must have an X Window system installed with the appropriate window manager for the platform, for example, CDE.

Euro Currency Symbol Support

MetaFrame for UNIX supports the ISO 8859-15 Euro-currency symbol if the underlying UNIX operating system supports it. To ensure this support, you may need to install patches recommended by your operating system and hardware vendor. See the Web site for your operating system manufacturer or contact your hardware vendor for details of the appropriate patches and for instructions for ensuring Euro symbol support.

Installing MetaFrame

This section explains how to:

- Create the Citrix server administrator user and group
- Install MetaFrame for UNIX Operating Systems for the first time
- Upgrade from a previous release of MetaFrame
- Perform an unattended install that allows you to install MetaFrame without intervention

Installation Overview

You need to perform the following steps to install MetaFrame:

1. If you are installing MetaFrame for the first time, you need to create the Citrix server administrator user and group accounts.
2. Install MetaFrame from the CD-ROM.
3. If you are installing MetaFrame for the first time, you need to add the MetaFrame path(s) to your path, so that you can run the MetaFrame commands.
4. Start the MetaFrame processes on the server.

The following describes the installation process in more detail.

Important You must set up the Citrix server administrator group and user account before you install MetaFrame. The installation will fail if the **ctxadm** group and **ctxsrvr** user have not been created.

Creating the Citrix Server Administrator User and Group

If you are installing MetaFrame for the first time, you need to create the Citrix server administrator group account, and a user in this group, before you install MetaFrame. This account is required by some MetaFrame commands that demand special administration rights for MetaFrame, but do not require root access to the UNIX system.

- Create the Citrix server administrator's group using the group name **ctxadm**.
- Create a Citrix server administrator using the user name **ctxsrvr**. Make sure that you add the ctxsrvr user to the ctxadm group, and that the ctxadm group is its primary group.

Important Do not use the Citrix server administrator user or group for any purposes other than MetaFrame system administration.

Installing MetaFrame on Solaris

Solaris Installing MetaFrame for the First Time

This section describes how to install MetaFrame for the first time. Before you begin, ensure you have created the Citrix server administrator group account **ctxadm** and the user **ctxsrvr**.

With this method you are prompted for information—if you want to perform an installation that uses a response file see “Performing an Unattended Install” later in this section.

➤ **To install MetaFrame**

1. Log on as root at the server on which you want to install MetaFrame.
2. Mount the MetaFrame CD-ROM.
3. Change to the directory for the version of MetaFrame you want to install. For example, type:

```
cd /cdrom/mfunix/solaris_version
```

where *solaris_version* is the name of the directory on the CD-ROM for the platform architecture (SPARC or Intel) of Solaris you are using. The path is usually **/cdrom/mfunix/...** but it may change depending on how your system mounts the CD.

4. To add the MetaFrame package, type:

```
pkgadd -d CTXSmf
```

This starts the package installation script.

5. At the prompt for the startup/shutdown script installation, type **y** if you want to start MetaFrame when the machine is booted and stop it when the machine is shut down. If you answer yes, the script “S99ctxsrv” is installed in the /etc/rc2.d directory.
6. At the prompt for the man page installation, type **y** to install the MetaFrame man pages.
7. At the prompt for anonymous users, type **y** to create 15 anonymous user accounts if you want to enable guest access.
8. At the prompt about security settings for setuid/setgid, type **y** to set the correct file permissions for the MetaFrame files and processes.

Important Do not type **n**, or MetaFrame will not operate correctly.

9. At the next prompt, type **y** to continue installing MetaFrame. When complete, a message tells you that the installation was successful and the command prompt is displayed.

Note Do not attempt to share or copy the MetaFrame for UNIX installation files between servers. The configuration database cannot be duplicated, and you will experience problems if you attempt to do this.

Solaris Upgrading from a Previous Release of MetaFrame

This section explains how to install an upgrade of MetaFrame for UNIX Operating Systems. When you upgrade to a new version, your existing configuration settings, such as licensing and published application information, are preserved.

During the installation of the upgrade you are prompted for information, such as whether you want to install anonymous users. The installation also takes into account some of the settings you had in your previous release, such as whether the start-up script and man pages were installed, and it re-applies these settings.

With this method you are prompted for information; if you want to perform an installation that uses a response file, see “Performing an Unattended Install” later in this section.

Note During the upgrade installation, script files are overwritten with new versions that contain a checksum. If you have changed a script file since MetaFrame was last installed, the installation process detects this and the existing files are backed up before being replaced. The backed-up files have **.bak** and a version number, such as **v1.0**, appended to them. You will need to copy your changes into the new script files when the upgrade installation is finished.

➤ **To upgrade to a new release**

1. Ensure that there are no active sessions, and stop MetaFrame, using the **ctxshutdown** command—see “Stopping MetaFrame” in this chapter for more information.
2. Log on as root at the server on which you want to upgrade MetaFrame.
3. Mount the MetaFrame CD-ROM.
4. Change to the directory for the version of MetaFrame you want to install. For example, type:

```
cd /cdrom/mfunix/solaris_version
```

where *solaris_version* is the name of the directory on the CD-ROM for the platform architecture (SPARC or Intel) of Solaris you are using. The path is usually **/cdrom/mfunix/...** but it may change depending on how your system mounts the CD.

5. To install the upgrade, type:

```
pkgadd -d CTXSmf -a /cdrom/mfunix/default CTXSmf
```

where **/cdrom/mfunix/default** is the sample administration defaults file used by the **-a** option of the **pkgadd** command. This is included on the MetaFrame CD-ROM for your convenience—if you want to use different default settings, copy and change this, as appropriate.

Note Ensure you specify the **-a** option or you will get the error message: *“Current administration requires that a unique instance of the <CTXSmf> package be created. However, the maximum number of instances of the package which may be supported at one time on the same system has already been met. No changes were made to the system”*.

6. At the prompt, type **y** to confirm you want to upgrade. This starts the package installation script.
7. At the prompt for anonymous users, type **y** to create 15 anonymous user accounts if you want to enable guest access.
8. At the prompt about security settings for setuid/setgid, type **y** to set the correct file permissions for the MetaFrame files and processes.

Important Do not type **n**, or MetaFrame will not operate correctly.

9. At the next prompt, type **y** to continue installing MetaFrame. When complete, a message tells you that the installation was successful and the command prompt is displayed.
10. Reapply any changes you made to the script files by copying your changes from the backed-up files to the new script files.

Solaris Performing an Unattended Install

This section explains how to set up a script and a response file so that you are not prompted for information during installation. Using this method, you can install MetaFrame quickly and easily on multiple servers. You can use this procedure to install MetaFrame for the first time or to upgrade from a previous release.

➤ To perform an unattended install

1. Log on as root at the server on which you want to install MetaFrame.
2. Create a *response* file that will be used by the **-r** option of the **pkgadd** command—for example: **/tmp/CTXSmf11**. In this file include the following:

```
CLASSES=none app sts man
ANONYORN=y
```

The **CLASSES** variable specifies what to install:

- **none** and **app** are mandatory.
- If **sts** is specified, the startup script is installed.
- If **man** is specified, the man pages are installed.

The **ANONYORN** variable specifies whether anonymous users are added (**y**=yes; **n**=no).

3. Create a script file to perform the unattended install, for example:

```
#!/bin/sh
# change permissions for root to execute this file - chmod 744
pkgadd -r /tmp/CTXSmf11 -a /cdrom/mfunix/default -d /cdrom/mfunix/
solaris_version/CTXSmf CTXSmf
```

where *solaris_version* is the name of the directory on the CD-ROM for the platform architecture (SPARC or Intel) of Solaris you are using, and **/cdrom/mfunix/default** is the sample administration defaults file used by the **-a** option of the **pkgadd** command. This is included on the MetaFrame CD-ROM for your convenience—if you want to use different default settings, copy and change this, as appropriate. The path is usually **/cdrom/mfunix/...** but it may change depending on how your system mounts the CD.

4. Run the script file to start the unattended install.

Installing MetaFrame on HP-UX

HP-UX This section describes how to install MetaFrame on an HP-UX platform. Before you begin, ensure you have created the Citrix server administrator group account **ctxadm** and the user **ctxsrvr**.

With this method you are prompted for information—if you want to perform an installation that does not prompt you see “Performing an Unattended Install on HP-UX” later in this section.

➤ **To install MetaFrame on HP-UX**

1. Log on as root at the server on which you want to install MetaFrame.
2. Insert the MetaFrame CD-ROM into the CD-ROM drive.
3. At the command prompt, type **swinstall**. The **SD Install** program begins and the **Specify Source** dialog is displayed.
4. In **Source Depot Type**, specify **Local CD-ROM**. The system auto-detects the CD-ROM and displays a path. Choose **OK**.
5. In the **SD Install** dialog, select **MetaFrame**.
6. To install the entire MetaFrame package (including man pages, 15 anonymous user accounts and the startup script) choose **Mark for Install** from the **Actions** menu.

Alternatively, to install particular components, expand **MetaFrame** to display the following filesets, then mark the filesets you want to install:

Fileset	Description
Anon	Choose to create 15 anonymous user accounts. You cannot install this fileset on its own—the Runtime fileset must also be installed.
Man	Choose to install the MetaFrame manual pages. You cannot install this fileset on its own—the Runtime fileset must also be installed.
Runtime	Choose to install the MetaFrame runtime environment (the programs and the configuration database).
Startup	Choose if you want to start MetaFrame when the machine is booted and stop it when the machine is shutdown. If you choose this fileset, the script ctxsrv is installed in the /sbin/init.d directory, and two symbolic links are added: - the startup link S999ctxsrv is installed in /sbin/rc3.d - the shutdown link K001ctxsrv is installed in /sbin/rc2.d You cannot install this fileset on its own—the Runtime fileset must also be installed.

7. From the **Actions** menu, choose **Install (analysis)** to display analysis information prior to the installation. If warnings are generated, display the **Logfile** for further details.
8. Choose **OK**.
9. At the confirmation prompt, choose **Yes** to complete the installation.

Note Do not attempt to share or copy the MetaFrame for UNIX installation files between servers. The configuration database cannot be duplicated, and you will experience problems if you attempt to do this.

HP-UX Performing an Unattended Install on HP-UX

This section explains how to perform an unattended install of MetaFrame on an HP-UX platform. With an unattended install, you are not prompted for information during the installation, so you can install MetaFrame quickly and easily on multiple servers.

➤ **To perform an unattended install of MetaFrame**

1. Log on as root at the server on which you want to install MetaFrame.
2. Insert the MetaFrame CD-ROM into the CD-ROM drive and mount it as a read-only filesystem. For example, at the command prompt type:

```
mount -r /dev/dsk/c0t0d0 /mnt/cdrom
```

where `/dev/dsk/c0t0d0` is the file that identifies the CD-ROM drive and `/mnt/cdrom` is the mount point of the CD-ROM.
3. To install the entire MetaFrame package (including man pages, 15 anonymous user accounts and the startup script), at the command prompt, type:

```
swinstall -s /mnt/cdrom MetaFrame
```

Alternatively, list the particular filesets you want to install. For example, to install MetaFrame and the man pages, at the command prompt, type:

```
swinstall -s /mnt/cdrom MetaFrame.Runtime MetaFrame.Man
```

Installing MetaFrame on AIX

AIX

This section describes how to install MetaFrame on an AIX platform using SMIT. Before you begin, ensure you have created the Citrix server administrator group account **ctxadm** and the user **ctxsrvr**.

With this method you are prompted for information—if you want to perform an installation that does not prompt you see “Performing an Unattended Install on AIX” later in this section.

➤ **To install MetaFrame on AIX**

1. Log on as root at the server on which you want to install MetaFrame.
2. Insert the MetaFrame CD-ROM into the CD-ROM drive.
3. Type **smit**. The **System Management Interface Tool** dialog is displayed.
4. Choose **Software Installation and Maintenance**.
5. Choose **Install and Update Software**.
6. Choose **Install and Update Software by Package Name (includes devices and printers)**. A dialog is displayed.
7. Choose **List**. The **Single Select List** dialog is displayed.
8. Select the CD-ROM drive.
9. Choose **OK**. The **Multi-select List** dialog is displayed.
10. Select **Citrix** and choose **OK**.
11. To install the entire MetaFrame package (including man pages, 15 anonymous user accounts and the startup script) select **Citrix.MetaFrame**.

Alternatively, to install particular components, select the filesets you want to install:

Citrix. MetaFrame...	Fileset description
.boot	Choose if you want to start MetaFrame when the machine is booted and stop it when the machine is shutdown. If you choose this fileset, the daemon ctxmfd is installed in /usr/lpp/CTXSmf/sbin, and starts up automatically. During the installation of the .boot fileset, an entry is made in the /etc/inittab file that starts up ctxmfd and the Metaframe server, when booting up. You cannot install this fileset on its own—the Citrix.MetaFrame.rte fileset must also be installed.

- | | |
|--------------|---|
| .anon | Choose to create 15 anonymous user accounts. You cannot install this fileset on its own—the Citrix.MetaFrame.rte fileset must also be installed. |
| .rte | Choose to install the MetaFrame runtime environment (the programs and the configuration database). |
| .man | Choose to install the MetaFrame manual pages. |
12. Choose **OK**. The **Install and Update Software by Package Name (includes devices and printers)** dialog is displayed.
 13. Choose **OK** to begin the installation.
 14. At the prompt, choose **OK** to continue installing MetaFrame. When complete, check the **Installation Summary** to make sure that the installation was successful.
 15. To exit from smit, select **Exit SMIT** from the **Exit** menu.

Note Do not attempt to share or copy the MetaFrame for UNIX installation files between servers. The configuration database cannot be duplicated, and you will experience problems if you attempt to do this.

AIX

Performing an Unattended Install on AIX

This section explains how to perform an unattended install of MetaFrame on an AIX platform. With an unattended install, you are not prompted for information during the installation, so you can install MetaFrame quickly and easily on multiple servers.

➤ To perform an unattended install of MetaFrame

1. Log on as root at the server on which you want to install MetaFrame.
2. Insert the MetaFrame CD-ROM into the CD-ROM drive.
3. To install the entire MetaFrame package (including man pages, 15 anonymous user accounts and the startup script), at the command prompt, type:

```
installp -X -d/dev/cd0 Citrix.MetaFrame
```

where *-d/dev/cd0* is the CD-ROM device, and *-X* ensures that there is sufficient disk space to install the package.

Alternatively, list the particular filesets you want to install. For example, to install MetaFrame man pages, at the command prompt, type:

```
installp -X -d/dev/cd0 Citrix.MetaFrame.man
```

Setting the Paths to MetaFrame Commands

There are two types of MetaFrame commands:

User commands

Any user can run these commands. They include the MetaFrame commands for logging off and disconnecting from a server.

User commands are installed in:

/opt/CTXSmf/bin **Solaris** **HP-UX**

/usr/lpp/CTXSmf/bin **AIX**

System administration commands

Only the **ctxsrvr** user (or members of the **ctxadm** group) can run these commands. They include server, published application, and ICA Browser configuration tools.

Administration commands are installed in:

/opt/CTXSmf/sbin **Solaris** **HP-UX**

/usr/lpp/CTXSmf/sbin **AIX**

Configuring User Access to Commands

Generally, you do not have to do anything to allow users to run user commands from their sessions. The path to these commands is added to each user's path upon connection to the server, so any user can access MetaFrame user commands from an ICA session.

However, you may have to configure access to MetaFrame commands if the user's shell script startup file (for example, **.profile** or **.login**) overrides the path. For example, on HP-UX, the default system profile (**/etc/profile**) sets the **PATH** environment variable explicitly.

➤ To configure user access to MetaFrame commands

- If you are using a C shell, use a **.login** file for the user, and add the path to the user commands. For example:

HP-UX

Solaris

AIX

```
setenv PATH ${PATH}:/opt/CTXSmf/bin
```

```
setenv PATH ${PATH}:/usr/lpp/CTXSmf/bin
```

- If you are using a Bourne or similar shell, use a **.profile** file for the user, and add the path to the user commands. For example:

HP-UX

Solaris

AIX

```
PATH=${PATH}:/opt/CTXSmf/bin
```

```
export PATH
```

```
PATH=${PATH}:/usr/lpp/CTXSmf/bin
```

```
export PATH
```

Configuring ctxsrvr Access to Commands

A Citrix server administrator needs to be able to run both user and system administration commands. If you are installing MetaFrame for the first time, you need to configure your system so that the **ctxsrvr** user can run all the commands from the MetaFrame server console, and also from an ICA session.

➤ To configure ctxsrvr access to MetaFrame commands

- If you are using a C shell, use a `.login` file for the `ctxsrvr` user, and add the path to the user and administrator commands. For example:

HP-UX

Solaris

```
setenv PATH ${PATH}:/opt/CTXSmf/sbin:/opt/CTXSmf/bin
```

AIX

```
setenv PATH ${PATH}:/usr/lpp/CTXSmf/sbin:/usr/lpp/CTXSmf/bin
```

- If you are using a Bourne, or similar, shell use a `.profile` file for the `ctxsrvr` user, and add the path to the user and administrator commands. For example:

HP-UX

Solaris

```
PATH=${PATH}:/opt/CTXSmf/sbin:/opt/CTXSmf/bin
export PATH
```

AIX

```
PATH=${PATH}:/usr/lpp/CTXSmf/sbin:/usr/lpp/CTXSmf/bin
export PATH
```

Setting the Path to the man pages

Generally, you do not have to do anything to allow users to display man pages for MetaFrame commands from a session. The path to these files is added to every user's `MANPATH` environment variable upon connection to the server.

However, you may have to configure access to the MetaFrame man pages if the user's shell script startup file (for example, `.profile` or `.login`) overrides the path. For example, on HP-UX, the default system profile (`/etc/profile`) sets the `MANPATH` environment variable explicitly.

To display the MetaFrame man pages from the server console when you log on as `ctxsrvr`, you must set up your `MANPATH` environment variable to point to the location of the installed man pages. You need to do this only if you are installing MetaFrame for the first time.

➤ To set the MANPATH environment variable

- If you are using a C shell:

HP-UX

Solaris

```
setenv MANPATH ${MANPATH}:/opt/CTXSmf/man
```

AIX

```
setenv MANPATH ${MANPATH}:/usr/lpp/CTXSmf/man
```

- If you are using a Bourne shell:

HP-UX

Solaris

```
MANPATH=${MANPATH}:/opt/CTXSmf/man
export MANPATH
```

AIX

```
MANPATH=${MANPATH}:/usr/lpp/CTXSmf/man
export MANPATH
```

Starting and Stopping MetaFrame

Starting MetaFrame

When installation is complete, you can start the MetaFrame process on each server using the **ctxsrv** command.

➤ **To start MetaFrame**

1. Log on at the MetaFrame server as a Citrix server administrator (for example, the default user ctxsrvr).
2. At the command prompt, type:

```
ctxsrv start
```

Note If, during installation, you chose to add the startup/shutdown script, MetaFrame will automatically start when the machine is booted.

Stopping MetaFrame

To stop the MetaFrame process on a server, use the **ctxshutdown** command. With ctxshutdown you can specify when the shut down process will begin, and notify users that the server is about to shut down. This allows users to save their work and log off gracefully.

When the shut down process begins, applications will terminate, except for those that have registered window hints. These applications will attempt to interactively log users off by displaying a series of prompts.

With ctxshutdown, you can specify the maximum duration that users have to respond to these prompts. Any sessions that are still active when this period expires are terminated and the users are automatically logged off.

The server prevents users from logging on during the shut down process.

➤ **To stop MetaFrame**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Shut down the server using the defaults. By default, the server shutdown process begins after 60 seconds; the message “Server shutting down. Auto logoff in 60 seconds” is sent to all users logged on to the server. Applications that have registered window hints (the WM_DELETE_WINDOW attribute) have a further 30 seconds to interactively log users off before terminating.	<code>ctxshutdown</code>
Operate in quiet mode. This reduces the amount of information displayed to the administrator by the <code>ctxshutdown</code> command.	<code>ctxshutdown -q</code>
Specify when the shut down process will begin, and how long the message will be displayed, in seconds. The default is 60 seconds. When this period expires and the shut down process begins, applications that have registered window hints (the WM_DELETE_WINDOW attribute) will attempt to interactively log the user off. Applications that have not registered window hints will terminate immediately.	<code>ctxshutdown -m <i>seconds</i></code>
Specify how long applications that have registered window hints (the WM_DELETE_WINDOW attribute) have to interactively log users off. The default is 30 seconds. When this period expires, any remaining sessions are automatically terminated, users are automatically logged off, and the MetaFrame process stops.	<code>ctxshutdown -l <i>seconds</i></code>
Specify the message displayed to all users logged on to the server. If you do not specify a message, the default message “Server shutting down. Auto logoff in x seconds” is displayed, where x = the number of seconds specified in the <code>-m</code> option (or the default of 60 seconds if this is not specified).	<code>ctxshutdown <i>message</i></code>

Example

The following example shows how to display a message and begin the shut down process after two minutes. Applications that have registered window hints are given a further three minutes to attempt to interactively log users off.

```
ctxshutdown -m 120 -l 180 "Please log off now"
```

About ICA Client Keyboard Support

MetaFrame for UNIX supports ICA Client devices that use the following keyboards:

Language	Locale ID
US English	409
UK English	809
French	40c
German	407
Swedish	41d
Spanish	40a
Italian	410

Configuring Non-English Keyboard Support

Your users can make connections to the MetaFrame server with ICA Client devices that use non-English keyboards. The keyboards that MetaFrame supports are shown in the previous table.

➤ **To configure non-English keyboard support**

1. Ensure you start the server in the country locale of the ICA Client keyboard that your users are using. For example, if your users have German keyboards, start the server in a German locale. This ensures that the session runs in an appropriate locale where fonts containing the required keyboard symbols are in the font path and keyboard symbols appear correctly on the screen.
2. Make sure your users select the appropriate keyboard in the **Settings** dialog on the client device. For further information about selecting keyboards, refer to the *Client Administrator's Guides* for the clients you are deploying.

Tip You can alter the locale for an individual user by setting environment variables in their start-up files—see “About Customizable Scripts and Files” in Chapter 5 for further information.

Troubleshooting Non-English Keyboard Support

If users experience problems obtaining accent symbols, such as the circumflex accent (^), it may be that the application they are using does not support *dead keys*. A dead key is a key that does not produce a character when pressed—instead, it modifies the character produced by the next key press. For example, on a generic French PC keyboard the “^” key, (circumflex) is a dead key. When this key is pressed, and then the “a” key is pressed, â is generated.

Configuring MetaFrame Event Logging

When you first install MetaFrame on a UNIX system, events are not configured to be sent to the system log (syslog).

MetaFrame uses the following event log levels:

- user.notice
- user.info
- user.warning
- user.err
- user.debug

To record MetaFrame events, you can add a line to the **/etc/syslog.conf** file and specify the event log levels that you want to record. You must be root to edit syslog.conf.

Note The event log level names that MetaFrame uses may also be used by other programs. You may see messages from other software in the event log.

For example, adding the following line to the end of syslog.conf (separated with a tab, not a space) causes all event log messages from MetaFrame for UNIX Operating Systems to be put in the file `/var/adm/messages`:

AIX

Solaris

```
user.notice;user.info /var/adm/messages
```

HP-UX

```
user.notice;user.info /var/adm/syslog/syslog.log
```

Note The file that you use (e.g. `/var/adm/messages`) must exist. If it does not, then create it.

You may also want to send certain types of MetaFrame event details to the console. For example, to ensure that all MetaFrame error messages appear on the console, add this line to the file `/etc/syslog.conf`:

```
user.err /dev/console
```

For more details about configuring system event logging, see the `syslog.conf` man page.

Removing MetaFrame

Solaris ➤ To remove MetaFrame on Solaris

1. Log on as a Citrix server administrator at the MetaFrame server.
2. Ensure that there are no active sessions and stop MetaFrame, using the **ctxshutdown** command. See “Stopping MetaFrame” in this chapter for more information.
3. Log on as root.
4. To remove MetaFrame, type:
`pkgrm CTXSmf`

HP-UX ➤ To remove MetaFrame on HP-UX

1. Log on as a Citrix server administrator at the MetaFrame server.
2. Ensure that there are no active sessions and stop MetaFrame, using the **ctxshutdown** command. See “Stopping MetaFrame” in this chapter for more information.
3. Log on as root.
4. To remove MetaFrame, type:
`swremove`
5. The **SD Remove** dialog is displayed. Choose **MetaFrame**.
6. From the **Actions** menu, choose **Mark for Remove**.
7. From the **Actions** menu, choose **Remove (analysis)** to display analysis information prior to the installation. If any warnings are generated, display the **Logfile** for further details.
8. Choose **OK** to remove MetaFrame.

Tip To quickly remove the entire MetaFrame package, at the command prompt, type: **swremove MetaFrame**

AIX ➤ To remove MetaFrame on AIX

1. Log on as a Citrix server administrator at the MetaFrame server.
2. Ensure that there are no active sessions and stop MetaFrame, using the **ctxshutdown** command. See “Stopping MetaFrame” in this chapter for more information.
3. Log on as root.
4. Type **smit**. The **System Management Interface Tool** dialog is displayed.
5. Choose **Software Installation and Maintenance**.
6. Choose **Software Maintenance and Utilities**.

7. Choose **Remove Installed Software**. The **Remove Installed Software** dialog is displayed.
8. In **SOFTWARE name**, type **Citrix.MetaFrame**
Or, to remove a particular fileset, type in its name—for example **Citrix.MetaFrame.man**.

Note If you want to remove the **Citrix.MetaFrame.rte** fileset, you must also remove the **Citrix.MetaFrame.boot** and **Citrix.MetaFrame.anon** filesets. If you do not, a ‘Dependency Failure’ occurs.

9. Set **PREVIEW only?** to **no**.
10. Choose **OK**.
11. At the prompt, choose **OK** to confirm you want to remove the software. When complete, check the **Installation Summary** to make sure that the removal was successful.

Note If the removal of MetaFrame fails it may be because you did not stop the server—see Step 2.

12. To exit from smit, select **Exit SMIT** from the **Exit** menu.

Tip To quickly remove the entire MetaFrame package, at the command prompt, type: **installp -u Citrix.MetaFrame**.

Reinstalling MetaFrame

You may need to reinstall MetaFrame if:

- You (or another user) have accidentally deleted MetaFrame files, such as binary files. If this happens, you need to reinstall MetaFrame but *preserve* your configuration so that information such as licensing and published application data is retained. To do this, simply follow the instructions for “Installing MetaFrame”.
- The configuration database is corrupted. For example, you may experience problems displaying or modifying settings such as Citrix licensing, or you may see “Internal Error” messages. If you are in doubt about whether you have a corrupt configuration database, contact Citrix support. You may need to reinstall MetaFrame and overwrite your configuration files. To do this, remove MetaFrame (see “Removing MetaFrame” for information) and then reinstall MetaFrame by following the instructions in “Installing MetaFrame”. You will need to reactivate your Citrix licenses.

AIX

Note When you reinstall MetaFrame on the AIX platform, ensure you set the **OVERWRITE same or newer versions** box in the **Install and Update Software by Package Name (includes devices and printers)** dialog to **yes**, or MetaFrame will not reinstall.

HP-UX

When you reinstall MetaFrame on the HP-UX platform, ensure you select the **Reinstall filesets even if same revision exists** option in the **Options** dialog (available from the **Options/Change Options** menu in swinstall) or MetaFrame will not reinstall.

CHAPTER 2

Citrix Licensing



Overview

This chapter explains Citrix Licensing. Topics in this chapter include:

- An introduction to Citrix Licensing and why you need to activate licenses
- How to activate a Citrix License
- Displaying information about licenses on a Citrix server
- Adjusting the pooling of license user counts across servers
- Removing a Citrix License

What Is Citrix Licensing?

Citrix Licensing controls the number of ICA connections permitted on your MetaFrame server and the features available on the server. To complete the installation of MetaFrame, you need to activate your Citrix licenses to permit ICA connections to the server and enable the software.

To further increase the user count or enable additional features on the server, you can purchase and activate additional licenses.

License Types

There are two types of Citrix license:

Base licenses

Every Citrix server comes with a base license. The base license enables the multiuser features of your Citrix server and can include a user count. If the base license is not present, ICA connections are not supported and server extension licenses cannot be added.

Server extension licenses

Server extension licenses increase the user count (and are sometimes called “user licenses”) or enable additional functionality, such as load balancing. Upgrade licenses are a type of server extension license.

The Grace Period

Some licenses have a 35-day grace period after installation. During this period, the MetaFrame software is fully functional and connections to the server work normally. However, periodic reminder messages are generated to remind you to activate the license within x number of days.

If the license is not activated by the end of the grace period, the MetaFrame software is automatically reduced to single user mode, and only one ctxsrvr user can log on to the server at a time.

About User Counts

Base licenses and user licenses have a user count. A server's user count is the number of ICA Client users who can have a session on that server simultaneously.

Pooling User Counts

Citrix user counts can be shared (pooled) by all servers on the same network subnet. Each server contributes its installed user count to the master ICA Browser.

For example, if Server A's user count is 15 and Server B's user count is 15, a total of 30 is available for use by either server. For example, Server A can support up to 20 users as long as Server B serves no more than 10. You can adjust how licenses are pooled on a given server—see “Adjusting the Pooled User Count” in this chapter. MetaFrame servers that pool licenses must be on the same subnet.

By default, all user counts are pooled.

Note User counts are not pooled across ICA Gateways.

Client Device Licensing

Client device licensing allows users to start multiple sessions on the same server or on different servers, while using only a single Citrix user count. All connections must be from the same client device.

When a user starts a second session on the *same* Citrix server as the first session, the new session does not consume a second user count.

When a user starts a second session on a *different* Citrix server, the new session does not consume a second user count under the following conditions:

- The first session consumed a pooled user count
- The user makes all connections from the same client device
- All servers are on the same subnet (using the same master ICA Browser)

Important Citrix servers exhaust all local (unpooled) user counts before consuming pooled user counts. Therefore, a user assigned a local user count uses a second user count when starting a second session on a different Citrix server.

Activating a License

There are three steps to activating a Citrix license:

1. Add the supplied *serial number* using **ctxlicense**. The serial number is the 25-character number on the sticker on the back of the CD booklet. When you add a serial number, MetaFrame generates a unique 35-character *license number*, based on the serial number.
2. Get an *activation code* from Citrix for the license. This step uses the license number generated in Step 1.
3. Activate the Citrix license using **ctxlicense**. You must provide the license number and activation code from previous steps.

The following section describes these steps in detail.

Step 1—Adding the License Serial Number

To add a license, you need to supply the serial number from the back of the CD booklet.

➤ **To add a license serial number**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxlicense -add serial-number
```

Type the serial number exactly as it appears on the serial number sticker on the back of the CD booklet (it is case sensitive). A message tells you that your Citrix license has been successfully added. MetaFrame generates and displays a unique 35-character license number, based on the serial number you entered.

3. Copy the license number to the clipboard, or make a note of it, because you will need it later.



Note If you install a base license for an older version of MetaFrame, you will be prompted for an upgrade license at this point.

Step 2—Getting an Activation Code

When you have added the license, you need to contact Citrix to get an activation code. The activation code is a number that validates and enables the Citrix license. The activation code is based on the license number generated when you added the license (see Step 1).

➤ **To get an activation code from the Citrix website**

Go to the Citrix product activation Web site: www.citrix.com/activate/ and follow the instructions on screen. You will need to supply the 35-character license number generated in Step 1.

Step 3—Activating the License

When you have added the license and obtained an activation code from Citrix, you are ready to activate the license. You need to supply both the license number and activation code in this step.

➤ **To activate a license**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxlicense -activate license-number activation-code
```

Paste in the license number from the clipboard, or type it in, and type in the activation code supplied by Citrix. A message telling you that your Citrix license has been successfully activated appears. The software is now ready for use.

Installing an Upgrade License

Solaris If you are upgrading from a previous version of MetaFrame, you need to add and activate the upgrade license after you install the upgrade software. *Only when you have installed the base license of the previous version and installed and activated the upgrade license is your MetaFrame server properly licensed.*

➤ **To install and activate an upgrade license**

1. Ensure your previous base license is installed—see “Displaying Information About Licenses” for details about how to check this. If the previous base license is not installed, add it now. See “Step 1—Adding the License Serial Number” for further information.
2. Install and activate the upgrade license—follow the three steps described in “Activating a License” earlier in this chapter. When you activate the upgrade license, the base license is automatically activated.

If you add a base license for an older version of MetaFrame, or you add an upgrade license on its own (without a previous base license), MetaFrame displays a warning message to prompt you for the missing license. However, you can suppress these warning messages and add the licenses individually.

➤ **To install licenses individually and activate an upgrade license**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Add the base and upgrade licenses in any order using the **-noprompt** option. At the command prompt, type:

```
ctxlicense -add -noprompt serial-number
```
3. Activate the upgrade license. See “Step 3—Activating the License” for further information.

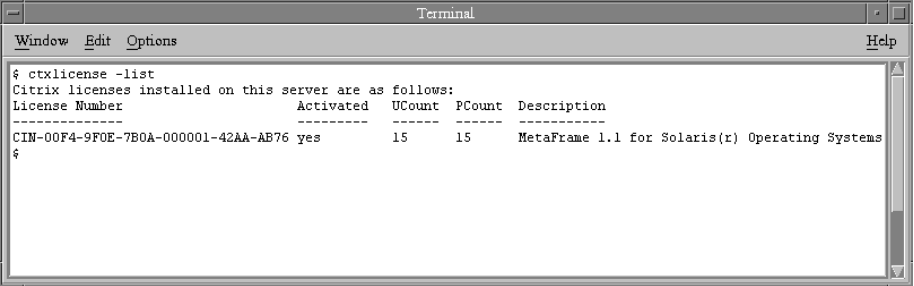
Displaying Information About Licenses

You can use **ctxlicense** to display information about all the licenses present on the Citrix server. A description of each license, the license number, user count and pooled user count, and an indication of whether it is activated or not, is provided.

➤ **To display license information**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxlicense -list
```



```
Terminal
Window Edit Options Help
$ ctxlicense -list
Citrix licenses installed on this server are as follows:
License Number      Activated   UCount   PCount   Description
-----
CIN-00F4-9F0E-7B0A-000001-42AA-AB76 yes         15       15       MetaFrame 1.1 for Solaris(r) Operating Systems
$
```

Note You can also display information about licenses on Citrix servers on the network using the **ctxqserver** command—see the “MetaFrame Command Reference” in Appendix A for further information.

Adjusting the Pooled User Count

By default, all user licenses are pooled across Citrix servers. You can reserve licenses for use only on a local Citrix server by lowering the number of pooled licenses. Un-pooled local licenses are not available to other Citrix servers and cannot be used for client device licensing.

To change the pooled user count you need to supply the 35-character license number. See “Displaying Information About Licenses” in this chapter for details about how to display license numbers.

➤ **To change the pooled user count across Citrix servers**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxlicense -pool license-number pooled-count
```

The pooled count must be between zero and the total number of user counts installed with this license number. Any remaining user counts become local to this server only.

Removing a License

You can use **ctxlicense** to remove a license from a Citrix server. You may want to do this so that you can use the license on a different server.

To remove a license you need to supply the 35-character license number—see “Displaying Information About Licenses” in this chapter for details about how to display license numbers.

➤ **To remove a Citrix license**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxlicense -remove license-number
```

3. Type **yes** to confirm the removal of the license.

Note If you remove a license and then use it on a different server, you need to get a new activation code from Citrix—the old activation code will not work.

Removing an Upgrade License



You must remove an upgrade license first, *before* removing the base license. If you remove an upgrade license, the server will no longer be properly licensed, even if the base license is present.

CHAPTER 3

Publishing Applications and Desktops



Overview

This chapter describes how to provide access to applications for ICA Client users. Topics in this chapter include:

- An introduction to application publishing
- Displaying the applications published on a server
- Publishing applications, desktops, shell scripts, and UNIX command lines
- Publishing applications on UNIX servers of different architecture
- Publishing applications for explicit and anonymous use
- Maintaining published applications
- Configuring an initial program
- Deploying applications that require a 3-button mouse

About Published Applications

To an ICA Client user, a published application is an application that appears similar to an application running locally on the client device.

Published applications:

- Give ICA Client users easy access to applications running on Citrix servers
- Increase your control over application deployment
- Shield users from the complexities of the UNIX environment hosting the ICA session

The **ctxappcfg** command is the main tool for publishing applications. You can publish any application that can run on the UNIX workstation or server on which MetaFrame is installed.

Why Publish Applications?

The main reasons for publishing applications are the ease of user access, the degree of administrative control, and the efficient use of resources.

Administrative Control

When you publish applications, you get greater administrative control over application deployment.

- **Load Balancing.** Application publishing, when used in conjunction with Citrix Load Balancing Services, lets you direct ICA Client connection requests to the least busy server in a group of servers configured to run an application.
- **Enabling and Disabling Applications.** You can disable published applications without having to delete their configuration. This allows you to temporarily stop users from connecting to published applications. A disabled application can be quickly enabled at a later stage.

User Access

When you publish applications, user access to those applications is greatly simplified in the following areas:

- **Addressing.** Instead of connecting to a Citrix server by its IP address or server name, ICA Client users can connect to a specific application or desktop by whatever name you give it. Connecting to applications by name eliminates the need for users to remember which servers contain which applications.

- **Navigation of the server desktop.** Instead of requiring ICA Client users to have knowledge of the UNIX desktop to find and start applications after connecting to Citrix servers, published applications present the user with only the desired application in an ICA session.

All ICA Client users can benefit from application publishing's simplified addressing and desktop navigation when they configure connections to published applications using their ICA Client.

Efficient Use of Resources

ICA connections to server desktops can consume considerable resources because, by default, CDE is loaded for each connection. For more efficient use of server resources, use published applications rather than server desktops.

Publishing Applications for Explicit or Anonymous Use

When you publish an application, you have to tell MetaFrame if the application is for anonymous or explicit use.

Publishing Applications for Explicit Use

An explicit user is a person who has his or her own user account.

If you publish an application for use by explicit users, when the users log on, they supply their username and password. Explicit users have a “permanent” existence—their desktop and security settings are retained between sessions and their files persist from one session to another.

Publishing Applications for Anonymous Use

Publishing applications for anonymous use allows you to provide “guest” user access to an application.

When a user starts an application published for anonymous use, no login box is displayed and the user does not have to supply a username or password. Instead, the server selects an available account from a pool of anonymous user accounts and assigns this to the user.

A temporary home directory is also assigned to the user for use during the session. Users do not have a persistent identity, and no information in the home directory is retained when they log off. Any desktop settings, user-specific files, or other resources created or configured by the user are discarded at the end of the ICA session.

If the session is idle (in other words, if there is no user activity for a specified time period), the session is terminated. Users are logged off after a broken connection or timeout.

Important Take care when choosing applications to publish anonymously, as no username or password is required to access these applications and therefore little meaningful audit data can be obtained. We recommend that you do not publish applications that will provide the user with a command shell, as the user may be able to access and affect the system in the same way as an explicit user.

For information about how to change or maintain anonymous user accounts, see “Configuring Anonymous Users” in Chapter 6.

For information about setting up configuration files for applications published for anonymous use, see “Publishing Pre-Configured Applications for Anonymous Use” in this chapter.

Note The total number of users, whether anonymous or explicit, who can be logged on to the MetaFrame server at the same time depends upon your licensed user count. See Chapter 2 for details.

Configuring Published Applications

This section covers how to:

- Publish an application, shell script, desktop, or UNIX command line
- Publish an application on a UNIX server of different architecture
- Display details of the applications that are published on a server
- Change the configuration of a published application
- Enable and disable published applications
- Publish applications on multiple servers
- Restrict connections to published applications only
- Delete published applications

Publishing an Application, Shell Script, or Desktop

Publishing an Application

Use the **ctxappcfg** command to publish an application. The command prompts you for the information required to publish the application.

Application installation is not part of the application publishing process. Before an application can be published, both MetaFrame and the application must be installed. The order in which you install the application and MetaFrame does not matter. Once an application is installed, it can be published at any time.

➤ **To publish an application**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxappcfg
```

You will see the following prompt:

```
App Config>
```
3. At the App Config prompt, type **add**. When you add a new application, the program requests each item of information you need to supply:

At the prompt	Type
Name:	The name you want to use for the published application. The user selects this name when setting up an ICA connection to this published application. The name does not need to be the same as the name of the executable file for a particular program.
Command line:	The command line required to run the application or script file, for example: /usr/bin/diary.bin .
Working directory:	The default working directory. This directory must exist. Leave blank to specify the user's home directory.
Anonymous [yes no]	y if the application is for anonymous use only, or n if it is for use by users with explicit accounts only.

4. At the App Config prompt, type **exit**.

The published application is automatically enabled. You can now connect to the server from an ICA Client and set up a connection to this published application.

You can change any parameter for a published application at any time—see “Changing the Settings of a Published Application” later in this chapter for more details.

Note To publish an application for both explicit and anonymous use, publish it under different names—once for explicit use, and once for anonymous use.

Publishing a Shell Script

You can also publish an application by writing a script file that sets up the application environment and then executes the application. You then publish the shell script file as a published application, using the **ctxappcfg** command. To publish a script file, enter the name of the script file at the command line prompt.

Publish a shell script if you want to publish an application that requires a particularly complex environment—for example, if you need to set particular environment variables.

Publishing a Desktop

You publish a server desktop in the same way as you publish an application, using the **ctxappcfg** command. However, to indicate you are publishing a desktop, you leave the command line blank.

Note ICA connections to server desktops consume considerable server resources because, by default, CDE is loaded for each connection. For more efficient use of server resources, use published applications rather than server desktops.

Publishing a Java Application

You can publish Java applications on your MetaFrame server by writing a script file that you publish using the **ctxappcfg** command. In the script file include any environment variables required to set up the application environment, and the commands to launch the Java application.

Publishing a UNIX Command Line

You can publish applications that only require use of the command line. For example, you may have a legacy application that you want to publish. You do this using the **ctxappcfg** command. At the command prompt, type:

```
xterm -e "commands"
```

where *commands* is the set of commands required to launch the application. Enclose the commands within double quotes. If the set of commands is complex, include this in a script file and run the script file:

```
xterm -e script_file
```

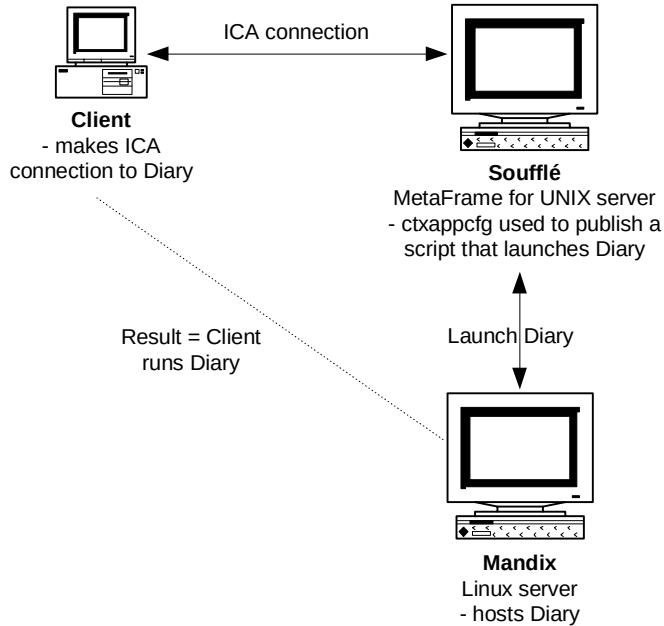
Publishing an Application on a UNIX Server of Different Architecture

You can publish applications on UNIX servers of a different architecture to your MetaFrame for UNIX server. For example, you can publish an application on your MetaFrame for UNIX server, although the application exists and runs on a Linux server.

To do this, you create a script file to run the application on the remote UNIX server. Then you create a script file on the MetaFrame server to set up the application environment and launch the application on the remote server. This script uses the remote shell command to run the script on the remote server and the **DISPLAY** environment variable to display back onto the MetaFrame server. Finally, you publish the script on the MetaFrame server using the **ctxappcfg** command.

Example

The following example shows how to publish an application on a MetaFrame for UNIX server that runs on a Linux server. In this example, the MetaFrame server is called 'Soufflé', the Linux server is called 'Mandix' and the application is called 'Diary'.



➤ Step 1—Create a script file on Mandix

1. Install Diary on Mandix.
2. Create a script file on Mandix that will run Diary on Mandix. For example, create a script file `/usr/local/bin/rundiary.sh` containing:

```
#!/bin/sh
DISPLAY=$1
shift
export DISPLAY
cd /export/home/apps/diary
/export/home/apps/diary/diary $*
```
3. Make sure that the script file works, by testing locally on Mandix.

➤ **Step 2—Create a script file on Soufflé**

1. Create a script file on Soufflé that will set up the application environment and launch **rundiary.sh** on Mandix. For example, create a script file **/export/home/apps/diary.sh** containing:

```
#!/bin/sh
my_machine=`uname -n`
DISPLAY=`echo $DISPLAY |sed "s/.*/:$my_machine:/"`
# allow everyone to access this display
xhost +
# launch app on the machine "Mandix"
rsh Mandix "/usr/local/bin/rundiary.sh $DISPLAY ~/group.cal"
```

HP-UX

Note On HP-UX, the remote shell command is **remsh**.

2. Make sure that the script file works on Soufflé, by testing that it correctly launches the application on Mandix, using a display on Soufflé. `~/group.cal` is the parameter passed to the diary application on Mandix.

➤ **Step 3—Publish the application on Soufflé**

1. Create a script file on Soufflé that uses the **ctxappcfg** command to publish **diary.sh**. For example:

```
ctxappcfg <<EOF
add
MY_DIARY
/export/home/apps/diary.sh

n
exit
EOF
```

2. Test that the script file works by making an ICA connection to MY_DIARY.

Displaying Details About Published Applications

- **To display details about the applications published on a server**
 1. Log on to the MetaFrame server as a Citrix server administrator.
 2. At the command prompt, type **ctxappcfg**. This starts the program and displays the following prompt:
App Config>
 3. At the command prompt, type **list**. This displays the names of the applications published on the server:
App Config> list
Name: Accounts
Name: Orders
Name: Diary
Applications that have been disabled have **(disabled)** displayed next to them.
 4. To find out more details about a particular published application, use the **select** command with the name, for example:
App Config> select Diary
This displays the details for the published application, for example:
Name: Diary
Command line: /usr/bin/diary.bin
Working directory: ~/tmp
Anonymous [yes|no]: yes
Enabled [yes|no]: yes
 5. If you want to list information for a different application, type **drop** to de-select the current application. You can then use the **select** command again with the appropriate application name.
 6. To exit from ctxappcfg, type **exit**.

Note You can also display information about published applications on the network using the **ctxqserver** command—see the “MetaFrame Command Reference” in Appendix A for more information.

Changing the Settings of a Published Application

After you publish an application, you can change the configuration settings by running the **ctxappcfg** utility again.

➤ **To change the details for a published application**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxappcfg**. This starts the program and displays the following prompt:

```
App Config>
```

3. At the command prompt, type **list** to check the names of the applications published on the server.

4. Select the published application you want to change, for example:

```
App Config> select Diary
```

5. This displays the details for the published application, for example:

```
Name: Diary
Command line: /usr/bin/diary.bin
Working directory: ~/tmp
Anonymous [yes|no]: yes
Enabled [yes|no]: yes
```

6. To change the configuration, use the **set** command with the following syntax:

```
set [cmd=[cmd_line], dir=[dir_name], anonymous=[yes|no],
enabled=[yes|no]]
```

Option	Description
cmd	The command line required to run the application or script file, for example: /usr/bin/diary.bin .
dir	The default working directory. This directory must exist. Leave blank to specify the user's home directory.
anonymous	Type y if the application is for anonymous use only, or n if it is for use by users with explicit accounts only.
enabled	Type n to disable an application—this stops users from connecting to the application without deleting its configuration. Type y to enable a previously disabled application.

7. When you are finished configuring the published application, type **save** to save the changes.
8. To exit from **ctxappcfg**, type **exit**.

Enabling and Disabling Published Applications

You can disable a published application without having to delete its configuration. This is useful when you want to temporarily stop users from connecting to a published application. A disabled application can be quickly enabled at a later stage.

When you disable a published application, the master browser is updated and users are no longer able to see or connect to the disabled application.

Note When you publish an application, it is automatically enabled.

➤ **To enable or disable a published application**

1. Use the **ctxappcfg** utility with the **set** command, as described previously in “Changing the Settings of a Published Application”.
2. Set the **enabled** option to **n** to disable an application, or set it to **y** to enable a previously disabled application.

Creating a New Published Application from Existing Details

After you publish an application, you can re-use the settings by copying the details to a new name.

➤ **To copy details to create a new published application**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxappcfg**. This starts the program and displays the following prompt:
App Config>
3. At the command prompt, type **list** to check the names of the applications published on the server.
4. Select the published application that has the details you want to copy, for example:
App Config> select Diary
5. This displays the details for the published application, for example:
Name: Diary
Command line: /usr/bin/diary.bin
Working directory: ~/tmp
Anonymous [yes|no]: yes
Enabled [yes|no]: yes
6. Type **copy** and the new name for the published application at the prompt. Type **drop**.
7. Change the details for the new published application using the **set** command.

8. When you are finished configuring the published application, type **save** to save the changes.
9. To exit from **ctxappcfg**, type **exit**.

Renaming a Published Application

After you publish an application, you can change the name by copying the settings to a new name, then deleting the original.

➤ To rename a published application

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxappcfg**. This starts the program and displays the following prompt:
App Config>
3. At the command prompt, type **list** to check the names of the applications published on the server.
4. Select the published application you want to rename, for example:
App Config> select Diary
5. This displays the details of the published application, for example:
Name: Diary
Command line: /usr/bin/diary.bin
Working directory: ~/tmp
Anonymous [yes|no]: yes
Enabled [yes|no]: yes
6. Type **copy** and the new name for the published application at the prompt. Type **drop**.
7. To delete the original published application settings, select the original application and use the **Delete** command. See “Deleting a Published Application” later in this chapter for more details.

Publishing an Application on More than One Server

You may want to publish the same UNIX application or shell script on more than one MetaFrame server. For example, you might publish an application on a number of servers, and then use load balancing to balance the load among the servers.

When you publish the same application on other servers, the location and working directory does not have to be the same—the application can be run in different locations, from different install directories. However, if the location and working directory are kept the same, you can use a script file to publish the application on other servers. You still need to install the application on each server before running the script file.

Note If you are going to use Citrix Load Balancing Services to balance the user load among servers, the name of the published application must be identical on each server.

➤ **To publish an application on multiple servers**

1. Install and publish the application on the first server.
2. Run **ctxappcfg** and type **list** to display the names of the applications published on the server. Select the name of the application you want to publish and make a note of the settings that **ctxappcfg** displays.
3. Create a script file using the **ctxappcfg** settings from the previous step. For example, type:

```
#!/bin/sh
/opt/CTXSmf/sbin/ctxappcfg <<EOF
add
Diary
/usr/bin/diary.bin
/work/diary
n
exit
EOF
```

AIX

Note On the AIX platform, substitute **/opt/CTXSmf/sbin/ctxappcfg** with **/usr/lpp/CTXSmf/sbin/ctxappcfg**.

4. Make sure that the script file is executable.
5. Install the application you want to publish on the other servers. Make sure that you install it in the same location as on the first server.
6. Run the script file on each server.

Restricting Connections to Published Applications Only

You can restrict users so that they can connect only to published applications on a MetaFrame server. Doing so prevents users from connecting to the server by name, or to the server desktop.

➤ **To restrict connections to published applications only**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Use the **ctxcfg** command to allow users to run only published applications:
`ctxcfg -i PUBONLY`

Note To restrict access on several servers, you must run `ctxcfg -i PUBONLY` at each server.

Deleting a Published Application

Deleting a published application removes all published application configuration information from the server. When you delete a published application, that application is no longer available to ICA Client users under the published application name (although it may be available as another published application or from a Citrix server desktop session).

Tip To temporarily stop users from connecting to a published application, disable the published application. Disabling a published application does not delete its configuration, and it can be quickly enabled at a later stage. See “Changing the Settings of a Published Application” in this chapter for further information.

If you want to make the application available again, republish it under its old name or with a new name.

➤ **To delete a published application**

1. Run **ctxappcfg**. At the prompt, type **list** to display the names of the applications published on the server.
2. Select the published application you want to delete, for example:
`App Config> select Diary`
3. Type **delete**.
4. Confirm the deletion by typing **y**.
5. Type **exit**.

Note If you have published an application on more than one server, you must delete its published application configuration from each server.

Configuring an Initial Program

An initial program is an application that MetaFrame starts automatically when a user logs on. Closing the initial program does not terminate the ICA session.

Initial programs:

- Can be set on the MetaFrame server by a Citrix server administrator.
- Can be set from an ICA Client device as part of the properties for a specific ICA Client connection. If an initial program is configured on the server and client, the initial program configured on the server is started when a user logs on.

➤ To configure an Initial Program on the MetaFrame server

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Configure the server so that if an initial program is set on the client, it is used.	<code>ctxcfg -i INHERIT</code>
Configure the server to start the initial program <i>progrname</i> whenever a user connects.	<code>ctxcfg -i prog=<i>progrname</i>,wd=<i>dir</i></code>
List the current initial program details.	<code>ctxcfg -i list</code>

Deploying Applications that Require a 3-Button Mouse

You may need to use MetaFrame to deploy UNIX applications that are designed for use with a 3-button mouse. However, many ICA Clients run on devices that have only a 2-button mouse, 1-button mouse, or pointing device available.

To do this, you publish another version of the application for use by these ICA Clients. This version of the application is published using a script file that includes **ctx3bmouse** settings. The `ctx3bmouse` command lets users represent a missing mouse button by combining an existing mouse button with a modifier key. For example, a missing button might be simulated by clicking the left mouse button and pressing the SHIFT key.

By running a script file that includes `ctx3bmouse` settings, you ensure the application is run in a session with the appropriate mouse mappings.

The following explains how to publish applications with mouse mappings.

Publishing an Application With Mouse Mappings

There are three steps to publishing an application with mouse mappings:

1. Identify which mouse buttons are available and which are missing.
2. Decide which mouse button and modifier key will represent the missing mouse button.
3. Publish the application using a script file that sets up the mouse mappings and then executes the application.

Step 1—Identify the Mouse Buttons

To identify the mouse buttons on the different types of mouse:

- On a 3-button mouse there are left, middle, and right buttons
- On a 2-button mouse there are left and right buttons
- On a 1-button mouse there is a left button only

The left and right buttons on a 3-button mouse are mapped automatically to the left and right buttons on a 2-button mouse. The left button on a 2-button or 3-button mouse is mapped automatically to the left button on a 1-button mouse.

For devices that have only a 1-button mouse (such as a Macintosh mouse) or a pointing device, please refer to the appropriate client documentation for information about emulating mouse buttons.

Step 2—Decide How to Emulate the Missing Mouse Button

You can emulate the right, middle, or left mouse buttons on a 3-button mouse, using the `ctx3bmouse` command.

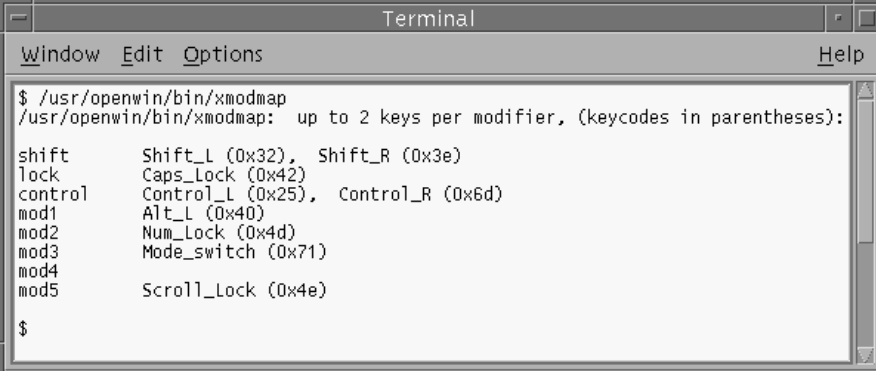
➤ To emulate a missing mouse button

Decide which mouse button and modifier key combination will represent the missing mouse button. To identify the number of the modifier key to use, run the **xmodmap** command to display a list of modifier keys for the system. `xmodmap` is located in:

Solaris `/usr/openwin/bin/xmodmap`

AIX **HP-UX** `/usr/bin/X11/xmodmap`

In the following example, the first modifier key is SHIFT, the second is CAPS LOCK, and so on.



```

Terminal
Window Edit Options Help
$ /usr/openwin/bin/xmodmap
/usr/openwin/bin/xmodmap: up to 2 keys per modifier, (keycodes in parentheses):

shift      Shift_L (0x32),  Shift_R (0x3e)
lock       Caps_Lock (0x42)
control    Control_L (0x25),  Control_R (0x6d)
mod1       Alt_L (0x40)
mod2       Num_Lock (0x4d)
mod3       Mode_switch (0x71)
mod4
mod5       Scroll_Lock (0x4e)
$
  
```

Step 3—Publish the Application With Mouse Mappings

When you have identified how to represent the missing mouse button, you publish the application using a script file that sets up the mouse mappings and then executes the application.

➤ To publish an application that includes mouse mappings

1. Create a script file that maps the missing mouse button using `ctx3bmouse` and then executes the application. Use the `ctx3bmouse` command with the following syntax:

```
ctx3bmouse missing-button=mouse-button,number-of-modifier-key
```

For example, to execute an application with mouse mappings that emulate the missing middle button of a 2-button mouse, using the SHIFT key and the left mouse button:

```
ctx3bmouse middle=left,1  
/usr/bin/2bmouse_diary.bin
```

2. Save the script file, for example as:

```
/usr/bin/2bmouse_diary.sh
```

3. Publish the script file:

```
Name: 2bmouse_Diary  
Command line: /usr/bin/2bmouse_diary.sh  
Working directory: ~/tmp  
Anonymous [yes|no]: yes
```

For further information about publishing applications and script files, see “Publishing an Application, Shell Script, or Desktop” in this chapter.

Important You need to tell your users the mouse mappings set in the script file.

Publishing Pre-Configured Applications for Anonymous Use

When a user logs on to use an application that you have published for anonymous use, MetaFrame assigns the user an empty home directory. MetaFrame deletes any files that the user creates in this directory when the user logs off.

Some applications use configuration files that initialize settings when the application starts. For example, an application such as a Web browser may use proxy settings, file paths, and font and display settings. In normal use, a user configures these settings once. If the configuration files are not available, the applications starts in its default configuration.

You can set up configuration files for applications that you publish for anonymous use. You create these in a special template directory called **/usr/anon/anontmpl**. MetaFrame copies all files in the template directory to the assigned home directory when a user logs on.

➤ To create configuration files for an application

1. Create a user (for example, called “anontmpl”) with home directory set to **/usr/anon/anontmpl**. Use this user account only to pre-configure applications for anonymous use.
2. Log on to the server as this user and run the application you want to configure.

3. Configure the application so that it mirrors the settings you want to provide when an anonymous user logs on. For example, for a Web browser application such as Netscape, you may want to set proxy server settings or clear the cache.
4. Exit the application.
5. Start the application again and make sure that the application works as required. If not, adjust the process and repeat until you are sure that the correct configuration is in use when the application starts.
6. Using `grep` or a text editor, search for occurrences of the user name or folder name (in this example, “anontmpl”) in each of the files in `/usr/anon/anontmpl`.
7. Make the template directory readable by everyone using:


```
chmod -R a+rX
```
8. Log on as a Citrix server administrator.
9. Edit the script file **ctxanoninit.sh**. This file is installed in the following directory:

HP-UX**Solaris**`/opt/CTXSmf/lib`**AIX**`/usr/lpp/CTXSmf/lib`

10. For each file containing occurrences of “anontmpl” in the files in `/usr/anon/anontmpl`, add lines to the end of **ctxanoninit.sh** that use the **sed** command to substitute the user name and home directory.

For example, a Netscape preferences file contains references to the home directory, so add the following lines to the end of `ctxanoninit.sh`:

```
sed -e "s,anontmpl,$USER," $ANON_TMPL_DIR/.netscape/preferences.js >
newprefs.js
rm .netscape/preferences.js
mv newprefs.js .netscape/preferences.js
```

Note Use the environment variable `$USER`, which is set automatically by `/bin/sh`, to determine the name to substitute.

11. Publish the application for anonymous use. Make sure that the application works by launching a session from an ICA Client, repeating the above steps as necessary.

CHAPTER 4

Managing Servers, Users, and Sessions



Overview

This chapter describes how to manage the users, sessions, and processes on a MetaFrame server. It includes how to:

- Display information about sessions and users
- Display information about the MetaFrame servers on the network
- Log off, disconnect, and reconnect sessions
- Reset sessions in case of error
- Shadow ICA sessions
- Send messages to users on your server
- Display available client printers and print files from the command line or from applications

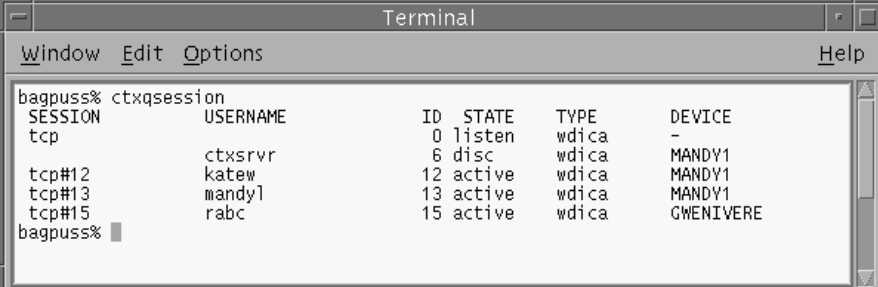
Displaying Information About Users and Sessions

You can display information about ICA connections to the Citrix server, such as user name, session name, session id, and state.

➤ **To display session details**

Run the **ctxqsession** command:

```
ctxqsession
```



A terminal window titled "Terminal" with a menu bar containing "Window", "Edit", "Options", and "Help". The command prompt is "bagpuss%". The output of the "ctxqsession" command is a table with the following data:

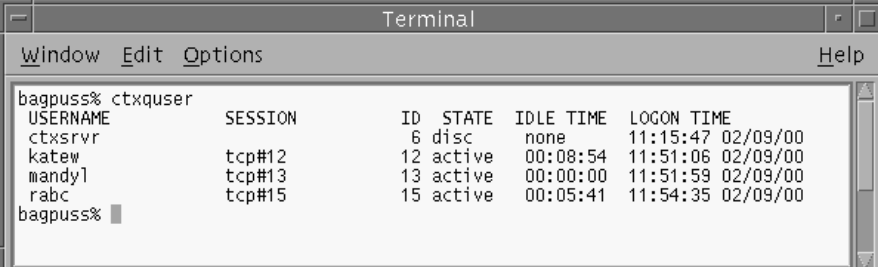
SESSION	USERNAME	ID	STATE	TYPE	DEVICE
tcp	ctxsrvr	0	listen	wdica	-
tcp#12	katew	12	active	wdica	MANDY1
tcp#13	mandyl	13	active	wdica	MANDY1
tcp#15	rabc	15	active	wdica	GWENIVERE

If you want information, by username, for all the current sessions, use the **ctxquser** command.

➤ **To display session details, by username**

Run the **ctxquser** command:

```
ctxquser
```



A terminal window titled "Terminal" with a menu bar containing "Window", "Edit", "Options", and "Help". The command prompt is "bagpuss%". The output of the "ctxquser" command is a table with the following data:

USERNAME	SESSION	ID	STATE	IDLE TIME	LOGON TIME
ctxsrvr		6	disc	none	11:15:47 02/09/00
katew	tcp#12	12	active	00:08:54	11:51:06 02/09/00
mandyl	tcp#13	13	active	00:00:00	11:51:59 02/09/00
rabc	tcp#15	15	active	00:05:41	11:54:35 02/09/00

Understanding the Display

The **ctxqsession** and **ctxquser** commands display:

Display	Description
SESSION	The session name.
USERNAME	The user name.
ID	The session id.
STATE	<code>listen</code> —indicates the session that is listening for new incoming connections. <code>active</code> —indicates an established, active connection. <code>connq</code> —indicates a brief session initialization phase that occurs before the login prompt appears. <code>init</code> —a brief session initialization phase. <code>conn</code> —indicates a session that is being connected. <code>disc</code> —indicates a disconnected session. <code>down</code> —indicates a broken session. <code>shadow</code> —indicates that the user of this session is shadowing another. <code>reset</code> —indicates a session currently being reset.
TYPE	<code>wdica</code> —indicates that the ICA protocol is being used.
DEVICE	The name of the ICA Client device.
IDLE TIME	The length of time since there was user activity on this session.
LOGON TIME	The time the user logged onto the system.

Displaying Information About Servers on the Network

Use the **ctxqserver** (query server) command to display information about Citrix servers on the subnet. You can display information such as server name and network address, transport protocol, and the number of connections available.

➤ **To display information about all Citrix servers on the subnet**

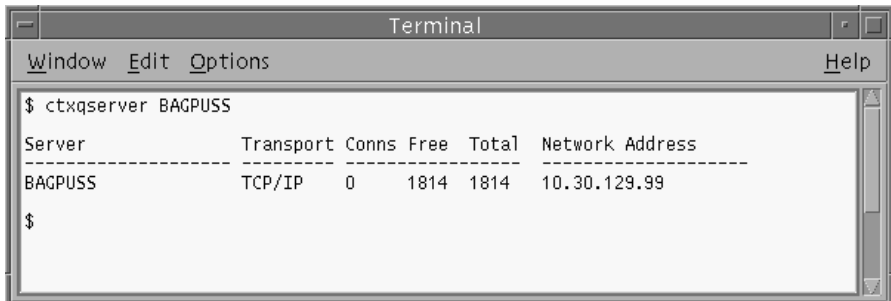
At the command prompt, type:

```
ctxqserver
```

➤ **To display information about a specific Citrix server**

At the command prompt, type **ctxqserver** and specify the server name:

```
ctxqserver server-name
```



Understanding the Display

The **ctxqserver** command displays:

Display	Description
Server	The server name.
Transport	The transport protocol; i.e. TCP/IP.
Conns	The current number of ICA connections on the server.
Free	The remaining number of connections the server is capable of receiving.
Total	The current number of ICA connections plus the number of free connections.
Network Address	The IP address of the server. An M next to the IP address indicates that the server is the master browser.

Note You can also use `ctxqserver` to display information specific to Citrix servers (such as ICA gateways and Citrix Licensing), or about published applications and ICA Client sessions on the subnet. You can also use `ctxqserver` to send requests to servers. For information about the other options available with `ctxqserver`, see the “MetaFrame Command Reference” in Appendix A.

Ending a Session

To end a session, you can use commands that either log off or disconnect the session.

- Disconnecting a session terminates the connection between the server and client. However, the user is not logged off and all running programs remain active, and the user can later reconnect to the disconnected session.
- Logging off a session terminates the connection and all running programs, and the user cannot reconnect to the session.

Logging Off a Session

Use the **ctxlogoff** command to log off a session.

➤ **To log off from your own session**

Type **ctxlogoff**.

➤ **To log off another user’s session**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxqsession** to display the current sessions at this server.
3. From the results of `ctxqsession`, identify the session id or session name of the connection session you want to forcibly log off.
4. At the command prompt:

To	Use the command
Log off a session by specifying the session id	<code>ctxlogoff <i>id</i></code>
Log off a session by specifying the session name	<code>ctxlogoff <i>name</i></code>

Disconnecting a Session

Use the **ctxdisconnect** command to disconnect a session.

➤ **To disconnect your own session**

Close the client or type **ctxdisconnect** at the command prompt.

➤ **To disconnect another user's session**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxqsession** to display the current sessions at this server.
3. From the results of ctxqsession, identify the session id or session name of the session you want to disconnect.
4. At the command prompt:

To	Use the command
Disconnect a session by specifying the session id	<code>ctxdisconnect <i>id</i></code>
Disconnect a session by specifying the session name	<code>ctxdisconnect <i>name</i></code>

If a user logs on to the server and there is a disconnected session on the server belonging to him or her, the user is given a choice of whether to connect to the disconnected session or start a new session.

Note You cannot disconnect an anonymous user session. This is because it is not possible to reconnect to the session because the identity of the user is not known.

Connecting to a Disconnected Session

Use the **ctxconnect** command, from within an ICA session, to reconnect to a disconnected session.

A user can connect to a previously disconnected session by logging on again with the same username. Once logged on, if there are disconnected sessions on the server, the user can reconnect to the disconnected session or start a new session.

A Citrix server administrator can connect to any user's session. Other users can connect only to their own sessions.

➤ **To connect to a disconnected session**

1. At the command prompt, type **ctxqsession**. This displays the current sessions at this server. A disconnected session shows **disc** in the **State** field.
2. From the results of the **ctxqsession** command, identify the session id associated with the session to which you want to connect.
3. At the command prompt, from within an ICA session, type:

```
ctxconnect id
```

where *id* is the session id of the session to which you want to connect.

The server disconnects your current session and connects you to the selected session.

Note Your connected session must be capable of supporting the video resolution used by the disconnected session. If the session does not support the required video resolution, the operation fails.

Resetting a Session

You can reset a session in the event of an error. However, the system will attempt to terminate all processes running within that session. Resetting a session may cause applications to close without saving data.

Use the **ctxreset** command to reset a session.

➤ **To reset a session**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type **ctxqsession**. This command displays the current sessions at this server.
3. From the results of the **ctxqsession** command, identify the session name or session id you want to reset.
4. At the command prompt:

To	Use the command
Reset a session by specifying the session id	<code>ctxreset <i>id</i></code>
Reset a session by specifying the session name	<code>ctxreset <i>name</i></code>

Reconnecting to Load Balanced Sessions

Published applications allow users to run applications or access a desktop session without knowing the name or address of a particular server. If the published application is located on a single server, users can disconnect and reconnect to the same session.

If the published application is configured to run on multiple servers, users must be reconnected to the same server to reconnect to their session. The ICA Browser can reconnect users to their previous session on the same server under certain conditions.

For a user to reconnect to disconnected load balanced sessions:

- The user must disconnect gracefully from the server, for example, by using **ctxdisconnect**.
- The user must reconnect from the same Citrix ICA Client device (using the same client name).

Use **ctxqsession** to view a list that displays disconnected sessions—see “Displaying Information About Users and Sessions” in this chapter.

Note If users frequently disconnect and reconnect their sessions rather than logging off, the number of sessions on a server farm may not be evenly distributed because users are reconnected to their previous sessions on the same servers.

Shadowing a User's Session

You can monitor the actions of users, and interact with their sessions, using the keyboard and mouse. This is called *shadowing*.

Use the **ctxshadow** command to shadow another user's session:

```
ctxshadow {id|name} [-v] [-h[[a][c][s]+]x]
```

The person who issues the `ctxshadow` command is called the *shadower*, and the session being shadowed is called the *shadowed session*.

Important Only ICA sessions can be shadowed or used to shadow ICA sessions. The MetaFrame server console cannot be shadowed or used to shadow ICA sessions.

The **ctxshadow** command is a user command, rather than a system administration command. Therefore, any user can shadow any other session, provided the shadowed user approves the shadowing, and MetaFrame security permits the user to shadow. Disabling shadowing notification means that a user may be unaware that shadowing is occurring. See “Enabling or Disabling Shadowing” in Chapter 5 for more information.

Note The following procedure assumes that you will use the CTRL key and * (asterisk) key combination to end shadowing. If you cannot use this hotkey combination, or you want to use an alternative combination to end shadowing, see “Ending Shadowing” later in this section.

➤ **To start shadowing a session**

1. Log on to an ICA session.
2. At the command prompt, type **ctxqsession**. This command displays the current sessions at this server.
3. From the results of the ctxqsession command, identify the session name or session id of the user’s session that you want to shadow.
4. At the command prompt, type **ctxshadow**. You must specify either a session id or session name. Use the -v (verbose) argument to display more information during the shadow session initiation, for example:

```
ctxshadow tcp#5 -v
```

The user is notified of the pending shadowing, and is given the opportunity to allow or deny the shadowing (unless notification has been disabled for shadowing using **ctxcfg**—see “Enabling or Disabling Shadowing” in Chapter 5 for details). If the user does not respond to the notification message, the shadow request times out and is terminated.

About Shadowing and the Clipboard

The user of the shadowed session can use the clipboard to copy and paste between the client session and applications running locally. As shadower, you cannot access the contents of the shadowed session’s clipboard—information in the clipboard belongs to the shadowed session. However, if you copy information to the clipboard while shadowing, this information is available to the shadowed session for pasting.

Ending Shadowing

By default, you can end shadowing using the CTRL and * (asterisk) hotkey combination.

➤ **To end the shadowing session**

Press the CTRL key and the * (asterisk) key of your keyboard's numeric keypad.

Configuring a Different Hotkey to End Shadowing

If you cannot use the default hotkey combination from the ICA Client device you are using, or you prefer to use an alternative, you can configure your own combination. You do this using the **ctxshadow** command.

The hotkey you configure applies only to the current shadowed session and therefore needs to be set up each time you shadow a session.

➤ **To configure a different hotkey to end shadowing**

1. Log on to an ICA session.
2. At the command prompt, type **ctxqsession**. This command displays the current sessions at this server.
3. From the results of the ctxqsession command, identify the session name or session id of the user's session that you want to shadow.
4. At the command prompt, type:

```
ctxshadow {id|name} [-h [[a][c][s]+]x]
```

where [a][c][s] and x is the hotkey combination you want to use to end shadowing—choose this combination from:

[a][c][s] a = ALT; c = CTRL; s = SHIFT

(Note: you can use any combination of a, c and s, including all or none.)

x Choose from the alphanumeric characters:
a to z and 0 to 9.

Example

To begin shadowing and to specify a hotkey combination of ALT and q to stop shadowing the session, type:

```
ctxshadow tcp#5 -h a+q
```

Sending Messages to Users

You can send a message to users using the **ctxmsg** command.

Tip To inform users that the server is about to shut down, use the message option with the **ctxshutdown** command. See “Stopping MetaFrame” in Chapter 1 for further information.

➤ **To send a message**

- 1. Log on to the MetaFrame server as a Citrix server administrator.
- 2. At the command prompt, type **ctxquser**. This command displays the current sessions at this server.
- 3. From the results of ctxquser, identify a session name or session id to identify the user and session you want to send a message to.
- 4. At the command prompt:

To	Use the command
Send a message by specifying a session id.	<code>ctxmsg id message</code>
Send a message by specifying the session name.	<code>ctxmsg name message</code>
Send a message that includes a timeout period, in seconds. The message is displayed on the user’s screen until it times out or the user dismisses it.	<code>ctxmsg {id name} message timeout</code>
Send a message that will suspend your terminal window, until the message times out or is dismissed by the user. In other words, you get your command prompt back only when the user responds or the message times out.	<code>ctxmsg -w {id name} message timeout</code>

Note If a message includes spaces or any other characters that have a special meaning in your UNIX shell, enclose all the text in double quotes.

Examples

```
ctxmsg tcp#5 "Contact System Admin"
ctxmsg 11 Hello
ctxmsg 5 "Fancy lunch?" 30
```

Printing

This section describes the information your ICA Clients need to know when they want to print. It explains how users can list available client printers and print files from the command line or from applications.

In the UNIX environment, the application performs the print rendering. The print driver is specified inside the application or, in the case of a desktop utility, raw, unformatted text is generated.

Displaying Client Printers or Printer Ports

When an ICA Client connects to a MetaFrame server, client printers are mapped and are available from the desktop command line and from applications running in the session.

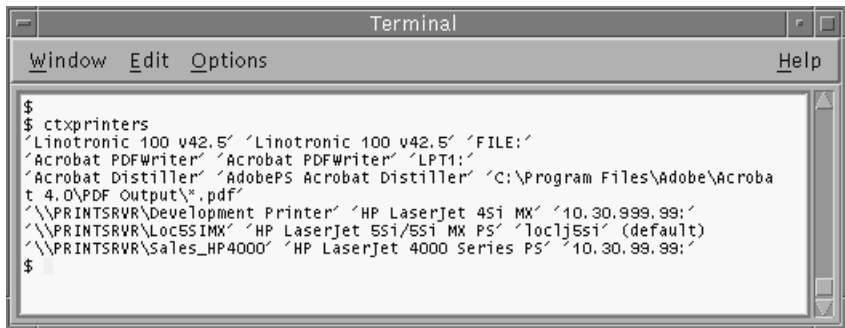
From an ICA Client session, users can list the mapped client printers, or available printer ports, using the **ctxprinters** command.

➤ **To display mapped client printers**

At the command prompt, type:

```
ctxprinters
```

A list of printers configured on the client and mapped for use from the ICA session is displayed.



```
Terminal
window Edit Options Help
$
$ ctxprinters
'Linotronic 100 v42.5' 'Linotronic 100 v42.5' 'FILE:'
'Acrobat PDFWriter' 'Acrobat PDFWriter' 'LPT1:'
'Acrobat Distiller' 'AdobePS Acrobat Distiller' 'C:\Program Files\Adobe\Acroba
t 4.0\PDF Output\*.pdf'
'\\PRINTSRVR\Development Printer' 'HP LaserJet 45i MX' '10.30.99.99:'
'\\PRINTSRVR\Loc55IMX' 'HP LaserJet 55i/55i MX PS' 'loc1j55i' (default)
'\\PRINTSRVR\Sales_HP4000' 'HP LaserJet 4000 Series PS' '10.30.99.99:'
$
```

(default) is displayed after the printer that is the default. The following information is shown for each printer:

- Printer name or printer port (for example, lpt1). This can be used in the **ctxlpr -P** command to specify a printer other than the default.
- Printer driver name. This is for information only.
- Printer connection description. This is for information only.

Printing From the Command Line

Within an ICA session, users can print a file from the command line by using **ctxlpr**, instead of **lpr** or **lp**. If no files are specified, the ctxlpr command takes its input from standard input (stdin).

➤ To print a file from an ICA Client session

1. At the command prompt, type **ctxprinters**.
2. From the results of ctxprinters, identify the printer or printer port, that you want to use. To print to a printer other than the default, note the printer name—the printer name is the first item in the ctxprinters listing.
3. At the command prompt:

To	Use the command
Print the file named <i>filename</i> to the default printer.	<code>ctxlpr <i>filename</i></code>
Print a series of files to the default printer. Each file is treated as a separate print job.	<code>ctxlpr <i>filename filename</i></code>
Print a file to a printer (or printer port) other than the default. This is the printer name or printer port shown in the first column of the output from ctxprinters.	<code>ctxlpr -P [Prntername Printerport] <i>filename</i></code>
Print a file in the background.	<code>ctxlpr -b <i>filename</i></code>
Print a file only if the printer is not in use. Use this option to stop an application waiting while other printer jobs are handled. If the printer is in use, an error message is displayed.	<code>ctxlpr -n <i>filename</i></code>

Examples

To send the file **mydoc.ps** to the printer **\\PRINTSRVR\Sales_HP4000**, use the following command:

```
ctxlpr -P '\\PRINTSRVR\Sales_HP4000' mydoc.ps
```

If you are using a client that uses direct printer port mapping:

```
ctxlpr -P lpt2 mydoc.ps
```

Printing From Applications

The exact configuration of how to set up printing from applications depends on the behavior and user interface of the UNIX application.

If the user interface for an application allows you to specify the actual printer command to use when printing, you can configure client printing by replacing the **lpr** or **lp** command with the **ctxlpr** command.

When a user connects to the server and prints from the application in a session, the server redirects the output to the mapped client printer.

Often, in this type of application, you can also specify the command line modifiers on a different line. You can use the same switches for **ctxlpr** as when printing from the command line. For example, **-P** with a printer name (or printer port) to print to a printer other than the default; **-b** for background printing, and so on.

Tip If the user interface of an application does not allow you to specify the actual printer command to use when printing, find out whether the application (or window manager) uses a configuration file where you can replace the **lpr** command functionality with **ctxlpr**.

Troubleshooting Printing

Because UNIX applications generally produce only UNIX ASCII text or PostScript output, PCL (Printer Control Language) printers are not suitable. Therefore, ensure your client printers support PostScript. If you do not have a PostScript printer, install a utility such as Ghostscript to convert PostScript files to a different output format, such as PCL.

If text does not print out correctly, this may be due to carriage return / line feed differences between UNIX and DOS text files. To print a UNIX text file to a Windows printer, use a utility such as **unix2dos**. For example, to print out a UNIX text file called “printfile” type:

Solaris `unix2dos printfile | ctxlpr`

HP-UX `ux2dos printfile | ctxlpr`

Alternatively, use Perl instead. For example, type:

```
perl -pe 's/\n$/\r\n/' printfile | ctxlpr
```

Or, create a script file called “unix2dos” that includes the following:

```
#!/bin/sh
perl -pe 's/\n$/\r\n/' "$@"
```

Make the script file executable using **chmod a+rx unix2dos**. You can now use the script file just like the **unix2dos** utility.

CHAPTER 5

Configuring a MetaFrame Server



Overview

This chapter describes how to configure the MetaFrame server to provide the required resource access and session behavior for the ICA Client users of your network. Topics in this chapter include:

- Configuring the server
- Understanding and configuring the ICA Browser Service
- Configuring ICA Gateways
- Customizing scripts

Configuring the Server

You can configure your server in different ways to control access to services for users connecting to the server. The combination of settings you use depends on how you intend to use your Citrix servers. From the server, you can configure settings that include:

- The number of ICA sessions you want to allow at this server.
- What happens to a session if the connection is broken or times out.
- Whether to allow local printer mapping. If you enable printer mapping from the server, ICA Client users can configure and use their local printer to print from applications that are actually running on the server.
- Whether to allow local clipboard mapping. If you enable clipboard mapping from the server, ICA Client users can copy and paste text between applications running on the client device and the remote applications running on the server.
- Whether to allow shadowing.
- The maximum permitted session duration, and how long to leave idle or disconnected sessions before timing out.

Note User access to MetaFrame commands and sessions is controlled by the MetaFrame **ctxsecurity** function. See “Configuring MetaFrame Security” in Chapter 6 for information.

Controlling Logon Settings

When ICA Client users connect to a MetaFrame server, the users need to supply a username and password (unless they are accessing an application published for anonymous use). Users can either type this information into the dialog box that appears when they connect to the server or published application, or configure the client so that their username and password are saved as part of the properties for a particular connection.

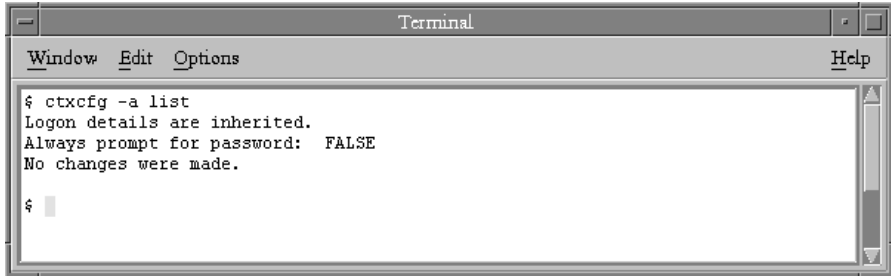
You can also use the **ctxcfg** tool on the server to configure settings that give you and ICA Client users flexibility and security when logging on.

➤ **To display the current logon settings**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxcfg -a list
```

This displays the current logon settings.



```

Terminal
Window Edit Options Help
$ ctxcfg -a list
Logon details are inherited.
Always prompt for password: FALSE
No changes were made.
$

```

➤ **To change the logon settings**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Configure the server so that if logon details are set on the client, they are used.	<code>ctxcfg -a INHERIT</code>
Configure the server so that a user logging on is always prompted for a password, regardless of any password set in the server or the client.	<code>ctxcfg -a prompt=TRUE</code>
Configure the server so that a user logging on is not prompted for a password.	<code>ctxcfg -a prompt=FALSE</code>
Set a default user name for all users who log on to the server. For example, you can use this to set up a guest user account.	<code>ctxcfg -a user=<i>name</i></code>
Set a password for all users who log on to the server. Type pass as a keyword; <code>ctxcfg</code> then displays a prompt where you can type in the password. Note that if you have not set up a user name, this setting is ignored.	<code>ctxcfg -a pass</code>
Erase any user name and password details that have been set (using the user and pass options) and configure the server to use logon details set on the client.	<code>ctxcfg -a ERASE</code>

Note The list argument never displays passwords.

Setting the Number of Permitted ICA Connections

You can specify a maximum number of concurrent ICA connections that a particular server will support.

- **To check the current number of permitted connections**
 1. Log on to the MetaFrame server as a Citrix server administrator.
 2. At the command prompt, type:

```
ctxcfg -l list
```

This command displays the number of logons permitted, or displays UNLIMITED if no limit is set.

- **To change the number of permitted connections**
 1. Log on to the MetaFrame server as a Citrix server administrator.
 2. At the command prompt:

To set	Use the command
A maximum, where <i>n</i> is the number of concurrent connections you want to allow	<code>ctxcfg -l max=<i>n</i></code>
No limit to the number of concurrent sessions you want to allow	<code>ctxcfg -l max=UNLIMITED</code>

Note The number of ICA connections that a server can support is also affected by Citrix Licensing—see Chapter 2 for more information.

Controlling Behavior for Disconnected or Broken Connections

A broken connection occurs when the communication link between the ICA Client and the MetaFrame server is interrupted—for example, as the result of a network failure.

Use the **ctxcfg** tool with the **-c** option to control the behavior for broken or timed-out connections, and to specify reconnection options.

- **To display the current configuration for broken and timed-out connections**
 1. Log on to the MetaFrame server as a Citrix server administrator.
 2. At the command prompt, type:

```
ctxcfg -c list
```

```

Terminal
Window Edit Options Help
$ ctxcfg -c list
On broken connection: DISCONNECT
Allow reconnection from: ORIGINAL
No changes were made.
$

```

➤ **To configure the settings for disconnected and broken connections**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To configure the server so that	Use the command
Broken connections are immediately reset	<code>ctxcfg -c broken=reset</code>
Broken connections are disconnected	<code>ctxcfg -c broken=disconnect</code>
A user can connect to a disconnected session from any client device	<code>ctxcfg -c reconnect=any</code>
A user can only connect to a disconnected session from the original terminal	<code>ctxcfg -c reconnect=original</code>

You can configure the system so that disconnected sessions are reset automatically after a timeout interval or continue until a user (or a Citrix server administrator) resets the session. See “Controlling Timeout Behavior” later in this chapter for details of how to set a timeout interval for disconnected sessions.

Enabling or Disabling Printing for Users

Client printer mapping allows ICA Client users to use printers that are available on the client device from applications running on a MetaFrame server.

Use the **ctxcfg** tool with the **-p** switch to enable or disable client printer mapping.

➤ **To check if client printing is currently enabled or disabled**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:
`ctxcfg -p list`

➤ **To enable or disable client printing**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Enable client printing	<code>ctxcfg -p enable</code>
Disable client printing	<code>ctxcfg -p disable</code>

Enabling or Disabling Clipboard Mapping

Users can copy text and graphics between server-based applications and applications running locally on the client device. Even if an application is running on the server, the clipboard behaves as if it is on the client device.

Use the **ctxcfg** tool with the **-C** switch to enable or disable client clipboard mapping.

➤ **To check if the client clipboard is currently enabled or disabled**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxcfg -C list
```



The screenshot shows a Windows Terminal window titled "Terminal". The menu bar includes "Window", "Edit", "Options", and "Help". The command prompt shows the command `$ ctxcfg -C list` being entered. The output of the command is displayed as follows:

```
$ ctxcfg -C list
Client clipboard enabled
No changes were made.
$
```

➤ **To enable or disable the client clipboard**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Enable client clipboard mapping	<code>ctxcfg -C enable</code>
Disable client clipboard mapping	<code>ctxcfg -C disable</code>

Providing Additional Graphics Clipboard Support

MetaFrame provides users with the **ctxgrab** tool that lets them grab windows or screen areas and copy them from an application in an ICA Client window to an application running on the local client device.

By default, **ctxgrab** is available to users connecting to published applications through the **ctxwm** window manager as follows:

- In a seamless window, right click the  button in the top, left hand corner of the screen to display a menu and choose the **Screen Grab** option.
- In a “full screen” window, right click to display the **ctxwm** menu and choose the **Screen Grab** option.

Users connecting to a server desktop can run the tool by typing **ctxgrab** at the command prompt.

If you have users who require more extensive graphics clipboard support, you can deploy the **ctxcapture** tool. With **ctxcapture** users can:

- Grab dialogs 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 xv. xv is a Shareware utility that is available for download from the Internet.

Providing Users with ctxcapture

You do not have to do anything to make **ctxcapture** available to users connecting to a server desktop—it is available from the command prompt by typing **ctxcapture**.

You can provide users who are connecting to published applications with **ctxcapture**, by making it available from the **ctxwm** window manager. To do this, you edit the Ctxcapture resource file to make ctxcapture, rather than ctxgrab, available.

➤ To make ctxcapture available to users of published applications

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Open the **Ctxcapture** resource file. This is located in the:

HP-UX

Solaris

AIX

/opt/CTXSmf/data/C directory

/usr/lpp/CTXSmf/data/C directory

3. Locate all the lines beginning with **ctxappgrab** and comment these out.
4. Locate all the lines beginning with **Ctxcapture** and copy these to the clipboard.

5. Paste the lines you copied into a new section of the file.
6. Edit these lines to change **Ctxcapture** to **ctxappgrab**.

Tip If you want to make ctxcapture available to a particular user only, copy the Ctxcapture resource file to this user's home directory and change it to make ctxcapture available.

Enabling or Disabling Shadowing

Session shadowing allows you to monitor the display of another active session. Shadowing allows you to see what users are doing and interact with their sessions, using the keyboard and mouse. You can shadow active sessions on the same server.

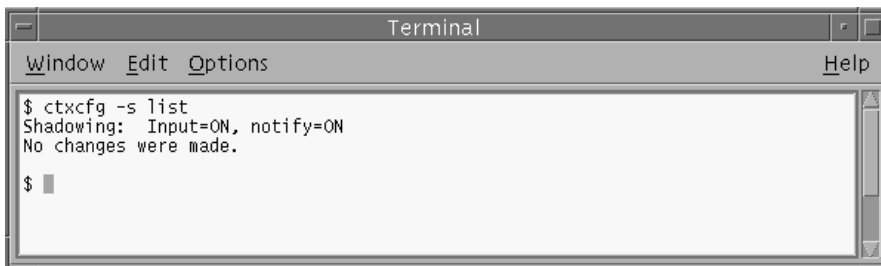
Use the **ctxcfg** tool with the **-s** switch to configure shadowing.

Note By default, any user can shadow any other session. To change this, use MetaFrame security—see “Configuring MetaFrame Security” in Chapter 6 for further information.

➤ **To display the current shadowing settings for the server**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:
`ctxcfg -s list`

The shadowing configuration for the current server is displayed, for example:



```
Terminal
Window Edit Options Help
$ ctxcfg -s list
Shadowing: Input=ON, notify=ON
No changes were made.
$
```

➤ **To change the shadowing settings for the server**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To enable shadowing	Use the command
So that sessions on the server can be shadowed. By default, input is set to on and notify to on .	<code>ctxcfg -s enable</code>
To change the input and notify options	Use the command
So that the shadower can input keyboard and mouse actions to the shadowed session.	<code>ctxcfg -s input=on</code>
So that the shadower cannot input keyboard and mouse actions to the shadowed session.	<code>ctxcfg -s input=off</code>
So that the shadowed user gets a notification message requesting confirmation that the shadowing can occur.	<code>ctxcfg -s notify=on</code>
So that the shadowed user does not get a notification message.	<code>ctxcfg -s notify=off</code>
To disable shadowing	Use the command
So that sessions on the server cannot be shadowed.	<code>ctxcfg -s disable</code>

Important Disabling shadowing notification means that users might be shadowed by another user, but be unaware that they are being shadowed. Some countries require by law that users be notified before shadowing occurs.

Example

You may want to set up shadowing to help you solve technical support issues. The system administrator can show the user how to complete a task by shadowing the user's session. To allow shadowing with notification and to allow the shadower to control the mouse and keyboard, use the command:

```
ctxcfg -s enable,input=on,notify=on
```

See “Shadowing a User's Session” in Chapter 4 for information about shadowing sessions.

Controlling Timeout Behavior

You can use the **ctxcfg** tool with the **-t** switch to specify timeout intervals for connected, disconnected, and idle ICA sessions.

These settings specify timeout intervals in minutes. The timeouts are:

Connection	The maximum connection duration. If a connection duration is specified, the session is disconnected or terminated when the specified duration elapses. If NONE is specified, the connection timer is disabled.
Disconnection	The maximum duration that a disconnected session is retained. If a disconnection duration is specified, sessions in the disconnected state are terminated when the specified duration elapses. If NONE is specified, the disconnection timer is disabled.
Idle	The maximum idle time (time without user activity) allowed before the session is disconnected or reset. If an idle duration is specified, the session is disconnected or reset when the specified interval elapses without any activity on the connection. If NONE is specified, the idle timer is disabled. To specify whether sessions are disconnected or reset, see “Controlling Behavior for Disconnected or Broken Connections” in this chapter. To specify an idle timeout period for anonymous users, see “Configuring Anonymous Users” in Chapter 6.

➤ To display the current timeout intervals

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxcfg -t list
```

The current timeout value for each setting is displayed. If a timeout interval is configured, the value is shown in minutes. If no timeout interval is configured, the keyword NONE shows that sessions will not be timed out.

```

Terminal
Window Edit Options Help
$ ctxcfg -t list
Max connection time: NONE
Max disconnection time: NONE
Max idle time: NONE
No changes were made.
$

```

➤ **To change the timeout intervals**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To set	Use the command
A connection timeout (in minutes). All connections are terminated after this period.	<code>ctxcfg -t connect=<i>num</i></code>
No connection timeout. All sessions continue until the user disconnects or logs off.	<code>ctxcfg -t connect=NONE</code>
A disconnection timeout (in minutes). Disconnected sessions are reset after this period.	<code>ctxcfg -t disconnect=<i>num</i></code>
No disconnection timeout. Disconnected sessions remain until reset by a user or a Citrix server administrator.	<code>ctxcfg -t disconnect=NONE</code>
An idle timeout (in minutes). If no user activity is detected during this time, the connection is terminated.	<code>ctxcfg -t idle=<i>num</i></code>
No idle timeout. All sessions continue until the user disconnects or logs off.	<code>ctxcfg -t idle=NONE</code>

Example

If you expect users to dial into the server, you may want to set the disconnect timeout to a suitable setting in case of a broken connection. Users can reconnect to their sessions during the timeout interval.

To set the disconnection timeout to 15 minutes:

```
ctxcfg -t disconnect=15
```

Generating and Using Server Configuration Details

You can generate a list of the current **ctxcfg** settings for a particular server. If you send the output to a file, you can use the file in a shell script to replicate identical configuration settings on other servers. You can also use a file as a temporary backup of the current configuration, allowing you to experiment with other settings, but easily restore your original settings, if desired.

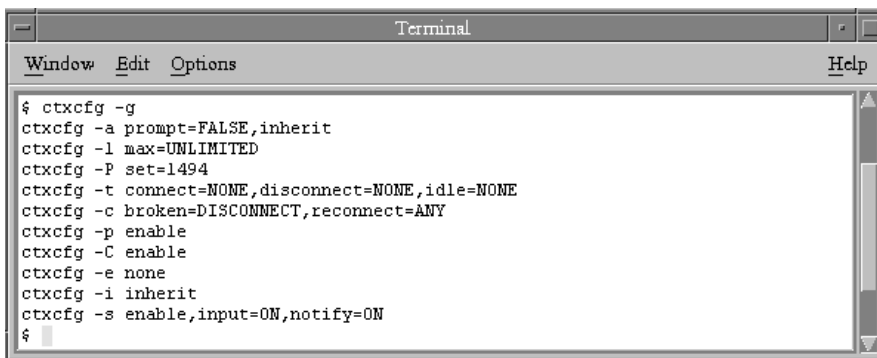
Displaying a List of the Current Configuration

➤ **To display a list of the current ctxcfg settings**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxcfg -g
```

This generates a list of the current settings.



```
Terminal
Window Edit Options Help
$ ctxcfg -g
ctxcfg -a prompt=FALSE,inherit
ctxcfg -l max=UNLIMITED
ctxcfg -P set=1494
ctxcfg -t connect=NONE,disconnect=NONE,idle=NONE
ctxcfg -c broken=DISCONNECT,reconnect=ANY
ctxcfg -p enable
ctxcfg -C enable
ctxcfg -e none
ctxcfg -i inherit
ctxcfg -s enable,input=ON,notify=ON
$
```

Creating a Shell Script of the Current Configuration

When you use the **-g** option with the **ctxcfg** command, the generated list contains the commands and settings for the current configuration using the **ctxcfg** command line syntax. You can redirect the output of this command to a file and use the file as a shell script to restore (or set) this configuration.

Propagating Server Configuration to Multiple Servers

You can use the output from **ctxcfg -g** if you want to configure a number of servers in the same way.

- **To propagate the same configuration from one server to another**
 1. Complete the configuration of the first server. Generate a list of the server configuration using the **ctxcfg** command and the **-g** option, piping the output of the command to a file. Note that **ctxcfg -g** does not generate the password, so you need to enter this manually.
 2. Log on to the next server as a Citrix server administrator and run the file as a shell script.

Tip You can use the **rsh** (remote shell) command to propagate the shell script on a remote server.

Screensaver Setting Recommendations

ICA connections running graphical screensavers can consume considerable server resources. Therefore, we recommend that screensavers are turned off.

- **To turn screensavers off**

Run the **xset** command with the **s** option and **off** parameter:

```
xset s off
```

Although it is best to turn screensavers off, if you prefer not to (for security reasons perhaps) you can use the X server “prefer blanking” screensaver option. This causes the screen to go blank, rather than display a pattern, when the screensaver is activated.

- **To set the screensaver blanking option**

Run the **xset** command with the **s** option and **blank** parameter:

```
xset s blank
```

To display the current settings, run the **xset** command with the **q** option.

Note To ensure published applications are run in sessions with the appropriate screensaver settings, we recommend you publish a script file that runs the **xset s off** command, and then the application.

MetaFrame for UNIX and the ICA Browser Service

The ICA Browser maintains data on Citrix servers and published applications. The ICA Browser consists of a master browser, member browsers, and client systems. The ICA Browser uses directed packets to communicate with other ICA Browser services running on Citrix servers.

Citrix ICA Clients query the ICA Browser service to obtain a list of Citrix servers and published applications. The Citrix ICA Client queries the ICA Browser service for the network address of servers and published applications when a session is launched. Citrix servers use the ICA Browser service to pool licenses and share administrative and performance information.

Controlling the Master Browser

Every Citrix server runs the ICA Browser service. One Citrix server is elected the *master browser*; all other Citrix servers on the network are *member browsers*. The master browser is an ICA Browser acting as a central information store. The master browser keeps track of the following information:

- The available Citrix servers
- The available published applications
- Pooled licenses (user counts)
- Performance and load information for Citrix servers

The master browser for each network is chosen by a *master browser election*. If the current master browser on a network is not responding, a new master browser election is held automatically. This provides high reliability for the ICA Browser service.

In general use, the ICA Browser service is invisible to you and does not affect the continued operation of MetaFrame for UNIX Operating Systems. However, you may want to manipulate the possibility of a particular machine becoming the master browser, because:

- Servers running different versions of MetaFrame provide different versions of the ICA Browser service when they are master browser. This can affect the features available to users. In particular, if you have MetaFrame 1.8 servers in your network, you may want to ensure that a MetaFrame for UNIX 1.1 server in the network does not become the master browser.
- The master browser service consumes more resources, so you may want to make sure that machines that receive many connections are less likely to become the master browser. If necessary, you can configure the settings of one or more dedicated machines in the network so that one of these machines is more likely to become the master browser.

Locating the Current Master ICA Browser

You can use the **ctxmaster** command to locate the Citrix server acting as the master browser.

➤ **To locate the master browser**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxmaster
```

The master ICA Browser address and server name are displayed.

If no master browser is running, for example during a browser election, the error message “Could not find any servers which match the requirements” is displayed.

Manipulating Master Browser Elections

The ICA Browsers on a network subnet elect a master browser under any of the following conditions:

- The current master browser does not respond to another ICA Browser
- The current master browser does not respond to an ICA Client
- A Citrix server is started
- Two master browsers are detected on the same network subnet

A combination of factors affect the outcome of the election. The two main factors are:

- The version of the Citrix server—for example, MetaFrame 1.8, MetaFrame for UNIX 1.1, *WINFRAME* 1.7 (note that MetaFrame for UNIX Release 1.0 and Release 1.1 are treated the same)
- The master browser preference setting for each server

You set this preference using the **ctxbrcfg** tool for MetaFrame for UNIX Operating Systems servers or, for other Citrix servers, Citrix Server Administration. Citrix servers can be set to:

- | | |
|----------------|---|
| neutral | The default value of “no preference” behavior in elections. |
| always | Always attempt to become master browser. |
| never | Never attempt to become master browser. |

Other factors, such as the length of time a server has been running and whether the Citrix server is also a Windows NT domain controller (not applicable to MetaFrame for UNIX Operating Systems), can also affect an election result.

Introducing a New Server

Introducing any type of Citrix server into your network forces an election.

- The default master browser preference setting for a MetaFrame for UNIX Operating Systems server is **neutral**; i.e. unbiased. If you add a server with this default setting to a network that includes MetaFrame 1.8 servers that are also configured as unbiased in elections, the MetaFrame for UNIX server will not become the master browser. Under these circumstances, a MetaFrame 1.8 for Windows server automatically has preference to become the master browser.
- If you add a MetaFrame for UNIX server to a network that includes only other MetaFrame for UNIX servers, it may become master browser if all the other servers are set to **neutral**.

Important It is the combination of settings for all servers on the network that decides the results of an election. A MetaFrame for UNIX server could win a master browser election with a MetaFrame 1.8 server, if the master browser preference on the MetaFrame 1.8 server is set to **never**.

Biasing the Results of Elections

➤ **If you do not mind which server becomes master browser**

Leave the master browser preference setting on each server to the default setting of **neutral** or **no preference**, as appropriate for the type of server.

➤ **If you want a particular server to be the master browser**

1. Configure the master browser preference on the server you want to become master browser to be **always**. Use the **ctxbrcfg** command for MetaFrame for UNIX. See “Configuring the ICA Browser” in this section for more detailed instructions. For other Windows-based Citrix servers, use Citrix Server Administration.
2. Leave the master browser preference on the other servers to be **neutral** or **no preference**, as appropriate for the type of server. Do not set the other servers to be **never**—reserve this setting for a particular server that should never become master browser.
3. The changes you make using **ctxbrcfg** will cause a master browser election to occur. Wait a few moments for the election to take place and then check the master browser status using the **ctxmaster** command.

➤ **If you want to stop a particular server from becoming the master browser**

1. Configure the master browser preference on the server that you do not want as the master browser to be **never**, using the **ctxbrcfg** command.

Important Do not set the master browser preference on all servers in a network to be **never** because unpredictable election results will occur.

2. Leave the master browser preference on the other servers that can become the master browser to be **neutral** or **no preference** as appropriate for the type of server. Any changes you make using the ctxbrcfg command will cause a master browser election to occur.

Configuring the ICA Browser

The ICA Browser maintains data about Citrix servers and published applications. You can display and change the ICA Browser settings on a server using the **ctxbrcfg** tool. Any changes you make using ctxbrcfg will cause a master browser election to take place.

➤ **To control how the ICA Browser behaves during browser elections**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Configure the server so that it always attempts to become the master browser in an election, subject to the presence and actions of other browsers.	ctxbrcfg -m always
Configure the server so that it refrains from participating in an election. Note that the server can still become the master browser under some circumstances.	ctxbrcfg -m never
Configure the default behavior of “no preference”.	ctxbrcfg -m neutral

The refresh period controls how often the ICA Browser on this server updates the master ICA Browser. The ICA Browser updates the master ICA Browser after the specified amount of time elapses. A short refresh period makes the master ICA Browser data more accurate, but increases CPU and network load.

➤ **To view or change the refresh interval for the ICA Browser service**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Display the current refresh interval.	<code>ctxbrcfg -r list</code>
Set a period (in minutes) at which the local ICA Browser service will update the master browser.	<code>ctxbrcfg -r set=<i>num</i></code>

Note The default settings work for most installations. Change them only when you understand the implication of each setting.

Starting and Stopping the ICA Browser

You can start and stop the ICA Browser process on a server, without having to stop and start all the MetaFrame processes, using the **ctxsrv** tool.

If you stop the ICA Browser on a server, users will be unable to connect to published applications on this server, although they will still be able to connect to the server desktop. If you stop the ICA Browser process on the master browser, a master browser election will occur among the other Citrix servers on the network.

➤ **To start or stop the ICA Browser using ctxsrv**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:
`ctxsrv {start|stop} browser`

Configuring ICA Gateways

For Citrix servers or ICA Clients to contact Citrix servers on a different network, an *ICA Gateway* must be configured.

The master ICA Browser maintains the browse list and periodically receives updates from other browsers (the Citrix servers) on the same network. The exchange of information between the master ICA Browser and the other browsers takes place over the local subnet. To communicate and exchange information with other networks, you must establish an ICA Gateway on the participating networks.

If you have more than one network subnet—for example, if you use a router or a WAN to connect two networks—you need to set up an ICA Gateway to allow the master browsers on each network to share information about available Citrix servers and published applications.

An ICA Gateway consists of at least two Citrix servers. The *local server* is responsible for contacting the other network and setting up a link between the master browsers on each network. The *remote server* is a Citrix server on the other network that communicates with the local server to establish the ICA Gateway.

➤ To display the ICA Gateways configured on a server

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:
`ctxbrcfg -g list`

➤ To add or remove an ICA Gateway

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt:

To	Use the command
Add a gateway host name or IP address to the list.	<code>ctxbrcfg -g add=<i>gateway</i></code>
Remove a gateway host name or IP address from the list.	<code>ctxbrcfg -g remove=<i>gateway</i></code>

Note You can also use the **ctxqserver** command to display information about the ICA Gateways known to each server on the network—see the “MetaFrame Command Reference” in Appendix A for details.

About Customizable Scripts and Files

The MetaFrame installation includes script files that you can customize. The scripts are in the:

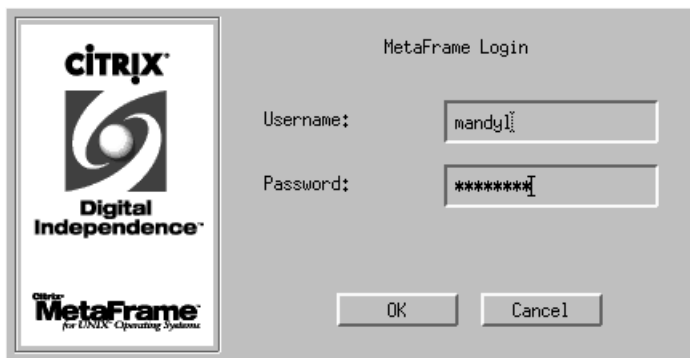
- HP-UX** **Solaris** /opt/CTXSmf/lib directory
- AIX** /usr/lpp/CTXSmf/lib directory

The script **ctxXtw.sh** runs when the X server starts, and includes X server configuration settings such as the font path and the X security policy. By default, the MetaFrame X server does not use a security policy (for the X security extension). It is disabled by the option **-sp /dev/null** in ctxXtw.sh. If you want to use a security policy, write the security policy (see the Xserver man page for details about how to do this) and then change this option in ctxXtw.sh to point to the policy file.

The script **ctxsession.sh** runs after a user logs on. You can use this script file to customize the local environment for user sessions, such as defining the default window manager, configuring scripts, or setting environment variables for users.

Customizing the Login Screen

You can change the appearance of the MetaFrame Login screen by substituting the Citrix logo frame for a graphic of your choice. The graphic you choose must be in .xpm (X pixmap) format.



The graphic used in the MetaFrame Login screen is located in:

- HP-UX** **Solaris** /opt/CTXSmf/data/C/logo.xpm
- AIX** /usr/lpp/CTXmf/data/C/logo.xpm

The image that is displayed is limited to 120 x 200 pixels and 256 colors. If you use a larger graphic, only the 120 x 200 pixels in the center of the graphic are displayed. If you use a smaller graphic, the image displayed is centered in the frame.

➤ **To display a different logo in the Login screen**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Rename the current Citrix **logo.xpm** file—for example: **old_logo.xpm**
3. Rename your new graphic to **logo.xpm** and move this to the appropriate **.../CTXSmf/data/C/** directory (see previous).

The new graphic is displayed in the MetaFrame Login screen.

Changing the Window Manager

Using **ctxsession.sh** you can configure the system to load a window manager, other than CDE. You can do this for every user who logs on to the MetaFrame server, or for a particular user.

Note You can change only the window manager that MetaFrame loads for connections to server desktops. MetaFrame loads the **ctxwm** window manager for all connections to published applications.

Changing the Window Manager for Every User

Use the following procedure to load a new window manager for every user who logs on to the MetaFrame server.

Note The window manager is not loaded for any initial programs that a user has set on the client. To do this, use the procedure described in “Changing the Window Manager for a Particular User” later in this section.

➤ **To use a different window manager for every user**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Install the new window manager.
3. Open the **ctxsession.sh** script and locate the following line:
`CDE_WM=/usr/dt/bin/dtsession`
4. Change this line to:
`CDE_WM=/path/window_manager`
where *path* is the location of the new window manager and *window_manager* is the name of the new window manager.

Changing the Window Manager for a Particular User

Use the following procedure to load a new window manager for a particular user each time the user logs on. The new window manager is also loaded for any initial programs that the user has set on the client.

➤ To use a different window manager for a particular user

1. Log on to the MetaFrame server as a Citrix server administrator.
2. Install the new window manager.
3. Open the **ctxsession.sh** script and locate the following lines:

```
#if [ -f $HOME/.ctx.session.sh ] ; then
#   $HOME/.ctx.session.sh
#fi
```
4. Remove the # character from the start of each line, so that these lines are no longer commented out.
5. In the user's home directory, create a file called **.ctx.session.sh**
6. In the **.ctx.session.sh** file, add the new window manager's bin directory to the path:

```
PATH=${PATH}:/path
```

 where *path* is the location of the new window manager's bin directory.
7. Add lines to load the new window manager and suppress the message that tells the user that the window manager is being loaded:

```
DESKTOP_WM="/path"
DESKTOP_MESSAGE=""
```

 where *path* is the location of the new window manager's start file.
8. Add lines to load the new window manager when an initial program is launched, and suppress the message that tells the user that the window manager is being loaded:

```
INITIAL_APPS_WM="/path"
INITIAL_APPS_MESSAGE=""
```

 where *path* is the location of the new window manager's start file.

Example

The following example shows the lines required in the user's **.ctx.session.sh** file to start the **kde** window manager:

```
PATH=${PATH}:/usr/local/kde/bin
DESKTOP_WM="/usr/local/kde/bin/startkde"
DESKTOP_MESSAGE=""
INITIAL_APPS_WM="/usr/local/kde/bin/startkde"
INITIAL_APPS_MESSAGE=""
```

The result is that every time a user logs on, the system checks for the **.ctx.session.sh** file in the user's home directory. If it finds this file, the system runs it and the new window manager is loaded. If the file is not found, or the user connects to a published application, CDE is loaded.

Advanced Topics



Overview

This chapter discusses advanced MetaFrame system administration topics. Topics discussed include:

- Load balancing published applications
- Configuring anonymous user settings
- Configuring MetaFrame security
- Configuring MetaFrame for non-standard shells
- Using ICA with network firewalls
- Configuring the TCP/IP port number
- Configuring the operating system for a large number of connections

Load Balancing Published Applications

Load balancing determines which servers are least busy and can best run an application. When a published application or desktop is launched from a Citrix ICA Client, load balancing selects which server will run the application or desktop session based on server load.

Load balancing also offers increased availability. By configuring a pool of servers capable of running your users' applications, you can easily bring servers off-line for maintenance or add more servers for increased performance without affecting application availability.

To use load balancing, you need to purchase and install Citrix Load Balancing Services. See "Citrix Licensing" in Chapter 2 for more information.

Load Balancing Using a Group of Servers

By default, MetaFrame for UNIX automatically balances the load among a group of servers, by sending connections to each server in the group, in turn.

➤ To load balance a group of servers

1. Install a Citrix Load Balancing Services license on each server.
2. Publish the application on each server. You can publish the application on one server and then create and run a script file to publish the same information on each server—see "Configuring Published Applications" in Chapter 3 for more information.
3. Allow users to connect to the published application. Connections are distributed among all the servers on which the application is published.

Note Load balancing works by identifying the names of published applications. You must make sure that applications you want to balance over a number of servers are given the same name in **ctxappcfg** on each server.

Different types of servers, for example, with different processor speeds or available memory, will be able to accept a different number of connections before becoming busy.

If you find that some servers become more busy than others with evenly distributed connections, you can further control the load by using the **ctxcfg** tool to tune the number of connections on each server.

➤ **To tune the number of connections on each server**

1. Install a Citrix Load Balancing Services license on each server.
2. Publish the application on each server. You can publish the application on one server and then create and run a script file to publish the same information on each server.
3. Allow users to connect to the published application. Connections are distributed among all the servers on which the application is published.
4. If a particular user experiences problems running a session, use the **ctxqsession** command to identify the server to which the user is connected.
5. Count the active number of sessions on the server that is causing the problem, and then limit the maximum number of users who can log on to the server, using **ctxcfg** and the **-l** option.
6. Limit the maximum number of users who can log on, on the other servers, in a similar way. For best results, set a value on each server.

Troubleshooting Load Balancing

If users frequently disconnect and reconnect to sessions on load balanced Citrix servers, rather than logging off, the load may not be distributed evenly among the servers.

Also, if users are using clients that support session sharing and they start multiple applications, the clients will attempt to start subsequent applications within the existing session, rather than create a new session. This may lead to uneven application load distribution among the servers.

Configuring Anonymous Users

During MetaFrame installation, you can create a special user group called **ctxanon** on the Citrix server, together with 15 local anonymous user accounts. These accounts allow 15 users guest access to applications that you publish for anonymous use.

Anonymous user accounts do not usually require further maintenance; however their properties can be displayed and modified using the **ctxanoncfg** command.

Note You must be root to display and update anonymous user settings.

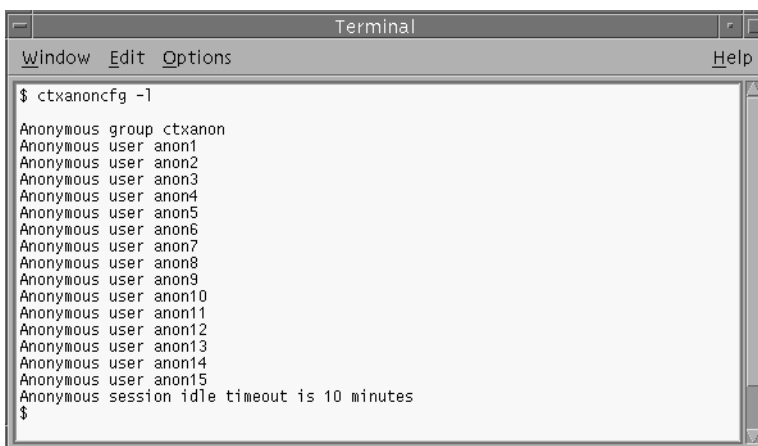
Displaying Anonymous User Settings

Use **ctxanoncfg** to display the current number of anonymous user accounts, the naming of the accounts, the anonymous user group name, and the idle timeout period.

➤ **To display anonymous user settings**

1. Log on to the MetaFrame server as the root user.
2. At the command line, run **ctxanoncfg** with the **-l** option to display anonymous user settings:

```
ctxanoncfg -l
```



```
Terminal
Window Edit Options Help
$ ctxanoncfg -l
Anonymous group ctxanon
Anonymous user anon1
Anonymous user anon2
Anonymous user anon3
Anonymous user anon4
Anonymous user anon5
Anonymous user anon6
Anonymous user anon7
Anonymous user anon8
Anonymous user anon9
Anonymous user anon10
Anonymous user anon11
Anonymous user anon12
Anonymous user anon13
Anonymous user anon14
Anonymous user anon15
Anonymous session idle timeout is 10 minutes
$
```

Configuring Anonymous User Settings

You can use **ctxanoncfg** to change the number of anonymous user accounts, the names of the anonymous user accounts, and the idle timeout period for anonymous user sessions.

Tip Use the **-v** (verbose) option with the **ctxanoncfg** command to display additional information about the command as it executes. This displays what the command is doing at each stage and any errors that occur. For example, type:

```
ctxanoncfg -v -n 20
```

Changing the Number of Anonymous Users

Use **ctxanoncfg** with the **-n** option to change the number of anonymous user accounts. You can create any number of anonymous user accounts, but the number you can use simultaneously is limited by your licensed user count.

Further options are available that allow you to change the naming of anonymous user accounts and the idle timeout period.

Important MetaFrame must be stopped on the server before you create anonymous users. If MetaFrame is running, use the **ctxshutdown** command to stop it—see “Stopping MetaFrame” in Chapter 1 for further information.

➤ **To create anonymous users**

1. Ensure MetaFrame is not running on the server and log on as root.
2. At the command prompt, use **ctxanoncfg** with the **-n** option to specify the new number of anonymous user accounts you require:

```
ctxanoncfg -n number
```

where *number* is the new number of anonymous user accounts.
3. Start the MetaFrame process on the server—see “Starting MetaFrame” in Chapter 1 for instructions.

Examples

To specify 20 anonymous user accounts, type:

```
ctxanoncfg -n 20
```

To delete all anonymous user accounts, type:

```
ctxanoncfg -n 0
```

Changing the Naming of Anonymous User Accounts

Use **ctxanoncfg** with the **-b** option to change how anonymous user accounts are named. By default, the **ctxanon** group contains 15 user accounts with names in the format **anonx**, where *x* is a number from 1 to 15. User account names can have a maximum of 8 characters.

➤ **To change how anonymous user accounts are named**

1. After stopping the MetaFrame process on the server, log on as root.
2. At the command prompt, type:

```
ctxanoncfg -n number -b name
```

where *number* is the new number of anonymous user accounts, and *name* is the new name of the accounts.

Example

To create 25 anonymous user accounts called “guest1”, “guest2” ... up to “guest 25”, type:

```
ctxanoncfg -n 25 -b guest
```

Setting an Idle Timeout Period

Use **ctxanoncfg** with the **-t** option to specify the idle timeout period, in minutes, for anonymous user sessions.

If there is no user activity within this time, a warning message informs the user that they will be logged off after 5 minutes, unless they resume use of the session. The default idle timeout period is 10 minutes.

➤ **To specify an idle timeout period**

1. After stopping the MetaFrame process on the server, log on as root.
2. At the command prompt, type:

```
ctxanoncfg -n number -t minutes
```

where *number* is the new number of anonymous user accounts, and *minutes* is the idle timeout period.

Examples

To create 25 anonymous user accounts with a timeout period of 30 minutes, type:

```
ctxanoncfg -n 25 -t 30
```

To alter only the timeout period to 20 minutes, type:

```
ctxanoncfg -t 20
```

Configuring MetaFrame Security

This section describes the MetaFrame security function that controls user access to MetaFrame commands and sessions. It provides an overview of MetaFrame security, and it tells you how to display and configure security settings.

When you install MetaFrame, default security settings automatically control user access to various MetaFrame commands. By default, users can log on, send messages, and shadow other sessions, but they are denied access to all other MetaFrame commands. See “Default Security Settings” in this section for information about which functions are allowed or denied.

Why Use MetaFrame Security?

MetaFrame security lets you tighten or relax user access to MetaFrame commands and sessions. With MetaFrame security you can:

- Change the default security settings. For example, by default, all users can shadow other users’ sessions, but you may want to prevent this.
- Provide groups of users with access to MetaFrame commands and sessions. For example, you may want to give the “helpdesk” user group the rights to connect, disconnect, and reset other users’ sessions.
- Deal with exceptions on an individual user basis. For example, you may want to prevent a particular user from being able to send messages to other users’ sessions.

Security Overview

This overview explains which users are affected by security, which functions are secured, how security can be controlled at different levels, and how the security checking process works.

Which Users Are Affected by Security?

MetaFrame security controls user access to specific MetaFrame commands and sessions.

However, MetaFrame security does *not* control Citrix server administrator access to commands and sessions. In other words, the `ctxadm` group (for example, `ctxsrvr`) is unaffected by MetaFrame security. MetaFrame security controls the access rights of the root user and ordinary users (i.e., explicit and anonymous users).

Who Can Do What in MetaFrame

The following table provides a brief summary of which users can do what in MetaFrame. Only the functions indicated by “ctxsecurity” are controlled using MetaFrame security.

MetaFrame functions/commands	root user	ctxadm	other users
Install and remove MetaFrame	Yes	No	No
Activate Citrix Licensing	No	Yes	No
Start and stop server processes	Yes	Yes	No
Configure the MetaFrame server	No	Yes	No
Log on to the server	ctxsecurity	Yes	ctxsecurity
Query who is on the server	Yes	Yes	Yes
Perform actions on others' sessions, such as shadowing or resetting sessions	ctxsecurity	Yes	ctxsecurity
Perform actions on their own sessions	Yes	Yes	Yes

Note If root is unable to log on at the server, this may be due to the CONSOLE setting in the **/etc/default/login** file (the **/etc/security/user** file on AIX), which can be used to prevent root logging on at a terminal other than the one specified. The ctxsecurity command cannot be used to override the CONSOLE setting.

Which Functions Can ctxsecurity Control?

MetaFrame security controls access to specific MetaFrame functions called *secured functions*. The secured functions are shown in the following table:

Secured function	MetaFrame security determines
login	Which users can log onto the MetaFrame server.
sendmsg	Which users can use ctxmsg to send messages to other users' sessions.
connect	Which users can use ctxconnect to connect to other users' sessions.
disconnect	Which users can use ctxdisconnect to disconnect other users' sessions.
logoff	Which users can use ctxlogoff to log off other users' sessions.
reset	Which users can use ctxreset to reset other users' sessions.
shadow	Which users can use ctxshadow to shadow other users' sessions.

Controlling Security at Different Levels

Security can be controlled at the:

- User level—i.e. UNIX user level
- Group level—i.e. UNIX group level
- Global level—i.e. UNIX global level

A global security setting exists for every secured function. When you install MetaFrame, security is automatically controlled at the global level for each secured function. For example, by default, all users can send messages to other sessions.

However, you can also configure security for individual users, or for groups of users. For example, if you want to prevent the user “fred” from sending messages to other sessions, you can set up user-level security to deny fred access to `ctxmsg`. You can set up group-level security for the “Support” group to allow members of this group to reset other users’ sessions using `ctxreset`.

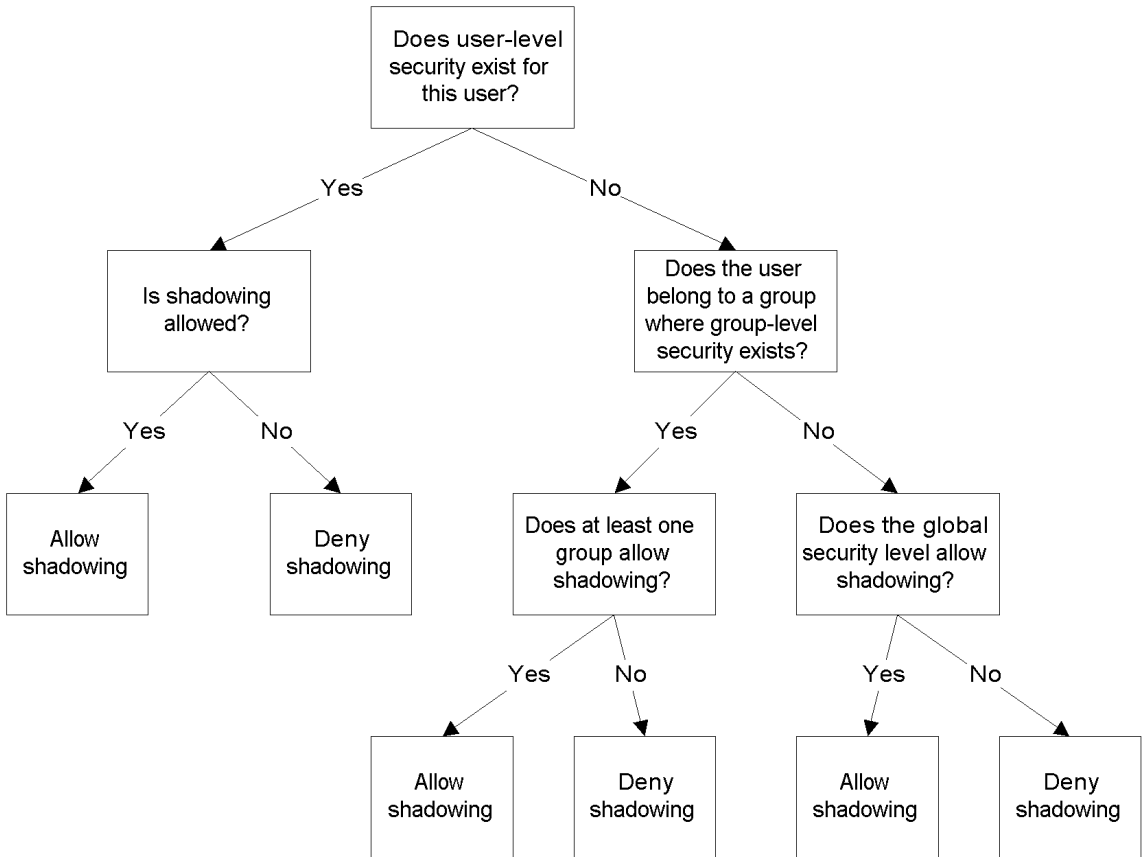
If no user or group-level security exists, the global security level determines user access rights.

The Security Checking Process

To configure MetaFrame security or troubleshoot security, you need to understand how the MetaFrame security checking process works.

When a user attempts to run a secured function, MetaFrame checks whether the user has the rights to do so. MetaFrame first checks the user security level, then, depending on the result, the group security level. If neither user nor group-level security exists, a final check is made at global security level.

The following diagram shows each step in the security checking process, using the example of a user attempting to run the `ctxshadow` command:



Default Security Settings

A global security setting always exists for each secured function. The global setting acts as the default, when neither user-level nor group-level security exists. Because the primary function of security is to deny access to unauthorized users, the global security setting can be thought of as the last line of defense.

After installation, the default settings are:

Secured function	Default global setting
Login	Allow
Sendmsg (ctxmsg)	Allow Deny for anonymous users
Connect (ctxconnect)	Deny
Disconnect (ctxdisconnect)	Deny
Logoff (ctxlogoff)	Deny
Reset (ctxreset)	Deny
Shadow (ctxshadow)	Allow Deny for anonymous users

Note By default, root cannot log on to the server from an ICA Client. However, if your super user has a different account name to “root” or multiple account names, he or she can log on from an ICA Client.

To change the default settings, see “Changing the Global Security Settings” in this section.

Displaying Security Settings for a Function

Use the **-l** (list) option with the **ctxsecurity** command to display security settings for a particular function. You must specify the secured function for which you want to display settings. All levels of security (user, group, and global) are displayed for the function.

➤ **To display security settings for a function**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxsecurity secured-function -l
```

Example

To display security settings for the ctxshadow function, type:

```
ctxsecurity shadow -l
```

Security settings such as the following are displayed:

```
global allow
group users deny
```

Tip Use **grep** to filter and identify specific information—for example, to display security settings for a particular user or group.

Configuring Security Settings

You can use the **ctxsecurity** command to change the global security settings or to configure user and group-level security.

Changing the Global Security Settings

You can use the **ctxsecurity** command with the **-a** (all) option to change the global security setting for a secured function.

➤ **To change a global security setting**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxsecurity secured-function -a allow|deny
```

Example

To change the global security setting for the ctxshadow tool to deny, type:

```
ctxsecurity shadow -a deny
```

Configuring Security for a User

You can use the **ctxsecurity** command with the **-u** (user) option to configure security at the user level. For example, you might want to allow a particular user access to a MetaFrame function that is denied at the global level.

➤ To configure security for a user

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxsecurity secured-function -u user-name allow|deny
```

Example

To allow the user “fred” to use the **ctxreset** command to reset other users’ sessions, type:

```
ctxsecurity reset -u fred allow
```

Configuring Security for Groups of Users

You can use the **ctxsecurity** command with the **-g** (group) option to configure security at the group level. For example, you might want to allow a group of users access to a MetaFrame function that is denied at the global level.

➤ To configure security for a group of users

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxsecurity secured-function -g group-name allow|deny
```

Example

To allow the group “support” to use the **ctxreset** command to reset other users’ sessions, type:

```
ctxsecurity reset -g support allow
```

Using Inherit to Remove Settings

You can use the **inherit** option with the **ctxsecurity** command to remove previously set security settings. This option is useful when you want to remove settings that are exceptional cases.

For example, the “management” group is allowed to shadow other sessions, but the user “fred”, a member of this group, is an exception and has been denied access to shadowing. When it is later decided to allow “fred” to shadow, the Citrix server administrator can use **inherit** to reinstate the group’s security setting. In effect, **inherit** removes fred’s user-level security setting, and picks up the security setting from group level.

Users can inherit settings from the group or global level, while a group can inherit settings from the global level. Global security settings cannot inherit values.

➤ **To inherit a security setting**

1. Log on to the MetaFrame server as a Citrix server administrator.
2. At the command prompt, type:

```
ctxsecurity secured-function {-u user-name|-g group-name} inherit
```

Example

To remove fred's user-level security setting for the ctxshadow command and reinstate the group's security setting, type:

```
ctxsecurity shadow -u fred inherit
```

The result is that the user "fred" inherits a security setting. MetaFrame checks which groups fred belongs to, and whether any of these groups have a group-level security setting. If at least one group-level setting exists that allows shadowing, fred inherits the right to shadow. Otherwise, if at least one group-level setting exists that denies shadowing, fred is not permitted to shadow. If no group-level setting exists, fred inherits rights from the global level.

Examples

In the following examples, MetaFrame security is used to tighten security, and then to provide a group of users with access to a particular MetaFrame function.

Example 1: Locking Down Security

After installation, the default MetaFrame security settings allow users to shadow other users' sessions. However, the Citrix server administrator decides to tighten security to prevent this.

The administrator does this by changing the global security setting for the shadowing function:

```
ctxsecurity shadow -a deny
```

Example 2: Giving Rights to a Group of Users

MetaFrame security is configured so that users are prevented from shadowing other users' sessions. However, the "helpdesk" group needs to be able to shadow so they can help users with problems.

The administrator uses MetaFrame security to provide the "helpdesk" user group with access to the shadowing function:

```
ctxsecurity shadow -g helpdesk allow
```

Configuring MetaFrame for Non-Standard Shells

Solaris

On the Solaris platform, MetaFrame allows the users of C, Bourne, and Korn shells to log on. By default, users of other, non-standard shells are prevented from logging on. However, you can configure MetaFrame to give access to users of other shells.

➤ **To configure MetaFrame for other shells**

1. Create a text file called **/etc/shells**; set the permissions to **rw-r--r--** and the owner to **root**.
2. In this file, include the full path name of all shells that are allowed to log on.

For example:

```
/usr/bin/sh
/usr/bin/csh
/usr/bin/ksh
/usr/bin/jsh
/usr/bin/bash
/bin/sh
/bin/csh
/bin/ksh
/bin/jsh
/bin/bash
/sbin/sh
/sbin/csh
/sbin/ksh
/sbin/jsh
/sbin/bash
```

Using ICA With Network Firewalls

Network firewalls can allow or block packets based on the destination address and port. If you are using ICA through a network firewall, use the information provided in this section to configure the firewall.

ICA TCP/IP Connection Sequence

1. The Citrix ICA Client sends a packet to port 1494 on the Citrix server requesting a response to a randomly selected port above 1023.
2. The Citrix server responds by sending packets to the Citrix ICA Client with the destination port set to the port requested in Step 1.

If you have a firewall or other TCP/IP network security, configure it to allow TCP/IP packets on port 1494 to pass to Citrix servers on your network.

Configure the firewall to allow TCP/IP packet on ports above 1023 to pass to Citrix ICA Clients.

If the firewall is not configured to pass ICA packets, users may receive the error, "There is no route to the specified address."

Note You can configure the Citrix server to use a different port number than 1494. See "Configuring the TCP/IP Port Number" in this chapter for details. Citrix ICA Clients must be configured to use the different port—see the *Citrix ICA Client Administrator's Guides* for the clients you plan to deploy.

The ICA Browser

The ICA Browser service uses UDP port 1604. ICA Browser responses are sent to a high port number above 1023.

The firewall must be configured to allow inbound UDP port 1604 packets to Citrix servers for load balancing, server farms, and ICA server browsing to function correctly. Note that with the Citrix XML Service you can avoid passing UDP through the firewall—for more information see the *Citrix XML Service Administrator's Guide*.

Warning Allowing untrusted access to the ICA Browser service entails some security risk. Configure the firewall to pass ICA Browser data only if load balancing and server browsing across the firewall are essential.

ICA Browsing With Network Address Translation

Some firewalls use IP address translation to convert private (Intranet) IP addresses into public (Internet) IP addresses. Public IP addresses are called “external” addresses because they are external to the firewall, whereas private IP addresses are said to be “internal” addresses.

Hosts on the internal network have one set of addresses that is translated to another set when passing through the firewall.

For example, an internal host has a private address 192.168.12.3. The firewall translates this into a different public address such as 206.103.132.20.

To browse Citrix servers and published applications, the Citrix ICA Client contacts a Citrix server and requests the address of the master ICA Browser. If the ICA Client is external to the firewall, it must be configured to use the public address of a Citrix server. The server returns the IP address of the current master ICA Browser to the ICA Client. By default, the IP address returned to the ICA Client is the internal address.

If the ICA Client is outside the firewall and the firewall is configured for address translation, the IP address returned to the client for the master browser will be incorrect.

Returning External Addresses to ICA Clients

Use the **ctxalt** command to configure the ICA Browser to return the external IP address to Citrix ICA Clients. You must configure every server that can be elected as the master ICA Browser.

The **ctxalt** command sets an alternate address for the ICA Browser on that machine. The external address for the server is specified as the alternate address. The Citrix ICA Client requests the alternate address when contacting servers inside the firewall. The alternate address must be specified for each server.

➤ To set an alternate address for a Citrix server

1. Determine the correct external IP address.
2. At the command prompt, type **ctxalt -a browser-address alternate-address**
See Appendix A for more information about the ctxalt command.
3. Repeat on each server.

In addition to specifying the alternate address on the Citrix server, the ICA Client must be configured to request the alternate address when contacting the master ICA Browser. For information about configuring ICA Clients to request the alternate address, see the *Citrix ICA Client Administrator's Guides* for the clients you plan to deploy.

Configuring the TCP/IP Port Number

By default, the TCP/IP port number used by the ICA protocol is 1494. You can change the port number using the **ctxcfg** command with the **-P** option.

The port number should be in the range 1024–65535 and must not conflict with other port numbers being used. Whenever the port number is changed, the server must be restarted for the new value to take effect.

Important If you change the port number on the MetaFrame server, you must also change it on every Citrix ICA Client that will connect to that server. For instructions about changing the port number on Citrix ICA Clients, see the *Citrix ICA Client Administrator's Guides* for the ICA Clients that you plan to deploy.

➤ **To display the current TCP/IP port number**

- 1. Log in as a Citrix server administrator at the MetaFrame server.
- 2. At the command prompt, type:
`ctxcfg -P list`



➤ **To change the TCP/IP port number**

- 1. Log in as a Citrix server administrator at the MetaFrame server.
- 2. At the command prompt:

To set	Use the command
The port number to the value <i>num</i>	<code>ctxcfg -P set=<i>num</i></code>
The port number back to the default, 1494	<code>ctxcfg -P reset</code>

Examples

To set the TCP/IP port number to 5000:

```
ctxcfg -P set=5000
```

To reset the port number to 1494:

```
ctxcfg -P reset
```

Configuring the Operating System for a Large Number of Connections

This section tells you how to configure your system for more than 30 connections. A large number of connections consumes resources, therefore it is important that you choose the optimum values for your environment.

This section also explains where to get more information on this topic.

Configuring a Solaris System

Solaris You may need to configure the total number of pseudo-terminals, or increase the limits on the number of files. Also, with the default configuration of Solaris, there is a limit to the number of concurrent CDE sessions that can be run, and you may need to increase this number.

For further information about how best to configure your system, see the *Solaris System Administrator Collection*.

Warning Be careful when using the **set** command in **/etc/system**—it causes unchecked, arbitrary, and automatic changes to variables in the kernel. If the server will not boot and you suspect a problem with **/etc/system**, use the **boot -a** command. See the **boot** man page for more information.

Changing the Number of Pseudo-Terminals

In a large number of MetaFrame connections, the number of ptys (pseudo-terminals) can easily surpass the default value (usually a MetaFrame session has at least one pty).

➤ To change the number of pseudo-terminals

1. Add the following lines to the **/etc/system** file:

```
# set limit on pseudo-terminals
set pt_cnt = 500
```

Note Do not set **pt_cnt** above 3000.

2. Shut down the server—for example, type:

```
init 0
```
3. Reboot the server:

```
boot -r
```

Increasing File Limits

There is a limit to the number of files a process can have open; the default value is 64. To increase the file limits for an individual process, use the **ulimit** command in a script before launching the process, as in the following example.

➤ **To change the file descriptor limits for all processes**

1. Add the following lines to the `/etc/system` file:

```
# set hard limit on file descriptors
set rlim_fd_max = 4096
# set soft limit on file descriptors
set rlim_fd_cur = 256
```

2. Reboot the server:

```
boot -r
```

Increasing the Number of Concurrent CDE Sessions

With the default configuration of Solaris, there is a limit to the number of concurrent CDE sessions that can be run (approximately 60, depending upon session configuration). This is due to the tooltalk database reaching a limit of available file descriptors. However, you can increase the number of possible concurrent CDE sessions.

➤ **To increase the limit on concurrent CDE sessions**

Check to see if the file `/usr/dt/bin/rpc.ttdbserverd` is a link to: `/usr/openwin/bin/rpc.ttdbserverd`.

If the file is a link, do the following (if it is not a link, see later):

1. Remove the file `/usr/dt/bin/rpc.ttdbserverd`:
2. Replace the link with the following script file. In this example, `ulimit` is used to increase the limit to 1024:

```
#!/bin/sh
ulimit -n 1024
exec /usr/openwin/bin/rpc.ttdbserverd
```

3. Make the file executable:
4. Kill the currently running `rpc.ttdbserverd` process.
5. Restart `rpc.ttdbserverd` to ensure the new limit is applied.

➤ **If the file is not a link, do the following:**

1. Edit the file **/usr/dt/bin/rpc.ttdbserverd** and change the limit.
2. Kill the currently running **rpc.ttdbserverd** process.
3. Restart **rpc.ttdbserverd** to ensure the new limit is applied.

If the Database Gets Corrupted

Files in **/TT_DB** occasionally get corrupted, and messages such as the following may appear in your **/var/adm/messages** file:

```
/usr/dt/bin/ttsession[11627]: Error: rpc.ttdbserverd on 127.0.0.1 is
not running
/usr/dt/bin/ttsession[11627]: _Tt_db_client::connectToDb():
fcntl(F_SETFD): Bad file number
/usr/dt/bin/ttsession[11627]: _Tt_db_file::_Tt_db_file(): _file_cache-
>insert(<your hostname>:/etc/tt/types.xdr), dbStatus 16
```

If you suspect that the database is corrupted, remove all the files in the **/TT_DB** directory and repeat Steps 4 and 5. Restarting the server automatically creates new database files.

Configuring an HP-UX System

HP-UX This section provides guidelines for configuring your HP-UX system for more than 30 connections.

For further information about how best to configure your system, see the relevant white papers on Hewlett-Packard's website at:
<http://docs.hp.com/hpux/>.

➤ **To configure your HP-UX system for more than 30 connections**

1. Choose **System_Admin** from the **Application Manager**.
2. Choose **Sam**.
3. Enter the root password at the prompt. The **System Administration Manager** dialog is displayed.
4. Choose **Kernel Configuration**.
5. Choose **Configurable Parameters**. The **Kernel Configuration** dialog is displayed.
6. Update your system with the following settings. This tunes your system to run multiple processes (each of which may have many threads and open files) and increases the number of users that can log on concurrently.

Note Change the value of **maxusers** first—this allows you to update the other settings.

Parameter	Setting	Description
maxusers	1000 (or as required)	Allocates system resources for macros on expected maximum users
dbc_max_pct	20	Maximum dynamic buffer cache
max_thread_proc	2048	Maximum threads per process
maxfiles	2048	Soft limit of files per process
maxfiles_lim	2048	Hard limit of files per process
maxssiz	134217728	Maximum process storage segment size
maxssiz_64bit	134217728	Maximum process storage segment size—64bit
maxswapchunks	4096	Maximum swap space configurable on the system
nflocks	3461	Maximum number of file locks on the system
npty	3461	Maximum number of ptys (pseudo-terminals) on the system

Configuring an AIX System

AIX

This section provides guidelines for configuring your AIX system for more than 30 connections.

For further information about how best to configure your system, see the relevant white papers on IBM's website at: <http://www.ibm.com/>.

Changing the Number of Pseudo-terminals

In a large number of MetaFrame connections, the number of ptys (pseudo-terminals) can easily surpass the default value (usually a MetaFrame session has at least one pty).

➤ **To change the number of pseudo-terminals**

1. Log on as root at the server that you want to configure.
2. Type **smit**. The **System Management Interface Tool** dialog is displayed.
3. Choose **Devices**.
4. Choose **PTY**.
5. Choose **Change/Show Characteristics of the PTY**.
6. Change the number of **Pseudo-Terminals**.
7. Select **OK**. **pty0 changed** is displayed.

Increasing the Number of Processes Per User

On AIX, by default, there is a limit to the number of processes a user can have running simultaneously. The default is 128 processes per user.

If a user runs out of processes he or she will be unable to run any commands or log off from the session until more processes are made available. For example, this situation may occur in a training scenario where several users are logged on to the MetaFrame server using only the one training user id.

You can increase the number of processes a user can run simultaneously using SMIT.

➤ **To increase the number of processes per user**

1. Log on as root at the MetaFrame server.
2. Type **smit**. The **System Management Interface Tool** dialog is displayed.
3. Choose **System Environments**.
4. Choose **Change/Show Characteristics of Operating System**.
5. Increase the value of **Maximum number of PROCESSES allowed per user (Num)**.

MetaFrame Command Reference



Overview

This appendix describes the MetaFrame command line utilities. The commands listed in this appendix are:

ctx3bmouse	configure 3-button mouse emulation
ctxalt	alternate address configuration for ICA browsers
ctxanoncfg	configure anonymous users
ctxappcfg	configure published applications
ctxbrcfg	configure ICA browser settings
ctxcapture	graphics cut and paste (between ICA and local applications)
ctxcfg	configure Citrix server settings
ctxconnect	connect to a session
ctxdisconnect	disconnect from a session
ctxgrab	graphics cut and paste (from ICA to local applications)
ctxlicense	configure Citrix licensing
ctxlogoff	log off from a Citrix server
ctxlpr	print to a client printer
ctxmaster	show master ICA browser
ctxmsg	send a message to a session
ctxprinters	list printers installed on the client
ctxqserver	display information about Citrix servers
ctxqsession	display session details
ctxquser	display session user details
ctxreset	reset a session
ctxsecurity	configure MetaFrame security
ctxshadow	start a shadowing session
ctxshutdown	shut down the Citrix server processes
ctxsrv	start up or stop the Citrix server processes

ctx3bmouse

Description

ctx3bmouse configures 3-button mouse emulation.

You may need to use MetaFrame to deploy UNIX applications that are designed for use with a 3-button mouse. However, many ICA Clients run on devices that have only a 2-button mouse, 1-button mouse, or pointing device available.

To do this, you publish another version of the application for use by these ICA Clients. This version of the application is published using a script file that includes ctx3bmouse settings. The ctx3bmouse command lets users represent a missing mouse button by combining an existing mouse button with a modifier key. For example, a missing button might be simulated by clicking the left mouse button and pressing the SHIFT key.

By running a script file that includes ctx3bmouse settings, you ensure the application is run in a session with the appropriate mouse mappings.

Syntax

ctx3bmouse	<i>missing_button=mouse_button, number_of_modifier_key</i>
ctx3bmouse	-r
ctx3bmouse	-c

Options

-r	Display mouse mappings for the current session.
-c	Clear all mouse mappings for the current session.

Parameters

<i>missing_button</i>	The missing button which is to be emulated: left middle right
<i>mouse_button</i>	The existing mouse button which, when pressed with the modifier key, simulates the missing mouse button.
<i>number_of_modifier_key</i>	Number of modifier to use. Use the xmodmap command to show which keys correspond to which modifiers.

Remarks

With xmodmap it is possible to remap almost any aspect of the keyboard and mouse. Take care when using xmodmap with ctx3bmouse because the combination may be confusing.

ctxalt

Description

ctxalt specifies alternate address configuration for ICA Browsers.

Syntax

ctxalt	-l
ctxalt	-d <i>alt_addr</i>
ctxalt	-a <i>browser_addr alt_addr</i>
ctxalt	-r <i>addr</i>

Options

-l	List current alternate address configuration.
-d	Set the default alternate address.
-a	Set an alternate address.
-r	Remove an alternate address.

Parameters

<i>alt_addr</i>	Specifies the alternate address. With the exception of the -r option, all addresses must be supplied in standard IP address format. The -r option also accepts the (case insensitive) keyword DefaultAddress to erase the default address setting.
<i>browser_addr</i>	Specifies the default alternate address.

Remarks

You must be a Citrix server administrator to run this command.

ctxanoncfg

Description

ctxanoncfg configures anonymous users.

Syntax

ctxanoncfg	-l
ctxanoncfg	[-v] [-n <i>number</i>] [-b <i>anonymous_user_name</i>] [-t <i>minutes</i>]

Options

-l	List current anonymous user settings.
-n	Specify the number of anonymous user accounts.
-b	Change how anonymous user accounts are named.
-v	Verbose output. Use with the other options to display what the command is doing at each stage and any errors that occur.
-t	Specify the idle timeout period, in minutes, for anonymous user sessions.

Parameters

<i>number</i>	New number of anonymous user accounts.
<i>anonymous_user_name</i>	New name of anonymous user accounts. By default, account names are in the format anonx where x is a number from 1,2 ... etc.

Remarks

You must be root to run this command.

You must stop the MetaFrame process on the server before you configure anonymous users.

See also

ctxshutdown—to stop the MetaFrame process.

ctxappcfg

Description

ctxappcfg is an interactive command that allows you to publish and configure applications.

Syntax

ctxappcfg

Usage

When you run **ctxappcfg**, the **App Config>** command prompt is displayed and the following commands can be entered:

list	Displays a list of published application names.
add	Allows you to publish an application. You are prompted for the following details: Name - the name used to refer to the published application. Command Line - the command line used to launch the application. To publish the desktop, press ENTER without specifying a command line. Working Directory - the working directory used by the application. To specify the user's home directory, leave this blank. Anonymous - whether the application is for use by anonymous or explicit users. Enter y if the application is for use by anonymous users only, or n if it's for explicit users only.

select [name]	Allows you to configure a published application. If you do not specify the name of the application you want to configure, you are prompted for it. Note that the application name is case-sensitive. When you have selected an application, the prompt changes to the name of the application and you can enter the following commands: list - lists the configuration details of the selected application. set - allows you to change the configuration. The full syntax is: set [cmd=cmd_line,] [dir=dir_name,] [anonymous=yes no] [enabled=yes no] where the parameters are as follows: cmd - the command line required for the program to run. dir - the initial working directory. anonymous - indicates if the application is for use by anonymous or explicit users. enabled - indicates if the application is enabled or disabled. copy - creates a new published application by copying the configuration of the current application. You are prompted to enter a name for the new application. The new configuration is saved and automatically selected. save - saves the changes you make. delete - deletes the currently selected application and returns you to the App Config> prompt. drop - de-selects the currently application and returns you to the App Config> prompt. help / ? - displays a brief usage message. exit - exits the command.
help / ?	Displays a brief usage message.
exit	Exits the program.

Remarks

You must be a Citrix server administrator to run this command.

See also

ctxqserver—to list all published applications on the network.

ctxbrcfg

Description

ctxbrcfg configures ICA browser settings.

Syntax

ctxbrcfg -g [add=*gateway*,] [remove=*gateway*,] [list]

ctxbrcfg -m [always | never | neutral,] [list]

ctxbrcfg -r [set=*num*,] [list]

Options

-g	Gateways. Allows you to add or remove gateways.
-m	Master election. Allows you to influence the criteria used for the master election. always makes the local browser try to become the master. never instructs the browser to refrain from standing in an election. neutral reinstates the default behavior of “no preference”.
-r	Refresh period. Allows you to specify the interval (in minutes) at which the local browser will update the master browser.

Parameters

<i>num</i>	Specifies the interval (in minutes) at which the local browser will update the master browser.
<i>gateway</i>	Specifies the gateway host name or IP address.

Remarks

You must be a Citrix server administrator to run this command.

See also

ctxqserver—to display information about gateways and the master browser.

ctxcapture

Description

ctxcapture lets you:

- Grab windows 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 xv. xv is a Shareware utility that is available for download from the Internet.

ctxcapture is available from the command prompt; it is also available when you connect to published applications through the **ctxwm** window manager, if your Citrix server administrator has made it available, as follows:

- In a seamless window, right click the  button in the top, left hand corner of the screen to display a menu and choose the **Screen Grab** option.
- In a “full screen” window, right click to display the **ctxwm** menu and choose the **Screen Grab** option.

When ctxcapture is started a dialog box is displayed.

Syntax

ctxcapture

See also

ctxgrab—a simple tool to cut and paste graphics from ICA applications to applications running locally on the client device.

ctxcfg

Description

ctxcfg configures Citrix server settings.

Syntax

```
ctxcfg -a [ERASE | [[prompt={TRUE | FALSE},] [INHERIT |
[user=name,] [pass]] [list]

ctxcfg -l [max={UNLIMITED | num } ] [list]

ctxcfg -t [connect={NONE | minutes },] [disconnect={NONE |
minutes },idle={NONE | minutes}] [list]

ctxcfg -c [broken={DISCONNECT | RESET},] [reconnect={ORIGINAL |
ANY},] [list]

ctxcfg -p [enable | disable] [list]

ctxcfg -C [enable | disable] [list]

ctxcfg -P [set=num | reset] [list]

ctxcfg -g

ctxcfg -e [require={none | basic}] [list]

ctxcfg -i [ INHERIT | PUBONLY | ([prog=name,][wd=dir))] [list]

ctxcfg -s [disable] [enable ([input={on | off},]
[notify={on | off }])] [list]
```

Options

-a Allows you to set automatic logon details. Use **INHERIT** to make the server use logon details specified on the client, rather than setting a user name and password for the server using **user** and **pass**. Alternatively you can specify a username and/or password for all users who log on to the server. Set **prompt** to **TRUE** to prompt users for a password, regardless of whether one is specified on the server or the client. Use the **pass** option to prompt users for a logon password. Note that using **-g** with the list option will not display the password.

ERASE erases any user name and password details that have been set using the **user** and **pass** options and makes the server use logon details specified on the client.

- l** Logons. Allows you to limit the number of users that can be concurrently logged on to the server. Specify an unsigned number or the keyword **UNLIMITED** to allow an unlimited number of users to log on.
- t** Timers. Allows you to specify timeout intervals (in minutes) for connected, disconnected, and idle sessions. For example, to specify a timeout interval of 10 minutes for disconnected sessions, use **-t disconnect=10**. Use the keyword **NONE** to disable all timeout settings.
- c** Connections. Allows you to define how the server handles timed out or broken sessions.

Set **broken** to **DISCONNECT** to disconnect sessions that are broken; set to **RESET** to terminate broken sessions. Set **reconnect** to **ORIGINAL** to allow reconnection only to a broken or timed out session from the original terminal; set to **ANY** to allow reconnection to the session from any terminal.
- p** Client printing. Use to enable or disable client printing.
- C** Client clipboard. Use to enable or disable the client clipboard.
- P** Port number. Use to specify a TCP/IP port number for the ICA server to listen for connections on. Use **set** to use a specific number, or **reset** to use the default number. You must restart the server for the new value to take effect.
- g** Generate. This generates a list of commands that, if executed, restores all settings to their current values (except the password). You can redirect these commands to a file that you can later execute as a shell script. **-g** cannot be used with any other argument.
- e** Encryption. Use to force clients to use encryption and prevent clients who do not use encryption from connecting.
- i** Initial program. Use to specify a program, and path if necessary, to run when the client initially connects. **INHERIT** uses the program and path specified on the client. **PUBONLY** prevents clients running any applications other than those published on the server.
- s** Shadowing. Use to enable or disable shadowing. Set **input** to **on** to allow the shadower to interact with the shadowed session using the keyboard and mouse. Set **notify** to **on** to give users the option to approve or deny the shadowing of their session.

Note that when enabling shadowing, the default for **input** is **on** and the default for **notify** is **on**.

Remarks

You must be a Citrix server administrator to run this command.

ctxconnect

Description

ctxconnect lets you connect to a session.

Syntax

```
ctxconnect      id
```

Parameters

<i>id</i>	Specifies the session id to connect to.
-----------	---

Remarks

By default, Citrix server administrators can connect to any session; other users can connect only to their own sessions.

See also

ctxsecurity—to control which users can connect to other users' sessions.

ctxdisconnect

Description

ctxdisconnect lets you disconnect a session.

Syntax

```
ctxdisconnect  [id | name]
```

Parameters

<i>id</i>	Specifies the session id to disconnect.
<i>name</i>	Specifies the session name to disconnect.

Remarks

If you do not specify a session id or session name, your own session is disconnected. By default, Citrix server administrators can disconnect any session; other users can disconnect only their own sessions.

See also

ctxsecurity—to control which users can disconnect other users' sessions.

ctxgrab

Description

ctxgrab lets you:

- Grab dialogs or screen areas and copy them from an application in an ICA Client window to an application running on the local client device.

ctxgrab is available from the command prompt or, if you are using a published application, from the ctxwm window manager, as follows:

- In a seamless window, right click the  button in the top, left hand corner of the screen to display a menu and choose the **Screen Grab** option.
- In a “full screen” window, right click to display the **ctxwm** menu and choose the **Screen Grab** option.

When ctxgrab is started a dialog box is displayed.

Syntax

ctxgrab

See also

ctxcapture—a more extensive tool that lets you cut and paste graphics between ICA applications and applications running on the client device.

ctxlicense

Description

ctxlicense manages Citrix licensing.

Citrix licensing controls the number of ICA connections permitted on your Citrix server, and the features available on the server .

Syntax

ctxlicense	-add [-noprompt] <i>serial_number</i>
ctxlicense	-activate <i>license_number activation_code</i>
ctxlicense	-list
ctxlicense	-pool <i>license_number pooled_count</i>
ctxlicense	-remove <i>license_number</i>

Options

-add	Add a Citrix license to the server. If you are installing an upgrade license, use the -noprompt option to add and activate the base and upgrade licenses in any order.
-activate	Activate a Citrix license using the generated license number and supplied activation code.
-list	Display information about all licenses present on the Citrix server.
-pool	Change the pooled user count.
-remove	Remove a license from a Citrix server. If you are removing an upgrade license, you must remove the upgrade license before removing the base license.

Parameters

<i>serial_number</i>	The 25 character number on the sticker on the back of the CD booklet.
<i>license_number</i>	The 35 character number generated by MetaFrame when you add a license. The license number is based on the serial number.
<i>activation_code</i>	The code supplied by Citrix that validates and enables the license.
<i>pooled_count</i>	The user count shared (pooled) by all Citrix servers on the same network subnet.

See also

ctxqserver—to display information about licenses on Citrix servers on the network.

ctxlogoff

Description

ctxlogoff logs off a user from the MetaFrame server.

Syntax

<code>ctxlogoff</code>	<code>[id name]</code>
------------------------	---------------------------

Parameters

<i>id</i>	Specifies the session id to log off.
<i>name</i>	Specifies the session name to log off.

Remarks

If a user is not specified, you are logged off.

By default, Citrix server administrators can log off any user; other users can log only themselves off.

See also

ctxsecurity—to control which users can log off other users' sessions.

ctxlpr

Description

ctxlpr prints to a client printer.

Syntax

```
ctxlpr          [-P printerName] [-b] [-n] [file1, ... file10]
```

Options

-
- | | |
|-----------|---|
| -P | Print a file to a printer (or printer port) other than the default. This is the printer name or printer port shown in the first column of the output from ctxprinters . |
| -b | Print the job in the background. |
| -n | Only one print job can be handled at a time in any one session. If a call is made to ctxlpr while a previous job is still printing, the default behavior is for the second command to wait for the first job to end before continuing.
Use the -n option to cause a second print job to fail rather than wait. Use this to stop applications waiting while other printer jobs are handled. |

Parameters

-
- | | |
|--------------------|--|
| <i>file</i> | Specifies the name of a file to print. Up to ten files can be specified; each file is treated as a separate print job. If no files are specified, ctxlpr takes its input from standard input (stdin). |
| <i>printerName</i> | Name of the printer (or printer port) other than the default. |

See also

ctxprinters—to list printers installed on the client.

ctxmaster

Description

ctxmaster shows the master ICA Browser address.

Syntax

ctxmaster

See also

ctxqserver—to display the master browser address.

ctxmsg

Description

ctxmsg sends a message to a session.

Syntax

ctxmsg [-w] {*id* | *name*} *message* [*timeout*]

Options

-w	Suspends the ctxmsg program until the message either times out or the user dismisses it. In other words, you get your command prompt back only when the user responds or the message times out.
----	---

Parameters

<i>id</i>	Session id of the user to whom you want to send the message.
<i>name</i>	Session name of the user to whom you want to send the message.
<i>message</i>	To send a message that contains spaces, enclose it within double quotes.
<i>timeout</i>	Specify a timeout (in seconds) for the message. If no timeout is specified, the message dialog is displayed until dismissed by the user.

See also

ctxquser or ctxqsession—to display users' session names and IDs.

ctxsecurity—to control which users can send messages to other users' sessions.

ctxprinters

Description

ctxprinters lists printers installed on the client and indicates which is the default.

For each printer, the list displays:

- The printer name or printer port (for example, lpt1). You can use this in the **ctxlpr -P** command to specify a printer other than the default.
- The name of the device driver.
- The name of the port to which the printer is attached.

Syntax

ctxprinters

See also

ctxlpr—to print to a client printer.

ctxqserver

Description

ctxqserver displays information about Citrix servers on the network.

Syntax

ctxqserver	[<i>server_name</i>]
ctxqserver	-addr <i>server_name</i>
ctxqserver	-app [<i>application_name</i> <i>server_name</i>]
ctxqserver	-disc [<i>application_name</i> <i>client_name</i>]
ctxqserver	-gateway [<i>server_name</i>]
ctxqserver	-gatewaylicense: <i>IP_address</i>
ctxqserver	-license [<i>server_name</i>]
ctxqserver	-load <i>server_name</i>
ctxqserver	-master
ctxqserver	-netlicense
ctxqserver	-ping [-count: <i>value</i> -size: <i>value</i>]
ctxqserver	-reset <i>server_name</i>
ctxqserver	-serial [<i>server_name</i>]
ctxqserver	-stats <i>server_name</i>
ctxqserver	-update <i>server_name</i>

Options

-addr	Display the network address of a specific Citrix server.
-app	List all published applications and the server load. Specify the name of an application or server to narrow the list.
-disc	List all disconnected sessions. Specify the name of an application or client to narrow the list.
-gateway	List the ICA gateways known to each server. Specify a server name to narrow the list.
-gatewaylicense	Display the number of remote licenses available from a gateway. Specify the IP address of the gateway; e.g., ctxqserver -gatewaylicense: 12.12.123.12

-license	List the licenses on each server. Mach displays the number of licenses kept local to the machine; Pool displays the number of pooled licenses; Total shows the sum of the local and pooled licenses. Specify a server name to narrow the list.
-load	Display the loading for a particular server.
-master	Display the IP address of the master browser.
-netlicense	Display information about the number of licenses installed and in use on the local network. The number of licenses kept local to the machine and the number pooled is also shown.
-ping	Ping the named server.
-reset	Reset statistics about the activities of the browser (e.g., elections sent/received, packets sent/received) for the named server.
-serial	Display the licenses on each server. Specify a server name to narrow the list.
-stats	Display statistics about the activities of the browser for a particular server.
-update	Request the named server to update the master browser with its data.

Parameters

<i>server_name</i>	Name of a specific Citrix server.
<i>application_name</i>	Name of a published application.
-count : <i>value</i>	Use with the ping option to specify the number of packets to send. The default is five packets.
-size : <i>value</i>	Use with the ping option to specify the packet size. The default is 256 bytes.
<i>IP_address</i>	TCP/IP address of a server.

ctxqsession

Description

ctxqsession displays session details.

ctxqsession displays information about ICA connections to the Citrix server. The information includes, where appropriate, user name, session name, session ID, state, type, and device.

Syntax

ctxqsession

See also

ctxquser—to display session user details.

ctxquser

Description

ctxquser displays session user details.

ctxquser displays information about users logged on to the Citrix server. The information displayed includes the user name, the session name, the session ID, the state, the time the user has been idle, and the total time the user has been logged on.

Syntax

ctxquser [user *usr_name*]

See also

ctxqsession—to display session details.

ctxreset

Description

ctxreset resets a session.

ctxreset resets an ICA connection to a Citrix server. You can specify the session to be reset using its session id or name.

Syntax

ctxreset *{id | name}*

Remarks

By default, Citrix server administrators can reset any session; other users can reset only their own sessions.

See also

ctxqsession—to display the current sessions.

ctxquser—to display session user details.

ctxsecurity—to control which users can use ctxreset to reset other users' sessions.

ctxsecurity

Description

ctxsecurity configures MetaFrame security.

MetaFrame security controls a user’s access to MetaFrame commands and sessions. When you install MetaFrame, default security settings are applied that automatically control access at a global level to MetaFrame-secured functions. Security can also be controlled at user and group level.

Syntax

ctxsecurity	<i>secured_function -l</i>
ctxsecurity	<i>secured_function -a {allow deny}</i>
ctxsecurity	<i>secured_function -u {user_name} {allow deny}</i>
ctxsecurity	<i>secured_function -g {group_name} {allow deny}</i>
ctxsecurity	<i>secured_function {-u user_name -g group_name} inherit</i>

Options

-l	Display security settings for a particular secured function.
-a	Change the global security setting for a secured function.
-u	Change security at user level for a secured function.
-g	Change security at group level for a secured function.

Parameters

<i>secured_function</i>	A particular MetaFrame tool; e.g., shadow.
<i>allow</i>	Permit access to the secured function.
<i>deny</i>	Prevent access to the secured function.
<i>user_name</i>	User account name.
<i>group_name</i>	Group name to which the user belongs.
<i>inherit</i>	Remove previous user or group security settings and inherit settings from the level above.

Remarks

You must be a Citrix server administrator to run this command.

ctxshadow

Description

ctxshadow starts a shadowing session. Shadowing lets you monitor and interact with another active session.

The session that issues the ctxshadow command is referred to as the *shadower*, and the session being shadowed is called the *shadowed* session.

Syntax

```
ctxshadow {id | name} [-v] [-h[[a][c][s]+] x]
```

Options

-v	Verbose output. Displays additional information.
-h	Configure a hotkey combination to end shadowing.

Parameters

id	Specify the session to be shadowed using its session ID.
name	Specify the session to be shadowed using its session name.
{a c s}+ x	Specify the hotkey combination you want to use to end shadowing. Choose this combination from: a c s where a = ALT; c = CTRL; s = SHIFT x where x is an alphanumeric character (a to z and 0 to 9). (Note: you can use any combination of a, c and s, including all or none.) For example, to begin shadowing and to specify a hotkey combination of ALT and q to stop shadowing, type: ctxshadow {id name} -h a+q

To End a Shadowing Session

By default, you can end shadowing by holding down the CTRL key and pressing the * (asterisk) key on your keyboard's numeric keypad. However, if you cannot use this hotkey combination, or you prefer to use an alternative, you can configure a different combination using the **-h** option.

Remarks

Note that virtual channel data (instructions to the server that affect only the shadowed session) is not shadowed. For example, if you print a file while shadowing a session, the file is queued at the shadowed session's printer.

You may also get some unexpected results using the clipboard channel. The user of the shadowed session can use the clipboard to copy and paste between the client session and applications running locally. As shadower, you cannot access the contents of the shadowed session's clipboard—information in the clipboard belongs to the shadowed session. However, if you copy information to the clipboard while shadowing, this information is available to the shadowed session for pasting.

See also

ctxcfg—for shadowing configuration at the server.

ctxquser or ctxqsession—to display session name and ID.

ctxsecurity—to control which users can shadow other users' sessions.

ctxshutdown

Description

ctxshutdown stops the Citrix server processes.

Syntax

```
ctxshutdown [-q] [-m seconds] [-l seconds] [message]
```

Options

-
- | | |
|-----------|---|
| -q | Quiet mode. Use to reduce the amount of information displayed to the administrator by the ctxshutdown command. |
| -m | Specify when the shut down process will begin, and how long the message will be displayed, in seconds. The default is 60 seconds. When this period expires and the shut down process begins, applications that have registered “window hints” (the WM_DELETE_WINDOW attribute) will attempt to interactively log the user off. Applications that have not registered “window hints” will terminate immediately. |
| -l | Specify how long applications that have registered “window hints” have to interactively log users off. The default is 30 seconds. When this period expires, any remaining sessions are automatically terminated, users are automatically logged off, and the MetaFrame process stops. |

Parameters

-
- | | |
|-----------------------|--|
| <i>message</i> | Specify the message displayed to all users logged on to the server. If you do not specify a message, the default message “Server shutting down. Auto logoff in <i>x</i> seconds” is displayed, where <i>x</i> = the number of seconds specified in the -m option (or the default of 60 seconds if this is not specified). |
|-----------------------|--|

Remarks

You must be a Citrix server administrator to run this command.

See also

ctxsrv—to stop the MetaFrame processes on a server.

ctxsrv

Description

ctxsrv starts up or stops the server processes.

You can use **ctxsrv** to start up and stop all the processes on the server, or to start up and stop an individual process, such as the ICA Browser.

Syntax

ctxsrv **start** [**browser** | **all**]

ctxsrv **stop** {**browser** | **all**}

Remarks

We recommend you use the **ctxshutdown** command to stop the MetaFrame processes on a server. If you use **ctxsrv** to stop MetaFrame, and sessions are still active when the server is stopped, the sessions are terminated and unsaved applications or user data can be lost.

You must be root or a Citrix server administrator to run this command.

See also

ctxshutdown—to shut down the MetaFrame processes on a server.

Glossary



Overview

This appendix provides a glossary of terms and acronyms used throughout the MetaFrame documentation.

A

activation code	The code you use to activate your Citrix software licenses. To get an activation code from Citrix, you contact the Citrix Activation System (CAS) and enter the license number.
ALE	See Application Launching and Embedding.
alternate address	The external address of a Citrix server. An external address is a public (Internet) IP address.
anonymous application	An application published exclusively for use by anonymous users.
anonymous session	The session of an anonymous user.
anonymous user	A guest user granted restricted access to a published application on a Citrix server.
anonymous user account	A user account defined on a Citrix server for accessing applications published for anonymous use.
Application Launching and Embedding (ALE)	A feature of MetaFrame and Citrix ICA Clients that enables full-function applications to be launched from or embedded into HTML pages without rewriting any application code.

B

base license	A type of Citrix license that enables the multiuser features of your Citrix server. If a base license is not present, ICA connections are not supported.
Bourne shell	A type of UNIX shell. In Solaris, Bourne is the default shell.
broken session	A broken session occurs when the communication link between an ICA Client and the MetaFrame server is interrupted, for example, as the result of a client device failure.
browser election	See master browser election.

C

C shell	A type of UNIX shell, based on the C programming language.
CDE	See Common Desktop Environment.
Citrix Connection Configuration	The Citrix utility you use to configure ICA and other connections to your MetaFrame servers.
Citrix Licensing	The utility you use to add, activate, configure and remove licenses and user counts.
Citrix server	Any MetaFrame, <i>WINFRAME</i> or VideoFrame server on which you publish applications or videos.
Citrix server administrator	A member of the user group ctxadm, who has special permissions regarding the administration of MetaFrame for UNIX Operating Systems.
client device	Any device capable of running one of the Citrix ICA Clients.
client device licensing	A feature of Citrix licensing that allows users to start multiple sessions on the same server, or on different servers, while using only a single user count. All connections must be from the same client device.
client device mapping	The feature that enables remote applications running on the Citrix server to access storage and peripherals attached to the local client device.

client printer mapping	See client device mapping.
Common Desktop Environment (CDE)	A standard desktop for UNIX that uses windows, icons, and menus to provide services to end-users, systems administrators, and application developers across platforms.
connection	See ICA connection.
csh	See C shell.
ctx3bmouse	Command line tool for configuring 3-button mouse emulation.
ctxadm	The Citrix server administrator group name. See Citrix server administrator.
ctxalt	Command line tool for alternate address configuration.
ctxanoncfg	Command line tool for configuring anonymous users.
ctxappcfg	Command line tool for controlling published applications.
ctxbrcfg	Command line tool for configuring ICA Browser settings.
ctxcapture	Command line tool for graphics cut and paste (between ICA and local applications).
ctxcfg	Command line tool for configuring MetaFrame servers.
ctxconnect	Command line tool for connecting to a session.
ctxdisconnect	Command line tool for disconnecting a session.
ctxgrab	Command line tool for graphics cut and paste (from ICA to local applications).
ctxlicense	Command line tool for licensing your MetaFrame software.
ctxlogout	Command line tool for logging off the MetaFrame server.
ctxlpr	Command line tool for printing to a client printer.
ctxmaster	Command line tool for showing the master ICA browser address and alternate address.

ctxmsg	Command line tool for sending messages to sessions.
ctxprinters	Command line tool for listing the printers installed on the client.
ctxqserver	Command line tool for displaying information about Citrix servers on the subnet.
ctxqsession	Command line tool for displaying session details.
ctxquser	Command line tool for displaying session user details.
ctxreset	Command line tool for resetting a session.
ctxsecurity	Command line tool for displaying and configuring user access to MetaFrame commands and sessions.
ctxshadow	Command line tool for shadowing a user's session.
ctxshutdown	Command line tool for stopping the MetaFrame process on a server.
ctxsrv	Command line tool for starting up or stopping the server processes on a Citrix server.
ctxsrvr	Default member of the Citrix server administrator group.

D

dtterm	The standard terminal emulator used by CDE.
disconnected session	An ICA session in which the ICA Client is no longer connected to the Citrix server. However, the user is not logged off, their applications are still running, and the user can reconnect to the disconnected session.

E

explicit user	A user who has an account name and password. Explicit users log on using user accounts created and maintained by system administrators, and when they log on they supply a username and password.
external address	The public (Internet) IP address of a Citrix server.

F

firewall	A network node that provides security by controlling traffic between network segments.
-----------------	--

G

grace period	A 35 day period which some Citrix licenses have, after they are added. During this period, the MetaFrame software is fully functional and connections to the server work normally. However, if the license is not activated by the end of the grace period, the MetaFrame software is reduced to single user mode, and only one ctxsrvr user will be able to log on to the server at a time.
---------------------	--

I

ICA	See Independent Computing Architecture.
ICA Browser	The background process on a Citrix server that maintains information about other servers and published applications.
ICA Client	The software installed on a client device that enables end users to connect to Citrix servers.
ICA connection	The logical “port” used by an ICA Client to connect to, and start a session on, a Citrix server. An ICA connection is associated with a network connection or a serial connection (modems or direct cables).

ICA Gateway	A gateway between two network subnets that enables the Citrix servers on those subnets to exchange information.
ICA PassThrough	Allows non-Win32 ICA Clients to take advantage of the Citrix Program Neighborhood features. This is done by publishing the pre-installed ICA Client on the server and having clients “pass through” the server’s Citrix Program Neighborhood client while trying to access a server farm.
ICA protocol	The protocol used by ICA Clients and Citrix servers to exchange information.
ICA session	A lasting connection between an ICA Client and a Citrix server, identified by a specific user ID and ICA connection. It consists of the status of the connection, the server resources allocated to the user for the duration of the session, and any applications executing during the session.
Independent Computing Architecture (ICA)	The architecture that Citrix uses to separate an application’s logic from its user interface. With ICA, only the keystrokes, mouse clicks, and screen updates pass between the client and server on the network, while the application's logic executes on the server.
ICCCM	Inter-Client Communication Conventions Manual. ICCCM is a proposed X Consortium standard for inter-client communications.
initial program	An application that starts automatically when a session begins.
internal address	The private IP address of a Citrix server.
K	
ksh	See Korn shell.
Korn shell	A superset of the Bourne shell. It has many of the Bourne shell features, plus additional ones such as aliasing and history.

L

license	A software license installed on your Citrix server that does one of the following: - enables the base server software - enables additional server functionality - increases the user count (number of ICA connections) supported by the server.
license number	A number you use to activate your Citrix software licenses. The license number is unique to the individual server where you are activating the software. To get the license number, you enter the serial number (that appears on the serial number sticker on the back of the CD booklet) into the Citrix Licensing utility.
license pooling	A feature that enables you to combine the connection licenses of all the Citrix servers on the same network subnet into a common pool of licenses, which users consume as they connect to any server. All licenses are pooled by default. Assigning a license to a server removes it from the pool.
load balancing	An optional feature for adjusting the load on Citrix servers. When a user launches a published application that is configured for load balancing, that user's ICA session is established on the most lightly loaded server, based on criteria you can configure.

M

man page	A man page (literally a “manual page”) exists for each UNIX command. Each man page includes a description of the command, the syntax, warnings and important notes, and related commands.
master ICA Browser or master browser	The ICA Browser on one Citrix server in a network that gathers information about licenses, published applications, performance, and server load from the other member browsers within the network, and maintains that information.

master browser election	The process ICA Browsers go through to choose (elect) a master browser from among the Citrix servers on a given network. Browser elections occur when a new Citrix server is started, when the current master browser does not respond, or when two master browsers are detected by another server or an ICA Client.
MetaFrame security	The feature that controls user access to MetaFrame commands and sessions. Default security settings are applied at installation—these defaults can be changed using the <code>ctxsecurity</code> command.

P

PAM	Pluggable Authentication Modules. PAM allows you to select the authentication service you want to use, and plug-in and make available new authentication service modules without having to modify your applications.
PassThrough client	See ICA PassThrough.
permissions	In UNIX Operating Systems, permissions determine which users have read, write, and execute access to files and directories. Permissions are associated with each file and directory.
pkgadd	A tool on the Solaris platform for installing software.
published application	An application installed on a Citrix server that is configured for multiuser access by ICA Clients. To an ICA Client user, the published application appears similar to an application running locally on the client device. With Load Balancing Services, you can load balance a published application among several servers.

R

root	The name given to the UNIX administration account that has special permissions. Also known as the <i>super user</i> .
-------------	---

S

script	See Shell script.
seamless window	One of the settings ICA Client users can specify for a published application. If a published application runs in a seamless window, the user can take advantage of all the client platform's window management features, such as resizing, minimizing, dragging and dropping between remote and local applications, and so forth.
secured function	A MetaFrame function that the MetaFrame security tool "ctxsecurity" controls user access to.
security	See MetaFrame security.
serial number	The first number you use to activate your Citrix software licenses. The serial number appears on the serial number sticker on the back of the CD booklet. Enter the serial number in the Citrix Licensing tool to get a license number that is unique to the server where you are activating the software.
server-based computing	A model in which applications are deployed, managed, supported, and executed on a server.
server extension license	A type of Citrix license that increases the user count, or enables additional features such as load balancing.
session	See ICA session.
session ID	A unique identifier for the session on a specific server.
sh	See Bourne shell.
shadowing	A feature of Citrix servers that enables authorized users to remotely join or take control of another user's session for diagnosis, training, or technical support.
shell	The interface between the user and the kernel. The shell acts as a command processor that accepts, interprets and executes commands.
shell script	An executable file that contains a set of UNIX shell commands.

smit	The System Management Interface Tool on the AIX platform.
subnet	A subset of a network.
super user	See root.
swinstall	A tool on the HP-UX platform for installing and configuring software.

T

TCP/IP	Transmission Control Protocol/Internet Protocol. A suite of communication protocols that enables resources to be shared among PCs, hosts, and workstations.
Terminal Server	Microsoft Windows NT 4.0, Terminal Server Edition.

U

UDP	User Datagram Protocol. A transport protocol in the Internet suite of protocols. UDP, like TCP, uses IP for delivery; however, unlike TCP, UDP provides for exchange of datagrams without acknowledgments or guaranteed delivery.
user count	Some Citrix licenses come with a user count. This is the number of ICA Clients who can have a session running on the server simultaneously. User counts can be shared (pooled) by all servers on the same subnet.

W

window manager	A program that handles general window-management jobs, such as creating borders around an application's main windows and controlling how you move and resize windows.
workstation	A computer designed for running UNIX Operating Systems.

X

X

See X Window system.

XML Service

A Citrix application portal technology that lets you integrate and publish applications into a Web browser from any standard Web server.

xterm

A terminal emulator for the X Window system.

xv

A UNIX graphics application that allows users to select and cut and paste areas of a screen and save graphics in different formats.

X Window system

The X Window system is a UNIX GUI (graphical user interface). Note that X runs on various platforms, not just UNIX.

Index

A

- accented characters 30
- activation code 39
- active sessions, displaying 64
- adding licenses 38
- alternate address configuration 115
- anonymous users
 - accounts 45, 46, 102
 - adding and configuring settings 103
 - displaying settings 102
 - installation 19, 20
 - security considerations 46
 - template directory 60
- application publishing
 - about application publishing 44
 - and load balancing 100
 - copying from existing details 53
 - deleting applications 56
 - displaying details about 51
 - for explicit or anonymous use 45
 - on more than one server 55
 - on servers of different architecture 48
 - pre-configured for anonymous use 60
 - renaming a published application 54
 - restricting access to 56
 - with mouse mappings 58

B

- backing-up script files 19
- base licenses 36
- broken connections
 - setting timeout intervals 86
- browser. See ICA Browser

C

- CDE
 - changing the window manager 97
 - increasing concurrent sessions 118
 - publishing a desktop 46
- Citrix Licensing. See licensing
- Citrix Load Balancing Services. See load balancing
- Citrix on the World Wide Web 11
- Citrix server administrator
 - configuring the group 18
 - permissions 106
- Citrix XML Service 9, 114

- Client Administrator's Guides
 - locating 11
- client device licensing 37
- clipboard and shadowing 71
- clipboard mapping
 - enabling and disabling 82
 - graphics clipboard support 83
- command line
 - conventions 3
 - printing 75
 - publishing 48
- commands 123–148
 - configuring access to 26
 - ctx3bmouse 58–60, 124
 - ctxalt 115, 125
 - ctxanoncfg 102–104, 126
 - ctxappcfg 44, 127
 - ctxbrcfg 93–95, 129
 - ctxcapture 83, 130
 - ctxcfg 80–89, 116, 131
 - ctxconnect 68, 133
 - ctxdisconnect 68, 133
 - ctxgrab 83, 134
 - ctxlicense 38–42, 135
 - ctxlogoff 67, 136
 - ctxlpr 74–76, 137
 - ctxmaster 91, 138
 - ctxmsg 73, 138
 - ctxprinters 74, 139
 - ctxqserver 66, 140
 - ctxqsession 64–65, 142
 - ctxquser 64–65, 142
 - ctxreset 69, 143
 - ctxsecurity 106–112, 144
 - ctxshadow 70, 145
 - ctxshutdown 28, 147
 - ctxsrv 28, 94, 148
 - types of MetaFrame commands 26
- concurrent sessions, limiting 80
- configuring
 - an initial program 57
 - anonymous users 102
 - applications to print 76
 - ICA browsers 93
 - ICA gateways 95
 - network firewalls 114
 - scripts 96
 - security 110
 - servers 78
 - shadowing 84
 - TCP/IP port number 116

- timeout intervals 86
- CONSOLE setting, and ctxsecurity 106
- conventions
 - command line 3
 - in the documentation 2
- copying
 - published applications 53
 - server configuration to other servers 89
- ctx3bmouse 58–60, 124
- ctxadmn group 18, 106
- ctxalt 115, 125
- ctxanon group 104
- ctxanoncfg 102–104, 126
- ctxanoninit.sh 61
- ctxappcfg 44, 47–56, 127
- ctxbrcfg 93–95, 129
- ctxcapture 83, 130
- ctxcfg 56, 131
 - configuring shadowing 84
 - configuring TCP/IP port number 116
 - controlling logon settings 78
 - enabling / disabling clipboard 82
 - enabling / disabling printing 81
 - listing server configuration 88
 - setting the number of permitted connections 80
 - setting timeout intervals 86
- ctxconnect 68, 133
- ctxdisconnect 68, 133
- ctxgrab 83, 134
- ctxlicense 38–42, 135
- ctxlogoff 67, 136
- ctxlpr 74–76, 137
- ctxmaster 91, 138
- ctxmsg 73, 138
- ctxprinters 74, 139
- ctxqserver 66, 140
- ctxqsession 64–65, 142
- ctxquser 64–65, 142
- ctxreset 69, 143
- ctxsecurity 106–112, 144
- ctxsession.sh 96
- ctxshadow 70, 145
- ctxshutdown 28, 147
- ctxsrv 28, 94, 148
- ctxsrvr user
 - configuring access to commands 27
 - creating 18
- ctxwm, window manager 97
- ctxXtw.sh 96
- currency symbol support 17

current sessions, displaying 64

D

- dead keys, on non-English keyboards 31
- default
 - printer 74
 - security settings 109
- deleting
 - anonymous user accounts 103
 - published applications 56
- desktop, publishing 48
- disabling
 - client clipboard mapping 82
 - client printing 81
 - notification of shadowing 85
 - published applications 44, 52, 53
 - shadowing 84
- disconnected sessions
 - setting timeout intervals 86
- disconnecting a session 68
- displaying
 - client printers or printer ports 74
 - event logs 31
 - information about Citrix servers 66
 - master browser name 91
 - server configuration 88
 - user/session information 64
- documentation
 - Client Administrator's Guides 11
 - comments and suggestions 12
 - conventions 2
 - online 11
 - other sources 11

E

- elections
 - about elections 91
 - configuring server behaviour 93
 - manipulating 91
- enabling
 - client clipboard mapping 82
 - client printing 81
 - notification of shadowing 85
 - published applications 44, 52, 53
 - shadowing 84
- encryption
 - forcing clients to use 132
- environment variables 26, 27, 47, 61, 96
- Euro currency symbol support 17
- event logging 31
- explicit users 45

external addresses 115

F

file limits, increasing 118

firewalls 114

- and network address translation 115

font path 96

foreign keyboards 30

G

gateways

- configuring ICA gateways 95

generating

- server configuration details 88

grace period 36

graphics clipboard support 83

H

hotkey, to end shadowing 72

I

ICA Browser 90

- configuring 93

- locating 91

- refresh interval 94

- restarting and stopping 94

- UDP port 114

- with firewalls 114

- with network address translation 115

ICA Client

- documentation 11

- keyboards 30

- software 4

ICA gateways 95

ICAPORT (configure TCP/IP port number) 116

idle sessions

- setting for anonymous sessions 104

- setting timeout intervals 86

inheriting logon details 79

initial program configuration 57

installing MetaFrame 17

- on AIX 24

- unattended installation 25

- on HP-UX 22

- unattended installation 23

- on Solaris

- for the first time 18

- unattended install 21

- upgrading to a new release 19

- overview 17

- reinstalling 34

integrating with other Citrix servers 9

ISO 8859-15 17

J

Java applications 48

K

kernel tuning

- on AIX 121

- on HP-UX 120

- on Solaris 117

keyboards, non-English support 30

L

license serial number 38

licensing 35–42

- activating a license 38

- client device licensing 37

- displaying licenses 41

- installing an upgrade license 40

- introduction to 36

- pooling licenses 42

- removing licenses 42

limiting the number of connections 80

listing

- information about Citrix servers 66

- license information 41

- mapped printers 74

- published applications 51

- server configuration 88

- the default printer 74

- user/session information 64

load balancing

- a group of servers 100

- about 100

- reconnecting to load balanced sessions 70

- troubleshooting 101

- tuning the number of connections 101

locale, and non-English keyboards 30

logging events 31

logging off a session 67

Login screen, customizing 96

logon settings 78

M

man pages

- displaying 27

- installing 19

- on AIX 25

- on HP-UX 22

- on Solaris 19

MANPATH environment variable 27, 96

mapping

- client clipboard 82
- client printers 81
- master browser 90–91
 - ctxmaster 91
 - elections 91
 - locating 91
 - refresh interval 94
- messages
 - during server shut down 29
 - sending to users 73
- MetaFrame
 - getting more information on 11
- mouse configuration 58–60
- multiple servers
 - installing MetaFrame on 21
 - propagating server configuration to 89

N

- network address translation
 - and ICA browsing 115
- network firewalls 114
- non-standard shells 113
- notification, of shadowing 85

O

- online documentation 11
- operating system
 - configuring for a large number of users 117, 120, 121
 - requirements
 - on AIX 17
 - on HP-UX 16
 - on Solaris 16
- optimizing MetaFrame
 - configuring for a large number of users 117, 120, 121
 - publishing applications 44
 - setting a blank screensaver 89

P

- PAM. See Pluggable Authentication Modules
- pass-through ICA client 9
- password
 - and explicit / anonymous users 45
 - forcing a user to enter 78
- pasting
 - enabling clipboard mapping 82
- patches 17
- PATH environment variable 26
- permissions 18, 106
- Pluggable Authentication Modules (PAM) 7

- pooling licenses 42
- port number
 - configuring 116
- printer mapping 81
- printing 74–76
 - configuring applications to print 76
 - enabling / disabling for users 81
 - from applications 76
 - from the command line 75
 - troubleshooting 76

- propagating
 - configuration between servers 89
- pseudo-terminals 117, 121
- ptys. See pseudo-terminals
- publishing a UNIX command line 48
- publishing applications
 - about 44
 - and load balancing 100
 - copying from existing details 53
 - deleting applications 56
 - displaying details about 51
 - enabling and disabling 44, 52, 53
 - for explicit or anonymous use 45
 - on more than one server 55
 - on servers of different architecture 48
 - pre-configured for anonymous use 60
 - renaming 54
 - restricting access to 56
 - using ctxappcfg 127
 - with mouse mappings 58
- publishing desktops 46
- publishing shell scripts 46

Q

- quiet installation
 - on AIX 25
 - on HP-UX 23
 - on Solaris 21
- quiet mode, during shut down 29

R

- reconnecting, to load balanced sessions 70
- refresh interval
 - for ICA Browser service 94
- reinstalling MetaFrame 34
- remote configuration of servers 89
- removing
 - licenses 42
 - MetaFrame 32
 - published applications 56
- renaming

- anonymous user accounts 104
 - published applications 54
- replicating
 - server configuration to other servers 89
- resetting
 - a session 69
- restarting
 - the ICA Browser 94
- root access 18, 106
- rsh command 89

S

- S99ctxsrv 19
- screensaver settings 89
- script files
 - backing-up 19
 - ctxanoninit.sh 61
 - ctxsession.sh 96
 - ctxXtw.sh 96
 - customizing 96
 - for installation 21
 - publishing 46
 - S99ctxsrv 19
 - server configuration 88
- security 105–112
 - and anonymous users 46
 - configuring 110
 - displaying settings 110
 - examples 112
 - removing settings 111
 - X security policy 96
- sending messages to users 73
- server extension licenses 36
- servers
 - configuring 78–88
 - displaying information about 66
 - integrating with other Citrix servers 9
- session name/id
 - displaying 64
- sessions
 - configuring session shadowing 84
 - connecting to disconnected sessions 68
 - disconnecting 68
 - displaying information about 64
 - ending a session 67
 - resetting 69
 - setting the number of 80
 - setting timeout intervals 86
 - shadowing 70
- shadowing
 - and the clipboard 71

- configuring a hotkey to end 72
 - configuring session shadowing 84
 - shadowing a user's session 70
 - stopping shadowing 72
- shell scripts
 - backing-up 19
 - ctxanoninit.sh 61
 - ctxsession.sh 96
 - ctxXtw.sh 96
 - customizing 96
 - for installation 21
 - publishing 46
 - S99ctxsrv 19
 - server configuration 88
- shells, non-standard 113
- shutting down MetaFrame 28
- silent installation
 - on AIX 25
 - on HP-UX 23
 - on Solaris 21
- starting
 - MetaFrame 28
 - the ICA Browser 94
- stopping
 - MetaFrame 28
 - shadowing 72
 - the ICA Browser 94
- subnet
 - configuring ICA gateways 95
- syslog.conf 31
- system requirements 14

T

- TCP/IP port number, configuring 116
- template directory 60
- timeout settings 86, 104
- troubleshooting
 - errors and warnings 31
 - load balancing 101
 - non-English keyboards 31
 - non-standard shells 113
 - operating system requirements 16
 - printing 76

U

- UDP port and ICA Browser 114
- unattended installation
 - on AIX 25
 - on HP-UX 23
 - on Solaris 21
- uninstalling MetaFrame 32

- UNIX command line, publishing 48
- unix2dos 76
- upgrade licenses 36, 40, 42
- upgrading to a new release 19
- user
 - anonymous users 45
 - counts 37
 - logon settings 78
 - permissions 18, 26, 106

V

- viewing
 - available client printers 74
 - default printer 74

- event logs 31
- master browser name 91
- published applications 51
- server configuration 88
- user/session information 64

W

- window manager
 - customizing 97–98
- World Wide Web, finding Citrix on 11

X

- X security policy 96
- XML Service 9