License Use Management

IBM

Using License Use Management Runtime for Windows®

Version 4.5.5 (September 6, 1999)

SH19-4347-02

License Use Management

IBM

Using License Use Management Runtime for Windows®

Version 4.5.5 (September 6, 1999)

SH19-4347-02

Note

Before using this information and the product it supports, be sure to read the general information under “Notices” on page xiii.

ISO 9001 Certification

This product was developed using an ISO 9001 certified quality system.

Certification has been awarded by the Italian quality system certification group, CSQ (Certification No. CISQ/CSQ 9150.IBM7).

CSQ is a member of the mutually recognized organization of European assessors, ITQS, which assesses and certifies quality systems in the field of information technology enterprises.

Third Edition (September 1999)

This major revision obsoletes and replaces SH19-4347-01. The major changes are described in “Summary of Changes” on page xix. In the hard copy version of this book, technical changes are marked by a vertical line in the left margin. In the .HTM version, technical changes appear in purple. Post-publication technical changes are marked in brown.

This edition applies to Version 4.5.5 of IBM License Use Management Runtime for Windows, a part of License Use Management for Windows, Program Number 5697-B23, and to Version 4.5.5 of IBM License Use Management Runtime for Windows NT Alpha, a part of License Use Management for Windows NT, Program Number 5697-E77, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

IBM welcomes your comments.

A form for reader's comments is provided at the back of this publication. If the form has been removed, address your comments to:

License Use Management Information Development
Rome Tivoli Lab
IBM Italia S.p.A.
Via Sciangai, 53
00144 Rome
Italy
Fax Number : (+39) 06 5966 2077
Internet ID: ROMERCF at VNET.IBM.COM

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright 1994, 1997 Isogon Corp.**

© **Copyright International Business Machines Corporation 1995, 1999. All rights reserved.**
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Notices xiii
Trademarks xiv

About This Book xv
Who Should Use This Book xv
How This Book is Organized xv
Where to Find More Information xvi
Online Books xvi
License Use Runtime README File xvii
IBM LUM Web Site xvii
Notational Conventions xviii
Notation for Drives and Directories xviii

Summary of Changes xix
Changes in the Third Edition xix
General Changes xix
Specific Changes xix

Chapter 1. Introduction to License Use Runtime 1
License Use Management at a Glance 1
License Use Management Vendor Perspective 1
License Use Management Customer Perspective 2
Basic Concepts of License Use Management 2
License Use Runtime Platforms 4
License Passwords 5
Password Use Control Levels 6
Vendor-Managed Use Products 6
Customer-Managed Use Products 7
License Types 7
Nodelocked Licenses 7
Simple Nodelocked Licenses 8
Concurrent Nodelocked Licenses 8
Use-Once Nodelocked Licenses 8
Per-Server Licenses 9
Network Licenses 9
Concurrent Licenses 10
Reservable Licenses 10
Use-Once Licenses 11
Per-Seat Licenses 11
License Policies 12
Vendor-Controlled Policies 12
Try-and-Buy Policy 12
Multiuse Rules 12
Product Wait Queues 12
License Annotation 13

© Copyright IBM Corp. 1995, 1999

Custom Configuration . . . 13
Customer-Controlled Policies . . . 13
Hard Stop/Soft Stop Policy . . . 13
User Access Restriction . . . 14
Switching from Per-Server to Per-Seat Licenses . . . 14
License-Enabling Models . . . 15
Scalable Installation and Configuration . . . 16
License Creation Tool . . . 17
License Administration Tool . . . 18
High-Availability Licensing . . . 20
Backup Procedure . . . 21
Working with Licensed Products . . . 21
Central Registry License Server . . . 21
Working with Nodelocked Licenses (Non-Runtime-Based Enabling) . . . 22
Working with Nodelocked Licenses (Runtime-Based Enabling) . . . 23
Working with Use-Once Licenses . . . 24
Working with Concurrent Licenses . . . 26
Working with Reservable Licenses . . . 28
Working with Per-Server Licenses . . . 31
Working with Per-Seat Licenses . . . 32

Chapter 2. Planning Your Network Licensing Environment . . . 35
Selecting Your Servers . . . 35
Network Computing System (NCS) . . . 36
Planning Direct Binding . . . 37
Planning the Central Registry . . . 40
Planning for Java Applications and Applets . . . 40
Planning Clusters . . . 41
Restrictions on Cluster Size and Composition . . . 41
Examples of Cluster Size Rules . . . 43
Cluster Membership Considerations . . . 45
Running the NetBIOS Socket Server . . . 45
Verifying Network Connections . . . 46

Chapter 3. Installing License Use Runtime . . . 47
Before You Install . . . 47
Selecting License Use Runtime Components . . . 48
Hardware Requirements . . . 49
Software Requirements . . . 49
Protocol Configuration Requirements . . . 50
Obtaining the License Use Runtime Code . . . 50
Deciding How to Install . . . 51
Installing License Use Runtime Interactively . . . 51
Installing after an Uninstallation . . . 57
Installing License Use Runtime Noninteractively . . . 58
Using the Installation Command . . . 59
Creating a Response File . . . 60
Editing the Sample Response File . . . 60

Saving a Response File During an Interactive Installation 61
Verifying and Updating the PATH Environment Variable 62
Installation Using a Code Server 62
Uninstalling License Use Runtime 63
Installing LUM Java Client Support 63
Before You Install 63
Disk Space Requirements 63
Software Requirements 63
Obtaining LUM Java Client Support Code 64
Installing LUM Java Client Support 64
Uninstalling LUM Java Client Support 64

Chapter 4. Getting Started with License Use Runtime 67
Setting Up Your Servers and Clients 67
Configuring to Handle Nodelocked Licenses 67
Configuring to Handle Network Licenses 67
Determining the Configuration Required 68
Before You Configure 70
Customizing Log Information 70
Automatically Starting License Servers 71
Disabling Remote Administration 71
Configuring Direct Binding 71
Using the Configuration Tool GUI 72
Using the Configuration Tool Command-Line Interface 73
Scenario 1: Configuring a Standalone Nodelocked License Server 73
Command-Line Equivalent 75
Scenario 2: Configuring a Nodelocked License Server in a Network 75
Command-Line Equivalent 79
Scenario 3: Configuring a Network License Server 79
Command-Line Equivalent 84
Scenario 4: Configuring a Network License Client 84
Command-Line Equivalent 87
Scenario 5: Configuring the Central Registry License Server 88
Command-Line Equivalent 90
Starting and Listing Your Processes 90
License Use Runtime Processes Defined as Windows NT Services 90
Managing License Use Runtime Processes with Service Manager Tool 91
Verifying Connections to Servers 92
Administering License Use 92
Using the Basic License Tool GUI 92
Starting the Basic License Tool GUI 92
Refreshing License Information 93
Selecting Servers 93
Using the Basic License Tool Command-Line Interface 94
Performing Basic Administration 94
Scenario 6: Managing a Licensed Product 94
Enrolling the Product 94
Distributing the Licenses 97

Generating Reports 99
Monitoring the Number of Product Users 100
Command-Line Equivalent 101
Scenario 7: Managing Reservable Licenses 102
Command-Line Equivalent 105
Exercising Customer-Controlled Policies 106
Scenario 8: Switching from Per-Server to Per-Seat Licenses 106
Command-Line Equivalent 109
Scenario 9: Using the Hard Stop/Soft Stop Policy 109
Command-Line Equivalent 112
Scenario 10: Restricting User Access 113
Administering High-Availability Licensing 114
Scenario 11: Creating and Administering a Cluster 114
Creating a Cluster 114
Activating Cluster Members 117
Adding a Cluster Member 119
Deactivating a Server 120
Viewing Licenses Being Served 121
Enrolling and Removing Licenses on a Cluster 122
Command-Line Equivalent 122
Upgrading a Custom Configuration 123
Command-Line Equivalent 126

Chapter 5. License Use Runtime Commands 127
i4blt - Basic License Tool 128
General Rules for the i4blt Command 128
Primary Command Options 129
-a Enroll a Product 130
-U Update a Product 132
-E Extract and Distribute Licenses 134
-d Delete a Product License 135
-R Reserve Licenses; Delete or Update Reserved Licenses 137
-C Clean Up Stale Licenses 138
-l Display a List 139
-s Display Product License Status 144
-r Generate a Report 145
-x Delete Server Log Entries 148
-m Monitor and Log Threshold Events 149
-H Administer High-Availability Licensing 150
-h Display Help 152
i4cfg - Configuration Tool 153
License Use Runtime Tools 157
i4tv - Test Verification Tool 157
i4target - Target View Tool 158
License Use Runtime Processes 160
i4ls - Network License Server Process 160
i4ls -b - Nodelocked License Server Process 161
i4ls -a - Central Registry License Server Process 163

i4nbssvr - NetBIOS Socket Server Process (Windows NT only) 164
i4lct - License Creation Tool 164
Defining Rules for Multiple-Use Concurrent Licenses 174
i4nat - Nodelocked Administration Tool (for Backward Compatibility Only) 175
-a Add a Nodelocked License 176
-d Delete a Nodelocked License 177
-l Display License Information 178
-u Update Concurrent Nodelocked License Information 178
-h Display Command Line Interface Usage 179

Chapter 6. Hints and Tips 181
Managing Time Zone 181
Using the Built-In Backup and Recovery Procedure 181
Causes for Corrupted Definition or Database Files 181
Automatic Backup Procedure 182
Recovery Procedure 182
Manual Backup 183
Manual Recovery 183
Managing the Reports Log Files 183
Managing a Custom Configuration 184
Before Requesting a License Upgrade 184
Deleting Products or Reducing Numbers 184
Deleting Keys 184

Chapter 7. Troubleshooting 185
Checking License Details 185
Troubleshooting Licenses (All Types) 187
Troubleshooting Nodelocked Licenses 187
Troubleshooting Network Licenses (All Types) 188
Troubleshooting Reservable and Reserved Licenses 189
Troubleshooting Per-Server and Per-Seat Licenses 189
Troubleshooting Licenses of Customer-Managed Use Products 189
Troubleshooting Licenses of Vendor-Managed Use Products 190
Troubleshooting Performance Problems 190
Basic License Tool Performance 190
Performance in a Direct Binding Environment 190
Troubleshooting Heavy Server Workloads 190
Troubleshooting License Use Runtime Processes 190
Starting Required Processes 190
Restart and Recovery 191
Troubleshooting Custom Configuration Licenses 192
Cannot Install a Custom Configuration License 192
Troubleshooting Network Connections 192
Troubleshooting Direct Binding 192
Troubleshooting Communication Protocols 192
NetBIOS 193
TCP/IP 193
IPX on Windows 98 194

Troubleshooting Coexistence with Other Products 194
Coexistence with Digital DCE for Windows NT 194
Troubleshooting the Hardware 194
Collecting Error Log Data 195
Running Processes in Traced Mode 195
Running Enabled Applications in Traced Mode 196
Running Tools in Traced Mode 196
Collecting Other Data 196
Troubleshooting LUM Java Client Support 197
Java Program Cannot Read the User Name 197
Incomplete View of an Applet 197
Troubleshooting Java Client Support Uninstallation 198

Appendix A. License Use Runtime Configuration File 199

Appendix B. Using the Nodelock File 209

Appendix C. Features and Functions Added in Version 4 211

Glossary 213

Index 219

Figures

1. Licensing Concepts Summary 6
2. Using a Nodelocked License (Non-Runtime-Based Enabling) 22
3. Using a Nodelocked License (Runtime-Based Enabling) 23
4. Using a Use-Once License for a C-Language Program 24
5. Using a Use-Once License for a Java Application or Applet 25
6. Using a Concurrent License for a C-Language Program 26
7. Using a Concurrent License for a Java Application or Applet 27
8. Using a Reservable License for a C-Language Program 28
9. Using a Reservable License for a Java Application or Applet 29
10. Using a Per-Server License 31
11. Using a Per-Seat License for a C-Language Program 32
12. Using a Per-Seat License for a Java Application or Applet 33
13. Direct Binding with Network License Servers and Nodelocked License Servers 38
14. Direct Binding with Java Client Support 39
15. Welcome Window 52
16. READ.ME Window 53
17. Choose Destination Location Window 54
18. Select Components Window 55
19. Setup Window 55
20. Setup Complete Window 56
21. Setup Complete Window 57
22. Configuration Tool Notebook - Standalone Nodelocked License Server . . 74
23. Configuration Tool Notebook - Log Page 75
24. Configuration Tool Notebook - Nodelocked License Server in a Network . 76
25. Configuration Tool Notebook - User Section 77
26. Configuration Tool Notebook - Direct Binding Section 78
27. Configuration Tool Notebook - Network License Server 80
28. Configuration Tool Notebook - Log Page 81
29. Configuration Tool Notebook - Direct Binding Section 82
30. Configuration Tool Notebook - User Section 83
31. Configuration Tool Notebook - Network License Client 85
32. Configuration Tool Notebook - Direct Binding Section 86
33. Configuration Tool Notebook - User Section 87
34. Configuration Tool Notebook - Central Registry License Server 88
35. Configuration Tool Notebook - Direct Binding Section 89
36. Service Manager Tool Window 91
37. Basic License Tool Window 93
38. Import Window 95
39. Enroll Product Window 95
40. Enroll Licenses Window 96
41. Basic License Tool Window with SMARTJava Enrolled 97
42. Distribute Licenses Window 97
43. Set Number of Licenses Window 98
44. Distribute Licenses Window with Number of Licenses Set 98

© Copyright IBM Corp. 1995, 1999

45. Basic License Tool Window with Distributed Licenses 99
46. Reports Window 99
47. Concurrent Users Page 101
48. Basic License Tool Window with Reservable Licenses 102
49. Details Notebook with Reservable Tab 103
50. Reserving Reservable Licenses 103
51. Details Notebook with Reserved Licenses 104
52. Reserved Users Page with Reservable Licenses in Use 104
53. Un-Reserved Users Page 105
54. Reservable Page with Unreserved Licenses in Use 105
55. Basic License Tool Window with Per-Server Licenses Enrolled 107
56. Enabling Per-Seat Licensing 108
57. Basic License Tool Window with Per-Seat Licenses Enrolled and Enabled 108
58. Basic License Tool Window with Soft-Stop Licenses in Use 110
59. Update Licenses Window with High-Water Mark 110
60. Update Licenses Window - Enrolling More Licenses 111
61. Resetting the High-Water Mark 111
62. Basic License Tool Window with Licenses Updated 112
63. Clusters Window 115
64. Create Cluster Window 115
65. Define Cluster Members Window 116
66. Clusters Window with New Cluster Added 116
67. Details of New Cluster 117
68. Details of Cluster after Three Activations 118
69. Clusters Window with Cluster ID 118
70. Basic License Tool Window with HAL Test Product 119
71. Add Cluster Members Window 120
72. Details of Cluster after Adding a Server 120
73. Details of Cluster after Deactivation 121
74. Details of Servers Serving HAL Test Product 122
75. Initial State of the Concurrent Page of the Details Notebook for a Custom Configuration License 123
76. Enroll Product Window for Custom Configuration 124
77. Import Window for Custom Configuration 125
78. Enroll Product Window for Custom Configuration 125
79. Upgraded State of the Concurrent Page of the Details Notebook for a Custom Configuration License 126

Tables

1. License Use Runtime Platforms 4
2. License-Enabling Models, License Types, and License Policies 16
3. Number of Servers in a Cluster 43
4. Example - Cluster with Three Initial Members 44
5. Example - Cluster with Six Initial Members 45
6. Required Components by Machine Role 48
7. Hardware Requirements 49
8. Disk Space Requirements 49
9. Transport Protocols 50
10. Configuration Required to Support All Types of Licenses 69
11. Configuration Options 70
12. Valid Uses of i4lct 166
13. License Use Runtime Processes 191
14. Features and Functions Added in Version 4 211

© Copyright IBM Corp. 1995, 1999

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
P.O. Box 12195
3039 Cornwallis
Research Triangle Park, NC 27709-2195
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

© Copyright IBM Corp. 1995, 1999

Trademarks

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX
IBM
OS/2

LicensePower and iFOR are registered trademarks of Isogon Corp.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

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

IRIX is a trademark of Silicon Graphics, Inc.

Solaris is a registered trademark of Sun Microsystems.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks, and Authenticode a trademark, of Microsoft Corporation in the U.S. and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Other company, product, and service names may be trademarks or service marks of others.

About This Book

This book provides a guide to setting up the environment required to support licensed software products. Refer to Chapter 1, “Introduction to License Use Runtime” on page 1 for details on licensed products. This book describes License Use Management Runtime for Windows (referred to as License Use Runtime). It contains information about how to install and configure the servers where licenses will be installed and the clients that will use the products, and how to set up a network licensing environment. It explains how to manage the licenses for licensed software products you have installed.

Before reading this guide, follow the procedures described in the documentation accompanying the licensed software product you acquired.

Who Should Use This Book

This book is intended for:

The system administrator who is responsible for setting up and administering the license management environment using License Use Runtime.

The License Use Runtime system administrator must have general knowledge of:

- Windows
- The NetBIOS, TCP/IP, or IPX transport protocol
- The network environment

The end users who run the software products on their client machines and, if required, install and configure License Use Runtime with the support of the system administrator.

End users require only general knowledge of the Windows operating environment.

How This Book is Organized

This book is divided into the following sections:

Chapter 1, “Introduction to License Use Runtime,” provides an overview of License Use Runtime features and benefits, describes supported license types, license policies, and license-enabling models, and presents some simple scenarios of the use of licensed products.

Chapter 2, “Planning Your Network Licensing Environment,” provides information on how to organize your network.

Chapter 3, “Installing License Use Runtime,” gives you instructions on how to install and uninstall License Use Runtime components.

© Copyright IBM Corp. 1995, 1999

About This Book

Chapter 4, “Getting Started with License Use Runtime,” provides scenarios for configuring License Use Runtime, administering product licenses, and exercising customer-controlled policies, using the License Use Runtime tools.

Chapter 5, “License Use Runtime Commands,” documents the License Use Runtime command line interface.

Chapter 6, “Hints and Tips,” provides hints and tips to better take advantage of License Use Runtime, and information on how to use the provided backup and recovery procedure.

Chapter 7, “Troubleshooting,” helps you to improve performance and to handle problems, should they arise when you use license-enabled products.

Appendix A, “License Use Runtime Configuration File,” provides reference information on the configuration file.

Appendix B, “Using the Nodelock File,” explains how to edit a nodelock file.

Appendix C, “Features and Functions Added in Version 4” lists the features and functions that have been added in Version 4.

“Glossary” defines terms used in this manual.

Where to Find More Information

This section lists other sources of information related to License Use Runtime.

Online Books

The online books for License Use Runtime, listed in the rest of this section, are available in .HTM format for viewing with a Web browser.

The .HTM files are in the directory:

\ifor\win\bin*language*

where *language* is the subdirectory for a language in which License Use Runtime has been installed on your machine. The directories for the languages are:

Brazilian Portuguese	PT_BR	Danish	DA_DK
English	EN_US	Finnish	FI_FI
French	FR_FR	German	DE_DE
Italian	IT_IT	Japanese	JA_JP
Korean	KO_KR	Norwegian	NO_NO
Simplified Chinese	ZH_CN	Spanish	ES_ES
Swedish	SV_SE	Traditional Chinese	ZH_TW

To view an .HTM file, open the file in your Web browser.

Command Reference

The *Command Reference,* in addition to being part of this book (Chapter 5, “License Use Runtime Commands” on page 127), is available separately as an online book. The file is named:

lumcmd

Message Reference

The *Message Reference* is an online book named:

lummsg

It lists Basic License Tool and Configuration Tool error messages, with an explanation, system action, and suggested user action.

Using License Use Runtime

This book is available as an online book, as well as in hard copy. The file is named:

lumusg

A printable copy of this book in PostScript and PDF formats is available for download from the IBM LUM Web site http://www.software.ibm.com/is/lum.

Using Application Developer's Toolkit

For information about how to license-enable software products for use with License Use Management, see *Using License Use Management Application Developer's Toolkit*, SH19-4362. If Application Developer's Toolkit is installed in your environment, it is available as an online book in .HTM format. The file is named:

lumtkt

License Use Runtime README File

For changes to License Use Runtime or to this book that were made after the book went to press, see the README.ARK file on the CD-ROM or in the download package from the LUM Web site.

IBM LUM Web Site

Visit the IBM License Use Management Web site at http://www.software.ibm.com/is/lum for information and news about IBM License Use Management, and to download License Use Runtime publications and code.

About This Book

Notational Conventions

This book uses the following notation in text:

Bold	Bold print indicates something you click on, select, or type, such as a menu option, field, or push button.
Italics	Italic print is used for variables for which you must supply a value, for introducing new terms in the text, and for emphasis.
Monospacing	Monospacing indicates system messages and examples.

This icon marks important information that can affect the operation of the product or the completion of a task.

Notation for Drives and Directories

When this book shows the path to a file in the \ifor directory, it means the \ifor directory on the drive where you installed License Use Runtime.

For example, if you installed License Use Runtime on the D: drive, when this book refers to:

\ifor\ls\conf\i4ls.log

it means:

D:\ifor\ls\conf\i4ls.log

Summary of Changes

This section provides an overview of major changes made to this book.

Changes in the Third Edition

The third edition of this book incorporates changes documented in the README.ARK and README.JCS files for Version 4.5.1 and Version 4.5.2, and changes made to Version 4.5.5 of the product.

General Changes

This section summarizes the general, pervasive changes made to this book.

Viewing Books Online

Books are no longer supplied in INF format. View the HTML-format versions of books in your web browser.

Additional Operating Systems

Support has been added for the following operating systems:

- Windows NT 4.0 Server, Terminal Server Edition (Windows Terminal Server), on the x86 platform. The behavior of License Use Runtime on Windows Terminal Server is the same as on Windows NT (x86).
- Windows NT 4.0 on the Alpha platform. On this platform, License Use Runtime functions are available only from the command line.
- Windows NT 4.0 Server, Terminal Server Edition, on the Alpha platform. On this platform, License Use Runtime functions are available only from the command line.

For information, see “License Use Runtime Platforms” on page 4 and Chapter 3, “Installing License Use Runtime” on page 47.

New Versions and Releases of Operating Systems

Support for previously supported operating systems has been extended to include the following new versions and releases:

HP-UX 11.0 32-bit
SGI IRIX 6.5
Sun Solaris 2.7

For information, see “License Use Runtime Platforms” on page 4 and Chapter 3, “Installing License Use Runtime” on page 47.

Specific Changes

This section summarizes the changes made to this book to reflect new and changed function and support.

© Copyright IBM Corp. 1995, 1999

Summary of Changes

Custom Configuration

Vendors can now offer combinations of products, tailored to the needs of each user, under a single custom configuration license.

A section has been added to Chapter 3, Installing License Use Runtime that introduces the concept of custom configuration.

A section has been added to Chapter 6, Hints and Tips that suggests how you might better manage a custom configuration.

A section has been added to Chapter 7, Troubleshooting that suggests what you might do should you have a problem with a custom configuration.

Installation on Windows NT

Installation and uninstallation of License Use Runtime on Windows NT (on the x86 and Alpha platforms) use InstallShield 5.1. See Chapter 3, “Installing License Use Runtime” on page 47.

On Windows NT and Windows Terminal Server, after installing License Use Runtime, you need not shut down and restart your machine. See “Installing License Use Runtime Interactively” on page 51.

Uninstallation Enhancements on Windows NT

Uninstallation of License Use Runtime on Windows NT has been enhanced to remove definitions of system and user variables and NT LUM services. The I4_INSTALL_DRIVE and I4_LANG variable are removed. LUM entries in the BOOKSHELF, HELP, IPF_PATH32, NLSPATH, and PATH variables are removed. See Chapter 3, “Installing License Use Runtime” on page 47.

License Creation Tool Enhancement

The License Creation Tool enables you to create licenses whose start date is one day earlier than the date the tool is run. This makes it possible for licenses to be used immediately in any time zone. See “i4lct - License Creation Tool” on page 164 for details.

Editing a Nodelock File

A new appendix, Appendix B, “Using the Nodelock File” on page 209, explains how to edit and use a nodelock file.

List of Features and Functions Added in Version 4

A new appendix, Appendix C, “Features and Functions Added in Version 4” on page 211, lists the features and functions added in Version 4. Do not use obsolete commands or APIs, which are supported for backward compatibility, with these newer features and functions.

Chapter 1. Introduction to License Use Runtime

License Use Runtime is part of IBM License Use Management, a combination of tools for software asset protection. The License Use Management tools enable software vendors and their customers to ensure that customers comply with the terms and conditions of license agreements. They check compliance through runtime monitoring of the usage of software assets.

License Use Management at a Glance

License Use Management consists of two products:

The License Use Management Application Developer's Toolkit contains the tools that are needed to implement licensing technology in an application program (called *license-enabling* the application). To do the enablement, vendors code API calls in their products and embed the code that services the API calls. The products thus become *license-enabled*. Vendors can license-enable C-language programs, Java applications, and Java applets.

The Application Developer's Toolkit offers the vendor great flexibility in:

- Level of control exercised by the enabled application
- Type of customer licensing environment for which the application is intended
- Implementation of various policies

The Application Developer's Toolkit is a priced product of IBM. The software vendor who acquires the kit receives a copy of the License Use Management software, and gets royalty-free rights to redistribute License Use Runtime within the license-enabled application.

License Use Management Runtime (License Use Runtime) contains the tools that are needed in an end user environment to manage licenses and to get up-to-date information about license usage. The License Use Runtime software is free of charge and is available for download from the IBM License Use Management (LUM) Web site http://www.software.ibm.com/is/lum.

License Use Management Vendor Perspective

License Use Management benefits software vendors by enabling them to:

Ensure that customers use software licenses within entitled limits

Base product prices on actual usage

Protect intellectual property from unauthorized use

Increase overall revenue as customers acquire all the licenses they need

Distribute software for a trial period with trial licenses that can be replaced by production licenses, thus minimizing distribution cost

© Copyright IBM Corp. 1995, 1999

License Use Management at a Glance

License Use Management Customer Perspective

License Use Management benefits the customers of software vendors by enabling them to:

- Ensure that they have enough licenses to satisfy their business requirements and, at the same time, that they are not paying for more licenses than they need
- Base software charges within the enterprise on actual usage
- Demonstrate license use compliance to internal and external auditors
- Protect organizations from inadvertent violations of license agreements
- Change software assets to alternative pricing policies that the vendor offers

Basic Concepts of License Use Management

A *license*, in the context of License Use Management, is permission to use an instance of a licensed software product or service, according to the basis on which the vendor charges for the product or service. The objective of License Use Runtime is to control the use of licenses in a customer's environment.

(Note that the term *license* does not refer to the license agreement that governs use of and rights to a product.)

In the license-enabling process, the vendor can:

- Select among various types of licenses (see “License Types” on page 7).
- Decide whether to distribute licenses one-by-one or in packages of multiple licenses from which individual licenses can be extracted (see “License Passwords” on page 5).
- Implement direct controls over the use of licenses, or make it possible for the customer to control use of licenses (see “Password Use Control Levels” on page 6).
- Impose, or allow the user to impose, various types of control over administration of licenses (see “License Policies” on page 12).

Vendors deliver licenses to customers in the form of a *license password*. The password contains an encryption of some terms of the acquisition of the software product. For example, the password may specify:

- How many licenses or concurrent copies of the product the customer can use
- The expiration date of the licenses
- The type of license

License Use Runtime checks that customers have a license that authorizes them to use the product when the product is executed, not when it is installed.

Depending on the terms for software product acquisition, vendors can implement licenses in two fundamental ways: *nodelocked* licenses and *network* licenses.

License Use Management at a Glance

A nodelocked license is stored on the workstation where the license-enabled product is installed, for the exclusive use of that node.

With network licenses, you set up a client/server configuration for License Use Management. Many License Use Runtime clients can share the licenses for enabled products. The licenses are stored on one or more *network license servers*. Each client workstation must be connected to a server. When the user at a client starts a licensed program, License Use Runtime at the license server determines whether a license is available.

License-enabled Java applications and applets must have network licenses. A Web server machine, rather than the end user machine where the application or applet runs, serves as the network license client. See “Planning for Java Applications and Applets” on page 40.

License Use Runtime includes an administration tool, called the *Basic License Tool*, which manages both nodelocked and network licenses on all the license servers in your network. The Basic License Tool enables you to:

- Add licenses to or delete licenses from the server database
- Display information about the licenses installed
- Distribute the licenses among the license servers available on the network
- Reserve licenses for the exclusive use of certain users
- Generate reports on license usage and server events

The Basic License Tool has a graphical user interface (GUI) and a command-line interface. For more information about what the Basic License Tool does, see “License Administration Tool” on page 18.

License Use Runtime Platforms

License Use Runtime Platforms

Table 1 shows which platforms License Use Runtime supports, and how to get the License Use Runtime code:

Table 1. License Use Runtime Platforms

AIX 4.3.3	License Use Runtime 4.5.5 base code is part of the base operating system, and is installed on every machine when the operating system is installed. Optional packages and filesets can be installed from the AIX installation media. Alternatively, you can install Version 4.5.5 from the License Use Management Version 4.5.5 CD-ROM or from the product package downloaded from the IBM LUM Web site http://www.software.ibm.com/is/lum.
AIX 4.3.2 **AIX 4.3.1** **AIX 4.3.0**	On AIX 4.3.2, License Use Runtime 4.5.0 base code is part of the base operating system, and is installed on every machine when the operating system is installed. On AIX 4.3.1, License Use Runtime 4.0.1 base code is part of the base operating system, and is installed on every machine when the operating system is installed. On AIX 4.3.0, License Use Runtime 4.0 base code is part of the base operating system, and is installed on every machine when the operating system is installed. Optional packages and filesets can be installed from the AIX installation media. To upgrade to License Use Runtime Version 4.5.5 without upgrading to AIX 4.3.3, download the code from the IBM LUM Web site http://www.software.ibm.com/is/lum.
AIX 4.1 **AIX 4.2**	The iFOR/LS license management product is part of the base operating system in AIX 4.1 and 4.2, and is installed on every machine when the operating system is installed. To upgrade to License Use Runtime Version 4.5.5, download the code from the IBM LUM Web site http://www.software.ibm.com/is/lum.
Windows NT® 4.0 (x86) **Windows NT 4.0 Alpha** **Windows NT Server 4.0, Terminal Server Edition (x86)** **Windows NT Server 4.0, Terminal Server Edition Alpha** **Windows® 98** **Windows 95** **OS/2 Warp Version 4** **Sun Solaris 2.6 and 2.7** **HP-UX 10.20 and 11.0** **Silicon Graphics IRIX 6.3, 6.4, and 6.5**	License Use Runtime Version 4.5.5 is normally incorporated into the license-enabled product. Otherwise, it is available for download from the IBM LUM Web site http://www.software.ibm.com/is/lum.

On AIX and OS/2, License Use Management Version 4.0 replaced previous License Use Management releases. On Windows, HP-UX, IRIX, and Solaris, Version 4.0 was the first IBM License Use Management release.

License Passwords

A license password (or *license key*) is an encrypted character string that specifies the characteristics of the license. This information, determined by the vendor, includes:

The specific number and type of license contained in the password
The date when the licenses become active
The date when the licenses expire

Vendors can create two types of password: *simple* and *compound.*

A simple password, once enrolled on a license server, represents one or more licenses that the license server can grant when an end user starts the product.

A compound password, once enrolled on a license server, is a single password from which you can extract multiple simple passwords. Each extracted simple password represents one or more licenses. The compound password is a means of:

Efficiently distributing multiple licenses from the vendor to the customer.

Distributing licenses to different license servers, when required. The compound password must be installed on a specific license server. Extracted passwords can be distributed as required to other license servers that are not specified in the compound password.

Providing a sales representative with a set of licenses that the representative can distribute to different customers.

A compound password contains an expiration date that the vendor sets. The duration of extracted licenses cannot be longer than the time remaining before the compound password expires.

The vendor includes the password, along with other information about the application, in the *enrollment certificate file* (ECF).

Figure 1 on page 6 summarizes the relationship among the license, the license password, the compound password, and the enrollment certificate file.

Password Use Control Levels

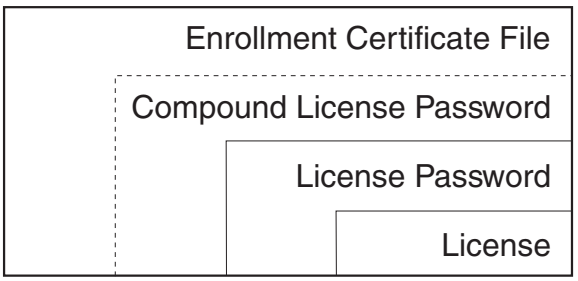


Figure 1. Licensing Concepts Summary

Password Use Control Levels

Vendors can enable their products according to either of the following predefined use control levels:

Vendor-Managed Use Control (VMU)
Customer-Managed Use Control (CMU)

Vendor-Managed Use Products

With *vendor-managed use* products, the vendor manages compliance with the terms of the acquisition of the software product.

When you acquire licenses for a vendor-managed use product, the product vendor will ask you to supply the unique identifier (target ID) of each machine where you intend to install the product licenses. For nodelocked licenses, this is the identification of the workstation where the enabled product is to be installed; for network licenses, this is the network license server. The password, tied to the specified workstation, cannot be used on another workstation. You must also supply the number of licenses you want to acquire. The vendor uses this information to create the password that you use to install and activate the licenses.

Vendors typically ship a vendor-managed use product with a simple password. They can also use compound passwords for this purpose. You can extract and distribute a limited number of licenses from the compound password, up to the maximum the vendor set in the compound password.

If you want to change the terms and conditions of the contract (for example, to increase the number of licenses), you provide the vendor with similar information for each of the machines on which you intend to install the licenses and get a new password.

A vendor can produce a vendor-managed use license password with target ID set to ANY, so that you can install it on any machine. The vendor can deliver such a password with the product package without your specifying how many licenses you want to acquire. Instead, the vendor sets an upper limit, possibly very large, on the number of licenses you can install on each license server. By generating and

delivering such a password, the vendor has decided not to perform the checks that are normally associated with vendor-managed use.

Customer-Managed Use Products

To provide vendors with greater flexibility in the way they deliver licensed software, License Use Runtime supports *customer-managed use* control. With products that are enabled in this way, the vendor does not directly associate licenses with a particular license server (or group of license servers). The vendor does not set an upper limit on the number of licenses that you are entitled to use. Instead, it is your responsibility to set that upper limit, depending on the terms of the software product acquisition.

License Use Runtime provides you with the information on the usage of the enabled products, thereby helping you stay within the boundaries of the acquisition agreement. Transactions, such as enrollment, distribution of licenses, updates, and deletions, are logged in a tamper-proof License Use Runtime database.

Vendors typically ship a customer-managed use product with a compound password that you can use to extract and distribute the number of licenses you have acquired. They can also use simple passwords for certain types of licenses.

License Types

This section describes the types of license the vendor can select. When you receive a license-enabled product, check the product documentation to determine the license type.

Nodelocked Licenses

A nodelocked license allows the use of a product on the particular machine for which the license was created for as long as the license remains valid. Vendors typically use nodelocked licenses for standalone, rather than client/server, applications.

A vendor who is enabling a product that uses nodelocked licenses can choose between two license-enabling models: *non-runtime-based* and *runtime-based.*

If the vendor chooses non-runtime-based enabling, the license-enabled product itself, rather than License Use Runtime, manages use of the nodelocked license. The password for such a product is stored in a file that is called the *nodelock file.* When you start the application, it checks the nodelock file to ensure you have a valid license.

If the vendor chooses runtime-based enabling, management of the nodelocked license is performed by the *nodelocked license server* on the local machine. You interact with the nodelocked license server through the Basic License Tool. It enables you to view and update information about the nodelocked licenses on the machine and get reports about their use.

See “License-Enabling Models” on page 15 for more information about license-enabling models.

License Types

Vendors can enable their products to use the following kinds of nodelocked licenses:

Simple nodelocked licenses
Concurrent nodelocked licenses
Use-once nodelocked licenses
Per-server licenses

Simple Nodelocked Licenses

A *simple nodelocked* license allows an unlimited number of simultaneous uses of the licensed application on the local machine. Simple nodelocked licenses are valid only for vendor-managed use products. A word processor is a typical example of a product that uses nodelocked licenses.

Concurrent Nodelocked Licenses

As with a simple nodelocked license, the *concurrent nodelocked* license is local to the node where the application has been installed. It allows a limited number of simultaneous uses of the licensed application. A typical example of a concurrent nodelocked license is a client/server application. The application server is able to recognize the number of clients connected to it and ask for a license for each of them.

Vendors can use concurrent nodelocked licenses for both vendor-managed and customer-managed products.

When you enroll a customer-managed product, you must specify how many concurrent nodelocked licenses you have acquired for the product. The administrator can modify this number at any time.

Use-Once Nodelocked Licenses

A *use-once nodelocked* license permits a single use of a licensed product on a particular machine during the period the license is valid. Every time the product is started, one license is consumed.

A typical use of use-once nodelocked licenses is to distribute promotional or demonstration versions of software.

Vendors also provide use-once nodelocked licenses to supplement concurrent nodelocked licenses during times when user demand for those products exceeds the number of available concurrent nodelocked licenses. The vendor designs the product so that when all concurrent nodelocked licenses for the product are in use, a user can request an available use-once license.

Vendors can use use-once nodelocked licenses for both vendor-managed and customer-managed products.

When you enroll the licenses for a customer-managed product, you must specify how many use-once nodelocked licenses you have acquired for the product. The administrator can modify this number at any time.

Per-Server Licenses

Per-server licenses are exactly like concurrent nodelocked licenses, except that at any time, you can change them into per-seat licenses (see “Per-Seat Licenses” on page 11 and “Switching from Per-Server to Per-Seat Licenses” on page 14).

Vendors use per-server/per-seat licenses to enable client/server applications constructed for multiple-server solutions. With both per-server and per-seat licenses, the server of a licensed client/server application can request licenses for its clients. The application clients need not be license-enabled.

With per-server licensing, each application server license is associated with a specific number of application clients. This represents the maximum number of application clients that may concurrently request services from that application server. The application client licenses are stored locally on the application server machine and are granted temporarily to requesting application clients. Multiple application servers grant licenses independently of one another; if the same application client connects to more than one application server, the application client is granted more than one license. You should therefore probably use per-server licenses only in an environment where:

Each application client connects to only a single application server, or
Each application client uses the application infrequently for brief periods.

When your environment grows in such a way that application clients are connecting to multiple application servers, you will probably convert your per-server licenses to per-seat. With *per-seat* licensing, unused application client licenses are kept in a central repository, which all the application servers share. They also share a central list of application clients to which a license has been assigned. When a license is assigned to an application client, the license remains assigned to the application client even when it is not using the product. If an application client connects to multiple application servers, it is assigned only one license.

Per-server licenses are valid only for customer-managed use products.

Network Licenses

Network licenses, rather than being restricted to a single machine, are stored on a network license server and shared among multiple network license clients.

License Types

Vendors can enable their products to use the following kinds of network licenses:

Concurrent licenses
Reservable licenses
Use-once licenses
Per-seat licenses

Concurrent Licenses

A *concurrent* license is a network license that can be temporarily granted to run the licensed application on a client.

When the product is running, that license remains unavailable to other users of the product. When the product stops running, the license is returned to the server, where it becomes available to other users.

Concurrent licenses allow as many users to run a licensed application simultaneously as there are valid licenses for the product available from the network license servers in your licensing environment.

A typical use of concurrent licenses is for products with relatively expensive licenses that each user will use only some of the time. The customer orders fewer licenses than there are users to optimize use of the licenses. Such applications may be either client/server applications, for which the client is enabled, or non-client/server applications.

Vendors can use concurrent licenses for both vendor-managed and customer-managed products.

Reservable Licenses

A *reservable* license is a network license that you can reserve for the exclusive use of a user, a group, or a node. The reservation is for a specified time period. A reservable license that has been reserved is called a *reserved* license. A reservable license that has not been reserved is called an *unreserved* license.

When a reserved license is granted from the network, the license is stored on the workstation where the licensed application is running. Thereafter, the license can be used on the workstation, even if the workstation is disconnected from the network, until the reservation expires.

A typical use of reservable licenses is for the client part of a client/server application that is likely to run on a portable computer that is often disconnected from the network. Another typical use is for a compiler being used in software development. During a build process involving many compilations, it is more efficient to reserve a compiler

license for a day or two than to make a separate request for a compiler license for every compilation.

You can reserve some of the reservable licenses for an application and leave others unreserved. Unreserved licenses are treated like concurrent licenses.

Vendors can use reservable licenses for both vendor-managed and customer-managed products.

Use-Once Licenses

A *use-once* license is a network license that permits a single use of a licensed product during the time the license is valid. Every time the product is started, one license is consumed.

A typical use of use-once licenses is to distribute promotional or demonstration versions of software.

Vendors also provide use-once licenses to supplement concurrent licenses when user demand for those products exceeds the number of available concurrent licenses. The vendor designs the product so that when all concurrent licenses for the product are in use, a user can request an available use-once license.

Vendors can use use-once licenses for both vendor-managed and customer-managed products.

Per-Seat Licenses

Vendors use per-server/per-seat licenses to enable client/server applications constructed for multiple-server solutions. With both per-server and per-seat licenses, the server of a licensed client/server application can request licenses for its clients. The application clients need not be license-enabled.

With per-seat licensing, unused application client licenses are kept in a central repository, which all the application servers share. They also share a central list of application clients to which a license has been assigned. When a license is assigned to an application client, that assignment is permanent. If an application client connects to multiple application servers, it is assigned only one license.

You will probably want to use per-seat, rather than per-server, licenses in an environment where application clients connect to multiple application servers. (See also “Per-Server Licenses” on page 9.)

Per-seat licenses are valid only for customer-managed use products.

License Policies

License Policies

Vendors can enable their products to implement various policy decisions regarding how licenses are managed.

Vendor-Controlled Policies

The vendor can implement the *try-and-buy, multiuse rules, product wait queues*, and *license annotation* license policies.

Try-and-Buy Policy

The vendor can enable a product with a special simple nodelocked license for customers to use during an evaluation period. The evaluation period (with duration set by the vendor) starts either when the product is enrolled or when the product is run for the first time.

Multiuse Rules

Multiuse rules define the conditions under which multiple invocations of a product require only a single license. These rules are applicable only to concurrent, concurrent nodelocked, and per-server licenses.

The vendor can enable a product so that after a license has been granted to a particular user, group, or node, a second invocation of the product does not require a second license. For example, if a user invokes a compiler repeatedly, a multiuse rule might specify that the second and subsequent invocations do not require additional licenses.

Multiuse rules may be based on any combination of the following tests that the server performs when a concurrent license is requested:

- The request for a license is associated with the same user as a previous request.
- The request for a license is associated with the same group as a previous request. The vendor can also change the meaning of the “same group” rule to implement a vendor rule.
- The request for a license is associated with the same node as a previous request (applicable to concurrent licenses only).
- The request for a license is associated with the same job ID as a previous request.

For information about how to implement multiuse rules, see “Defining Rules for Multiple-Use Concurrent Licenses” on page 174.

Product Wait Queues

Some license-enabled products with concurrent licenses may use wait queues.

When a user invokes such a product, and there are no concurrent licenses currently available, the product can be enabled to ask if the user wants to wait for a license. If

the user responds affirmatively, the user is added to the wait queue on each License Use Runtime network license server that provides concurrent licenses for the product. User names are added to the wait queues in chronological sequence. When a license becomes available, it is granted to the first user in the queues. The next user in the queues becomes eligible for the next available license.

License Annotation

License annotation is data that is defined and included as part of the license information when a license is created. When the license is granted, the data is passed to the enabled application for its own use. Licenses of any type can be annotated.

A typical use of license annotation is to create licenses that correspond to different configurations of the same product. Consider an application that has several optional priced features, all delivered as part of the product package. The vendor can create license annotations to define which options the customer has bought and, therefore, which features are accessible to the end user.

Custom Configuration

Vendors who want to offer selected combinations of products, tailored more precisely to the needs of users, can define custom configurations by adding functions and products to a base configuration.

You specify the required content of a custom configuration when you order the configuration. You can order a custom configuration for one seat or for a block of any number of identical seats. If you order a configuration for a block of seats, the quantity of each add-on function or product must equal the number of seats in the block.

Each custom configuration, whether for a single seat or for a block of two or more seats, is assigned a separate custom configuration license. A custom configuration license is a special case of either a concurrent network license or a simple nodelocked license that contains a unique serial number identifying that custom configuration. The single serial number and license for a block configuration helps you to manage your installed licenses more easily.

After initial installation of a custom configuration, you can better manage the evolution and growth of your configurations, by ordering additional “add-on” functions and products, as necessary. To retain a single serial number and license, however, any changes made to the custom configuration must be applied to all seats under that serial number.

Customer-Controlled Policies

The customer can exercise the *hard stop/soft stop, user access restriction*, and *per-server/per-seat switch* license policies.

Hard Stop/Soft Stop Policy

The vendor can enable a product so that you can choose the behavior of the product when the end user starts it and no licenses are available.

License Policies

If no license is available, one of two things can happen:

The product does not start, and there is no way for the end user to go on (*hard stop policy*).

The product starts (*soft stop policy*).

Only applications with customer-managed use licenses can be enabled for the hard stop/soft stop policy. Vendor-managed use licenses are always hard stop licenses.

When you enroll a product enabled for hard stop/soft stop, the default is soft stop. You can use the Basic License Tool to change the policy to hard stop and back again at any time.

License Use Runtime keeps track of the soft stop licenses in use, so that you can get meaningful information about license use.

When the soft stop policy is set, License Use Runtime keeps track of the *high-water mark*. The high-water mark is the maximum number of licenses ever granted for a given product beyond the number of licenses that are enrolled for that product. You can see this number through the Basic License Tool, and you can reset it to 0. Use this number to help you decide the number of additional licenses you need. When the hard stop policy is selected, the number of in-use licenses cannot exceed the number of enrolled licenses, so the high-water mark is not maintained.

User Access Restriction

You can use the *user file* to control which users have access to licenses for specific products. The user file is a flat ASCII file that you create using a text editor. For each product in the file, there is a list of users. It lists either those who are allowed to use the product (in which case no one else can use it) or those who are not allowed to use it (in which case anyone else can use it).

See “Scenario 10: Restricting User Access” on page 113 for details.

Switching from Per-Server to Per-Seat Licenses

Vendors of client/server applications who choose per-server/per-seat licensing provide you with two enrollment certificates:

The per-server certificate, containing a per-server password.
The per-seat certificate, containing a per-seat password.

You have the option to start in per-server mode, and switch at any time to per-seat mode, or start directly in per-seat mode. Once the per-seat mode has been activated, it is not possible to go back to per-server mode.

See “Per-Server Licenses” on page 9 and “Per-Seat Licenses” on page 11 for information to help you decide between per-server and per-seat. See “Scenario 8: Switching from Per-Server to Per-Seat Licenses” on page 106 for information about how to perform the switch.

License-Enabling Models

To summarize, the product vendor can create license-enabled products that use nodelocked or network licenses. The enablement of nodelocked licenses can be either *non-runtime-based* or *runtime-based*.

If the vendor chose non-runtime-based enabling (nodelocked licenses only), the product does not make use of License Use Runtime on the machine where the product runs. Following the vendor's installation instructions, you may be required to store the password for such a product in a vendor-selected nodelock file. When you start the application, it checks the nodelock file to ensure you have a valid license. It is not necessary for the nodelocked license server to be running for the license to be granted. Information about use of the product is not logged. You cannot use the Basic License Tool to view information or get reports about the product and its usage.

If the vendor chose runtime-based enabling for a product with nodelocked licenses, the product makes use of License Use Runtime on the machine where the product runs. It does not require configuration unless the end user has special requirements. The password for such a product is stored in the nodelocked license database. When you start the application, it contacts the nodelocked license server, which checks its database to ensure you have a valid license. Information about use of the product is logged. You can use the Basic License Tool to view information or get reports about the product and its usage.

A network license-enabled product makes use of License Use Runtime on the machine where the product runs and requires some limited configuration on that machine. The licenses are stored on one or more network license servers. When the user at a client starts a licensed program, License Use Runtime at the license server determines whether or not a license is available.

Table 2 on page 16 summarizes the license-enabling models, license types, and license policies.

Scalable Installation and Configuration

Table 2. License-Enabling Models, License Types, and License Policies

License-Enabling Model	License Types Available	License Policies Available
Nodelocked* License-Enabled Products (Non-Runtime-Based Enabling)	Simple Nodelocked	Try-and-Buy License Annotation Custom Configuration
Nodelocked* License-Enabled Products (Runtime-Based Enabling)	Simple Nodelocked	Try-and-Buy License Annotation
	Use-Once Nodelocked	User Access Restriction License Annotation
	Concurrent Nodelocked, Per-Server	Hard Stop/Soft Stop Multiuse Rules User Access Restriction License Annotation
Network License-Enabled Products	Concurrent	Hard Stop/Soft Stop Multiuse Rules User Access Restriction Product Wait Queues* License Annotation Custom Configuration
	Reservable	Hard Stop/Soft Stop (when unreserved) User Access Restriction License Annotation
	Use-Once	User Access Restriction License Annotation
	Per-Seat	Hard Stop/Soft Stop User Access Restriction Per-Server/Per-Seat Switch* License Annotation

Note: * This applies only to C-language applications.

Scalable Installation and Configuration

License Use Runtime consists of separate installable components, so that you can install exactly what you need on each workstation.

For example, in AIX 4.3.3, all required License Use Runtime components are automatically installed on every AIX workstation as part of AIX 4.3.3 installation. You can install optional components, such as the graphical user interface, either when you install AIX or later.

On OS/2 and Windows, you can select the appropriate components depending on the role the workstation is to play in your licensing environment. On OS/2, Windows NT (x86), Windows NT Alpha, Windows Terminal Server (x86), Windows Terminal Server

Alpha, Windows 95, and Windows 98, there are components for runtime, communications, and online documentation. On OS/2, there is also a component for namespace binding support. Vendors can, optionally, incorporate the communications component into the installation images of their license-enabled products at the minimum level of installation and configuration the products require. Alternatively, they can specify that you should download and install License Use Runtime.

When you configure License Use Runtime, the configuration tool recognizes which components are installed and presents only the options consistent with the installed component.

The configuration tool, for configuring License Use Runtime license servers and clients, has a command-line interface on all platforms, a graphical user interface on AIX, Windows, and OS/2, and an interactive script interface on all UNIX platforms.

License Creation Tool

License Use Runtime includes a tool that creates product licenses for the use of vendors who create license-enabled products. Two uses of the license creation tool are:

The tool enables vendors to create these kinds of passwords:

- Test passwords, for use in testing while enabling a product.
- Production passwords, to deliver to customers.

  To create production passwords, vendors must acquire the license for this tool from IBM or from Isogon Corp. The address of Isogon Corp. is:

  Isogon Corporation
  330 Seventh Avenue
  New York, New York 10001
  U.S.A.
  Tel: (+1) 212-376-3200
  Fax: (+1) 212-376-3280

  Distribution of production passwords to customers depends on the use control level of the license-enabled products:

  For customer-managed use control products, the customer receives the license password together with the product package.

  For vendor-managed use control products, for IBM license-enabled products, the customer requests the license password from the IBM country software password distribution center. For non-IBM license-enabled products, the customer requests the license password from the vendor software password distribution center.

License Administration Tool

The tool is also useful for vendor sales representatives, who can be provided by the vendor with a production compound password for a vendor-managed use product. The compound password contains many licenses, from which the sales representative extracts licenses for individual customers.

For details about how to use this tool, see “i4lct - License Creation Tool” on page 164.

License Administration Tool

License Use Runtime includes a license administration tool, which is called the *Basic License Tool*.

The Basic License Tool has a command-line interface on all platforms, and a graphical user interface on AIX, OS/2, Windows NT (x86), Windows NT Server Terminal Server Edition (Windows Terminal Server) (x86), Windows 95, and Windows 98. It enables you to:

Manage all types of licenses

The administrator can use the Basic License Tool to manage nodelocked and network licenses.

Add, update, or delete licenses

Add licenses to or delete licenses from the network license server or nodelocked license server database; update information about existing customer-managed use licenses.

Display information

Display a notebook of information about the licenses that are installed for each product.

Distribute licenses

Extract licenses from a compound password and distribute them among the network license servers available on the network.

Reserve licenses

Manage the reservation of reservable licenses for the exclusive use of certain users.

Manage multiple network and nodelocked license servers

From any properly configured machine, you can view and manage licenses that are installed on any network license server and on any nodelocked license server in the network. Working at a single administration site, you can manage all kinds of licenses on all machines. The capability to manage licenses on nodelocked license servers is particularly useful for per-server and concurrent nodelocked licenses.

Generate reports

Standard Event Report. Displays detailed information about significant events that occur on the license servers that you specify.

License Request by Product Report. Displays statistical information about the use of the licenses of a product in the time interval that you specify. For each product, it reports the licenses that were requested, the licenses that were granted, and the percentage of rejections.

License Request by User Report. Displays statistical information about the use of products by users in the time interval that you specify. For each user, it reports the licenses that were requested, the licenses that were granted, and the percentage of rejections.

License Use by Product Report. Displays statistical information about the use of the licenses of a product in a specified time interval. For each product, it reports:

- The maximum number of nodes that used licenses for the product at the same time
- The maximum number of users that used licenses for the product at the same time
- The average time the licenses were in use

License Use by User Report. Displays statistical information about the use of the licenses by each user in a specified time interval. For each user, it reports the times the user requested licenses and the length of time the user kept the licenses in use.

Customer-Managed Use Audit. Reports the following information for customer-managed use product transactions:

- Vendor name
- Product name
- Product version
- Administrator information
- Time stamp
- Number of licenses
- Transaction type (for example, product enrolled, license distributed, license deleted, license updated, per-server/per-seat license migrated)
- Signature stamp (user, group, and node)
- Signature information

Use the high-water mark

When the soft stop policy is in effect, the high-water mark is recorded in the licensing database. The high-water mark is the maximum number of licenses ever granted for a given product beyond the number of licenses that are enrolled for that product. You can see this number through the Basic License Tool and can reset it to 0. This number assists you in deciding how many additional licenses you need. When the hard stop policy is selected, the number of in-use licenses cannot exceed the number of enrolled licenses, so the high-water mark is not maintained.

High-Availability

Set the threshold

You can set a threshold percentage of licenses. If more than the threshold percentage of licenses for a product are in use, messages about the level of usage are logged. There is a single threshold that applies to all vendor-managed products and, by default, to customer-managed products. You can change that threshold, and you can also set a separate threshold for each customer-managed product.

Exercise customer-controlled policies

You can switch between hard stop and soft stop, switch from per-server to per-seat, and manage the identifiers of application clients using per-seat licenses.

High-Availability Licensing

High-availability licensing enables you to set up an environment in which there is a very high degree of certainty that concurrent licenses will be available, even if a network license server goes down.

When you use this option, you create a *cluster* of network license servers. A cluster is a group of from 3 to 12 network license servers that jointly serve vendor-managed concurrent licenses that are enrolled on the cluster rather than on an individual server.

You can create and administer a cluster, and administer high-availability licenses, from any machine. However, only AIX, HP-UX, IRIX, Solaris, Windows NT, Windows NT Alpha, and Windows Terminal Server network license servers can be members of a cluster: no OS/2, Windows 95, or Windows 98 network license server can be a member of a cluster.

While some servers in the cluster are serving licenses, one or more servers remain in reserve, ready to take over should an active server fail.

Each active server serves an equal share of the licenses enrolled on the cluster. When a server becomes unavailable and another server takes its place, responsibility for the licenses is automatically redistributed among active servers.

For high-availability licensing to work for a particular product, the product vendor must supply a password tied to a cluster rather than to an individual target server.

High-availability licensing works only with the IP protocol and does not support the product wait queue policy. Before you decide to use high-availability licensing for a product, make sure such a password is available from the product's vendor.

High-availability licensing is recommended only for users who are already experienced with managing individual license servers and who already have a stable licensing environment working.

See “Planning Clusters” on page 41 for planning information, and “Scenario 11: Creating and Administering a Cluster” on page 114 for an example.

Backup Procedure

On license servers, there is an automatic backup procedure for License Use Runtime databases and files.

Working with Licensed Products

This section explains what happens when a user starts a licensed product, depending on how the product is enabled.

In the figures and text in this section, references to the *enabled application* or *enabled applet* refer to the application or applet itself (which contains API calls) plus the embedded code that services the API calls (which carries out the described steps).

This section assumes that, if required:

License Use Runtime is installed properly.
License Use Runtime is configured properly.
A Web server machine is set up properly for Java applications and applets.
The network is running properly.

If not, what happens depends on how the vendor enabled the product. See the product documentation for details.

Central Registry License Server

Some of the scenarios in this section show the use of a special server, which is called the *central registry* license server. The central registry is a repository of information that all the other network license servers can use. If you plan to install customer-managed use products with network licenses, or products with reservable licenses, you must identify one (and only one) central registry. Otherwise, the central registry license server is not required.

Some of the uses of the central registry are:

All the per-seat licenses in the licensing environment are installed on the central registry.

The list of application clients to which per-seat licenses have been granted is maintained in the central registry.

Soft stop licenses are tracked in the central registry.

The high-water mark is recorded in the central registry.

Reserved licenses that have not yet been granted to a user are kept in the central registry.

Working with Licensed Products

Working with Nodelocked Licenses (Non-Runtime-Based Enabling)

Figure 2 shows what happens when an end user invokes an application with nodelocked licenses for which the vendor chose non-runtime-based enabling. The licenses must be simple nodelocked licenses.

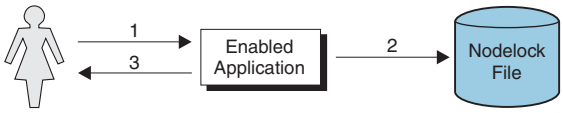


Figure 2. Using a Nodelocked License (Non-Runtime-Based Enabling)

1 The end user invokes the application.

2 The application checks the nodelock file to ensure a license is stored on the local system.

3 If there is a valid license in the nodelock file, the application runs. If not, depending on how the vendor enabled the application, it may return information to the end user, or it may run even with no license available.

For information about how to edit a nodelock file, see Appendix B, “Using the Nodelock File” on page 209.

Working with Nodelocked Licenses (Runtime-Based Enabling)

Figure 3 shows what happens when an end user invokes an application with nodelocked licenses for which the vendor chose runtime-based enabling. The licenses can be simple nodelocked, use-once nodelocked, or concurrent nodelocked licenses.

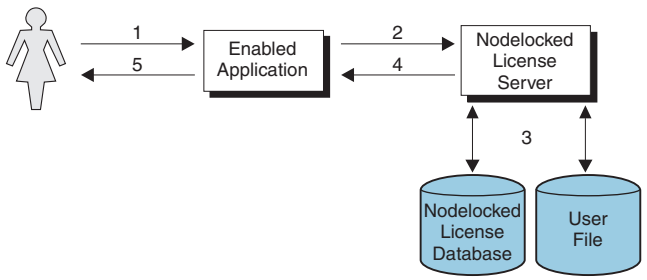


Figure 3. Using a Nodelocked License (Runtime-Based Enabling)

1 The end user invokes the application.

2 The application requests a license from the nodelocked license server on the local system.

3 The nodelocked license server checks that there is a valid license on the machine and that this user is authorized to use it.

If there is no nodelocked license but the application uses concurrent nodelocked licenses and implements the soft stop policy, the nodelocked license server checks for a soft stop license and checks the user file for authorization.

4 The nodelocked license server returns the status of the license request to the application.

5 If a license was found and granted, or if a soft stop license was granted, the application runs. If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Working with Licensed Products

Working with Use-Once Licenses

Figure 4 shows what happens when an end user invokes a C-language application with use-once licenses.

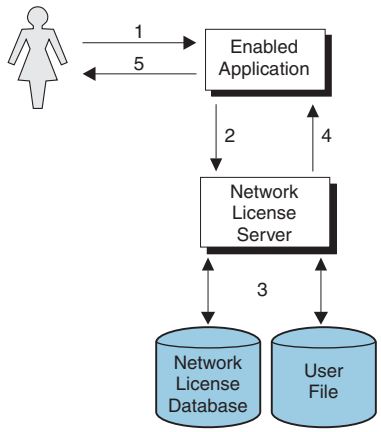


Figure 4. Using a Use-Once License for a C-Language Program

1 The user invokes the application.

2 The application requests a license from the network license server.

3 The network license server checks its license database for an available license and the user file for authorization.

4 The network license server returns the status of the request to the application. If a license was found and granted, the application runs, and one license is subtracted from the number of available use-once licenses.

5 If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Figure 5 on page 25 shows what happens when an end user invokes a Java application or applet with use-once licenses. The primary difference between usage of use-once licenses for C and Java programs is that in the Java case, the Web server machine, rather than the end user's machine, serves as the network license client.

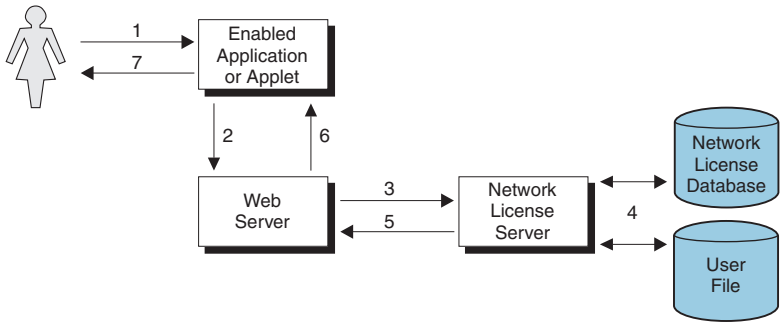


Figure 5. Using a Use-Once License for a Java Application or Applet

1 The user invokes the application or downloads the applet through a Web browser.

2 The application or applet sends a license request to the Web server using the http protocol.

3 The Web server requests a license for the application or applet from the network license server.

4 The network license server checks its license database for an available license and the user file for authorization.

5 The network license server returns the status of the request to the Web server. If a license was found and granted, one license is subtracted from the number of available use-once licenses.

6 The Web server returns the status of the request to the application or applet, using the http protocol. If the status is OK, the application or applet runs.

7 If no license can be granted, depending on how the vendor enabled the application or applet, it may return information to the end user, or it may run even with no license available.

Working with Licensed Products

Working with Concurrent Licenses

Figure 6 shows what happens when an end user invokes a C-language application with concurrent licenses.

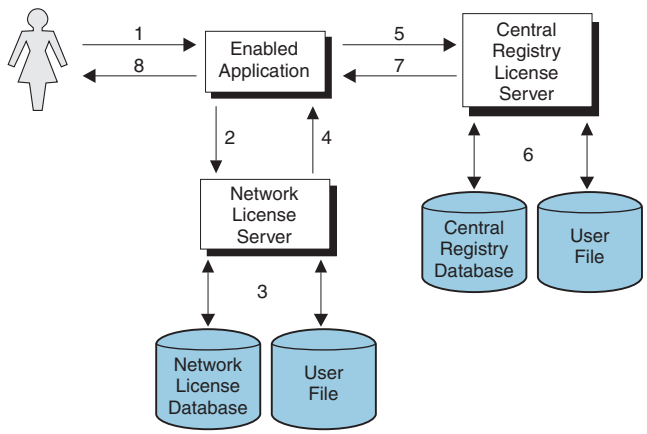


Figure 6. Using a Concurrent License for a C-Language Program

1 The user invokes the application.

2 The application requests a license from the network license server.

3 The network license server checks its license database for an available license and the user file for authorization.

4 The network license server returns the status of the request to the application. If a license was found and granted, the application runs.

5 If a network license was not found, and the application implements the soft stop policy, the application requests a soft stop license from the central registry license server.

6 The central registry license server checks its database for a soft stop license and the user file for authorization.

7 The central registry license server returns the status of the request to the application. If a soft stop license was granted, the application runs.

8 If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Figure 7 on page 27 shows what happens when an end user invokes a Java application or applet that has concurrent licenses. The primary difference between usage of concurrent licenses for C and Java programs is that in the Java case, the Web server machine, rather than the end user's machine, serves as the network license client.

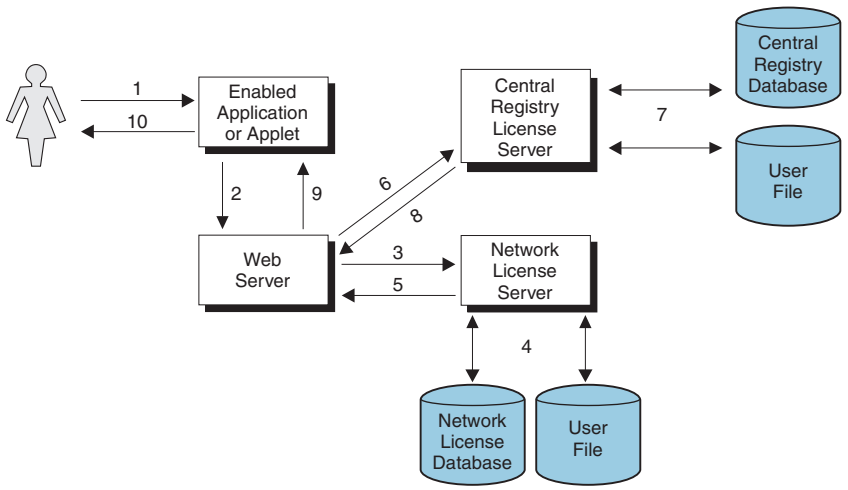


Figure 7. Using a Concurrent License for a Java Application or Applet

1 The user invokes the application or downloads the applet through a Web browser.

2 The application or applet sends a license request to the Web server using the http protocol.

3 The Web server requests a license for the application or applet from the network license server.

4 The network license server checks its license database for an available license and the user file for authorization.

5 The network license server returns the status of the request to the Web server. If a license was found and granted, the Web server returns a positive status to the application or applet, and it runs.

6 If no concurrent license was found, the Web server requests a soft-stop license from the central registry license server.

7 The central registry license server checks its database for a soft-stop license and the user file for authorization.

8 The central registry license server returns the status of the request to the Web server.

9 The Web server returns the status of the request to the application or applet, using the http protocol. If the status is OK, the application or applet runs.

10 If no license can be granted, depending on how the vendor enabled the application or applet, it may return information to the end user, or it may run even with no license available.

Working with Licensed Products

Working with Reservable Licenses

Figure 8 shows what happens when an end user invokes a C-language application with reservable licenses.

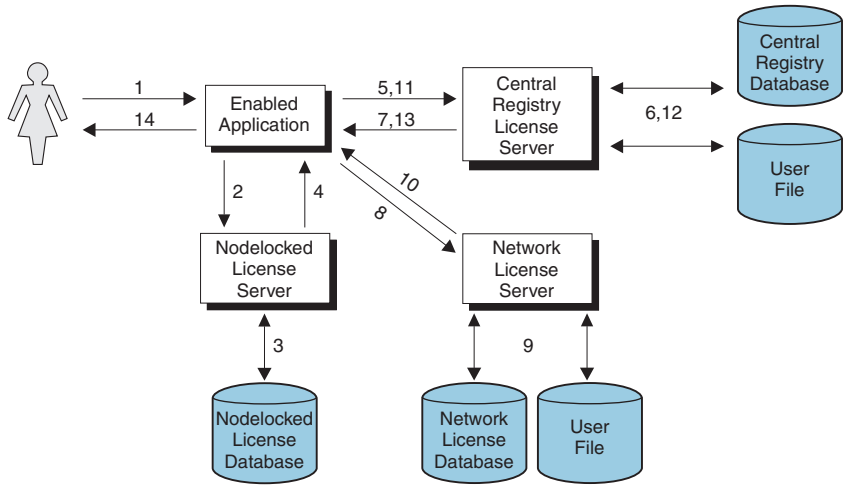


Figure 8. Using a Reservable License for a C-Language Program

1 The user invokes the application.

2 The application requests a reserved license from the nodelocked license server.

3 The nodelocked license server checks its database for a reserved license. This is a license that you reserved for the user. It was granted to the user and stored on the local machine, in response to a previous request.

4 The nodelocked license server returns the status of the request to the application. If a license was found, the application runs.

5 If the nodelocked license server does not find a license, the application requests a reserved license from the central registry license server. This is a license that you have reserved for this user, group, or workstation.

6 The central registry license server checks its database for a reserved license and the user file for authorization.

7 The central registry license server returns the status of the request to the application. If a reserved license was found and granted, it is stored in the nodelocked license server's database, and the application runs.

8 If a reserved license was not found, the application requests a reservable license from the network license server. This is a reservable license that you have not reserved for anyone.

9 The network license server checks its license database for a reservable license and the user file for authorization.

10 The network license server returns the status of the request to the application. If a reservable license was found and granted, the application runs.

11 If a reservable license was not found, and the application implements the soft stop policy, it requests a soft stop reservable license from the central registry license server.

12 The central registry license server checks its database for a soft stop reservable license and the user file for authorization.

13 The central registry license server returns the status of the request to the application. If a soft stop license was granted, the application runs.

14 If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Figure 9 shows what happens when an end user invokes a Java application or applet with reservable licenses. The primary differences between usage of reservable licenses for C and Java programs are that in the Java case:

The Web server machine, rather than the end user's machine, serves as the network license client.

Reserved licenses, when granted, are not moved to the nodelocked license server.

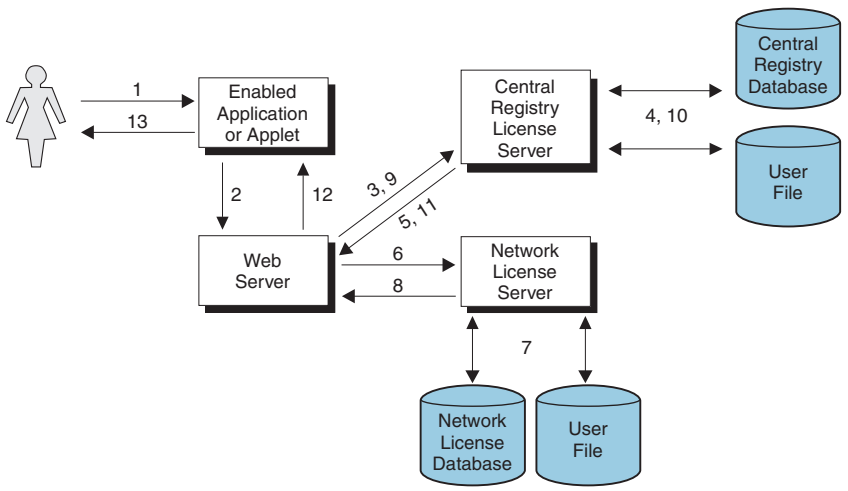


Figure 9. Using a Reservable License for a Java Application or Applet

1 The user invokes the application or downloads the applet through a Web browser.

2 The application or applet requests a reserved license from the Web server using the http protocol.

3. The Web server requests a reserved license for the application or applet from the central registry license server. This is a license that you have reserved for this user, group, or workstation.
4. The central registry license server checks its database for a reserved license and the user file for authorization.
5. The central registry license server returns the status of the request to the Web server. If a reserved license was found and granted, the Web server returns a positive status to the application or applet, and it runs.
6. If a reserved license was not found, the Web server requests a reservable license from the network license server. This is a reservable license that has not been reserved for anyone.
7. The network license server checks its database for a reservable license and the user file for authorization.
8. The network license server returns the status of the request to the Web server.
9. If no reservable license was found, the Web server requests a soft-stop license from the central registry license server.
10. The central registry license server checks its database for a soft-stop license and the user file for authorization.
11. The central registry license server returns the status of the request to the Web server.
12. The Web server returns the status of the request to the application or applet, using the http protocol. If the status is OK, the application or applet runs.
13. If no license can be granted, depending on how the vendor enabled the application or applet, it may return information to the end user, or it may run even with no license available.

Working with Per-Server Licenses

Figure 10 shows what happens when an end user invokes an application with per-server licenses when per-seat has not been enabled. In the figure, the application server is license-enabled.

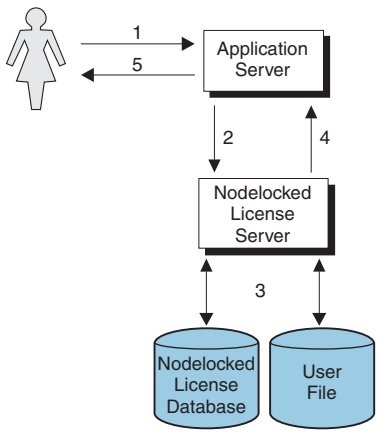


Figure 10. Using a Per-Server License

1 The application client user invokes the application.

2 The application server requests a per-server license from the nodelocked license server. This is a license that you have stored on the nodelocked license server.

3 The nodelocked license server checks the nodelocked license database for such a license and the user file for authorization.

If no license is found, but the application implements the soft stop policy, the nodelocked license server checks for a soft stop license.

4 The nodelocked license server returns the status of the request to the application server. If a license was found, or if a soft stop license was granted, the application runs.

5 If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Working with Licensed Products

Working with Per-Seat Licenses

Figure 11 shows what happens when an end user invokes a C-language application with per-server/per-seat licenses when per-seat has been enabled. In the figure, the application server is license-enabled.

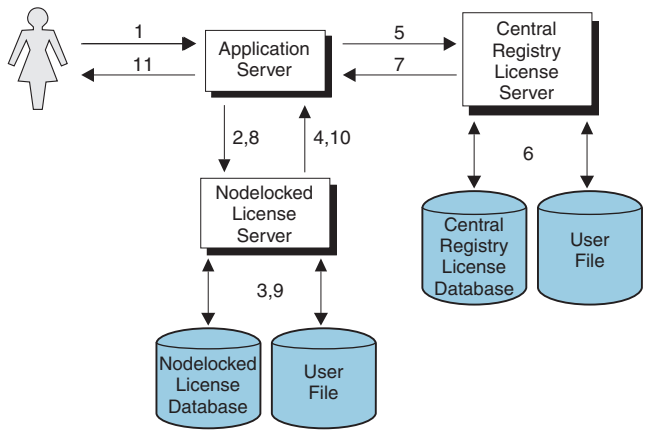


Figure 11. Using a Per-Seat License for a C-Language Program

1 The application client user invokes the application.

2 The application server requests a per-seat license from the nodelocked license server. This is a license that has already been granted to the user on a previous request and stored on the local machine.

3 The nodelocked license server checks the nodelocked license database for such a license.

4 The nodelocked license server returns the status of the request to the application server. If a per-seat license was found, the application runs.

5 If no per-seat license was found on the nodelocked license server, the application server requests a per-seat license from the central registry license server.

6 The central registry license server checks whether a license is already being used by the requesting application client, possibly granted through another application server. In such a case the application can start without having a new license granted.

Otherwise, the central registry license server checks whether a per-seat license is available. If so, it grants the license and records the application client identifier.

If no per-seat license is found, but the application implements the soft stop policy, the central registry license server checks for a soft stop license.

7 The central registry license server returns the status of the request to the application server.

8 If a per-seat or soft stop license was granted, the application sends a shadow copy of the granted per-seat license to the nodelocked license server.

9 The nodelocked license server adds the shadow copy to the nodelocked license database.

10 The nodelocked license server returns the status of the request to the application server, and the application runs.

11 If no license can be granted, depending on how the vendor enabled the product, the application may return information to the end user, or it may run even with no license available.

Figure 12 shows what happens when an end user invokes a Java application or applet with per-server/per-seat licenses when per-seat has been enabled. The primary differences between usage of per-seat licenses for C and Java programs are that in the Java case:

The Web server machine, rather than the end user's machine, serves as the network license client

When a per-seat license is granted no shadow copy is stored on the nodelocked license server.

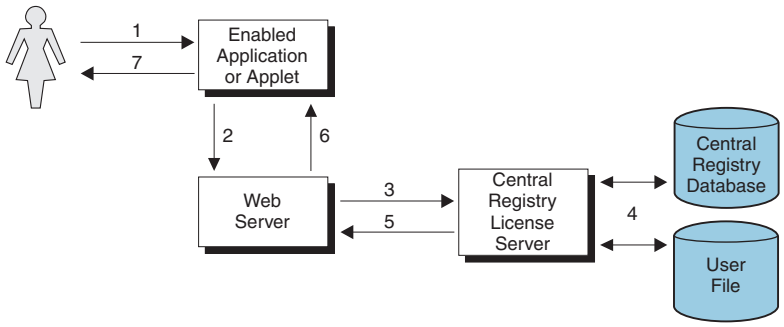


Figure 12. Using a Per-Seat License for a Java Application or Applet

1 The user invokes the application or downloads the applet through a Web browser.

2 The application or applet sends a license request to the Web server. It requests a per-seat license from the Web server, using the http protocol.

3 The Web server requests a license for the application or applet from the central registry license server.

4 The central registry license server checks whether a license is already being used by the requesting application client, possibly granted through another application server. In such a case the application can start without having a new license granted.

Otherwise, the central registry license server checks whether a per-seat license is available. If so, it grants the license and records the application client identifier.

If no per-seat license is found, but the application implements the soft-stop policy, the central registry license server checks for a soft-stop license. If a soft-stop license is available, it is granted.

5 The central registry license server returns the status of the request to the Web server.

6 The Web server returns the status of the request to the application or applet, using the http protocol. If the status is OK, the application or applet runs.

7 If no license can be granted, depending on how the vendor enabled the application or applet, it may return information to the end user, or it may run even with no license available.

Chapter 2. Planning Your Network Licensing Environment

This chapter is for the administrator who is setting up an environment to allow multiple client machines to share licenses stored on one or more network license servers. Skip this chapter if:

- You are using only applications with nodelocked licenses, and
- You do not intend to use the Basic License Tool from one machine to administer licenses on other machines

Designing the network licensing environment that provides the best solution for your business requires careful and thoughtful planning. This chapter will assist you, as the system administrator, to plan the environment for network licensed products and to decide on the configuration options you need.

It is important that you allow enough time for planning, especially when using License Use Runtime in large networks or across subnetworks.

The decisions you need to make include:

- How many network license servers you will set up
- Which machines, if any, will be the network license servers
- How you will distribute product licenses among the network license servers
- Which clients will have access to which servers
- Whether all servers will serve all clients, or whether you will set up independent groups of servers and clients
- Which machine will be the central registry (if required)
- Whether you will make use of License Use Runtime high-availability licensing; if so, which network license servers will be part of a cluster
- Whether and how you will make use of remote administration to administer from one machine the network and nodelocked licenses on other machines
- Which machines will be used as Web servers for Java applications and applets

Selecting Your Servers

The license server system depends on a stable network. If name resolution and routing in a network are not running properly, then the network license servers, network license clients, and central registry license server may be unable to communicate properly.

© Copyright IBM Corp. 1995, 1999

NCS

In designating machines to be network license servers or the central registry license server, keep the following criteria in mind:

A license server should be a computer that stays on at all times. Machines that are frequently unavailable or unreliable, such as those that are brought down often for testing or maintenance, are not good candidates.

It is important to keep license servers for production environments separate from those for test environments.

A computer that is already acting as a file server may well be a good choice, because it satisfies these criteria.

If you have multiple subnetworks, then ideally, the servers should be on the same subnetwork as the majority of clients that will run the licensed products. Accessing in another subnetwork, across a bridge or router, may not be quite as fast.

If the network spans subnetworks, you need to spread the licenses out among network license servers. Also, inside the same network, each client request for a license generates network traffic. Therefore, it can be useful to spread the application licenses across more than one network license server, and across multiple platforms. When a computer is down, the licenses assigned to the network license server on that system are unavailable, but licenses assigned to other network license servers remain available. Having several license servers on the network will help to prevent bottlenecks that result when many clients communicate with a single network license server.

The number of network license servers in the network should be proportional to the frequency of license requests rather than to the number of users. For example, suppose that a compiler and a word processor are both license-enabled. A single user running many short compilations will place a heavier load on the license server than many users each starting the word processor once.

Computers that function as network license servers or as the central registry can also run the license-enabled products. The license server software does not have a noticeable effect on the performance of products.

When you have identified the network license servers, and before you configure them, you must organize the servers into one or more groups. The servers in a group form an independent licensing environment and serve a common set of clients. You must also identify a central registry license server for the group, if it requires one. “Planning Direct Binding” on page 37 will help you to group your servers.

Network Computing System (NCS)

The network computing system (NCS) is a set of tools for distributed computing, some of which are included in the License Use Runtime components.

The Remote Procedure Call (RPC) Runtime Library of NCS provides the calls that enable local programs to execute procedures on remote hosts. These calls transfer requests and responses between clients (the programs that call the procedures), and

servers (the programs that execute the procedures). The RPC that is embedded in all license servers and in enabled products provides a common mechanism to support the request and acquisition of licenses.

Planning Direct Binding

Network license clients locate the network license servers by means of the *direct binding* mechanism.

With direct binding, you make a list of your network license servers and the central registry. (The list is called the *direct binding servers list* in this chapter and in the configuration scenarios in "Setting Up Your Servers and Clients" on page 67).

During configuration of servers and clients, you specify the network addresses of all the servers on the list. They are stored on every server and every client in a local text file, called the *configuration file*.

All network license servers, and the central registry license server, listen for incoming communications on well-known ports

All network license servers, and the central registry license server, listen for incoming communications on well-known ports:

Protocol	Ports	Operating Environments and Systems
NetBIOS	115 and 109	Windows NT and Windows Terminal Server
IPX	1515 and 10999	Windows 95, Windows NT, and Windows Terminal Server
TCP/IP	1515 and 10999	Windows 95, Windows 98, Windows NT, Windows NT Alpha, and Windows Terminal Server

Note: Only TCP/IP is available for the Windows NT Alpha operating system.

The network license client uses these port numbers, together with the network addresses of the server systems that are specified in the configuration file, to locate and connect to the servers.

In addition to enabling clients to locate license servers, the direct binding mechanism makes it possible, from any license server, to use the Basic License Tool remotely administer licenses on all the servers in the direct binding list. By adding nodelocked license servers to the direct binding servers list, you can administer licenses that are on remote nodelocked license servers.

Nodelocked license servers that are configured for remote administration listen on port 12999 if you use TCP/IP or IPX; 129 if you use NetBIOS. The Basic License Tool uses this port number, together with the network addresses of the nodelocked license servers, to locate and connect to the servers for remote administration.

In the same way, you can enable remote administration from a machine configured only as a nodelocked license server. When you configure a nodelocked license server, you

Direct Binding

can create a direct binding servers list that contains all the license servers (both network and nodelocked) whose licenses you want to administer remotely.

Figure 13 shows a simple example of a network configuration that uses direct binding. The example shows a network license server and two nodelocked license servers in a network. From any of the license servers, it is possible to administer licenses on all three license servers.

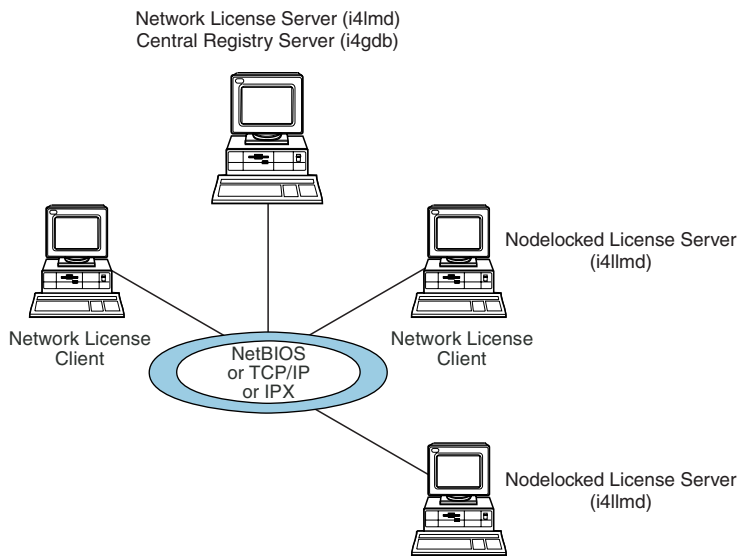


Figure 13. Direct Binding with Network License Servers and Nodelocked License Servers. (Only TCP/IP is available under the Windows NT Alpha operating system.)

Figure 14 on page 39 shows an example of a Java configuration. The example shows two network license servers, a network license client, and a network application client in a Java-enabled network. The Java-enabled applet or application communicates with Server A by means of the Hypertext Transfer Protocol (HTTP), if necessary being protected by a firewall.

Direct Binding

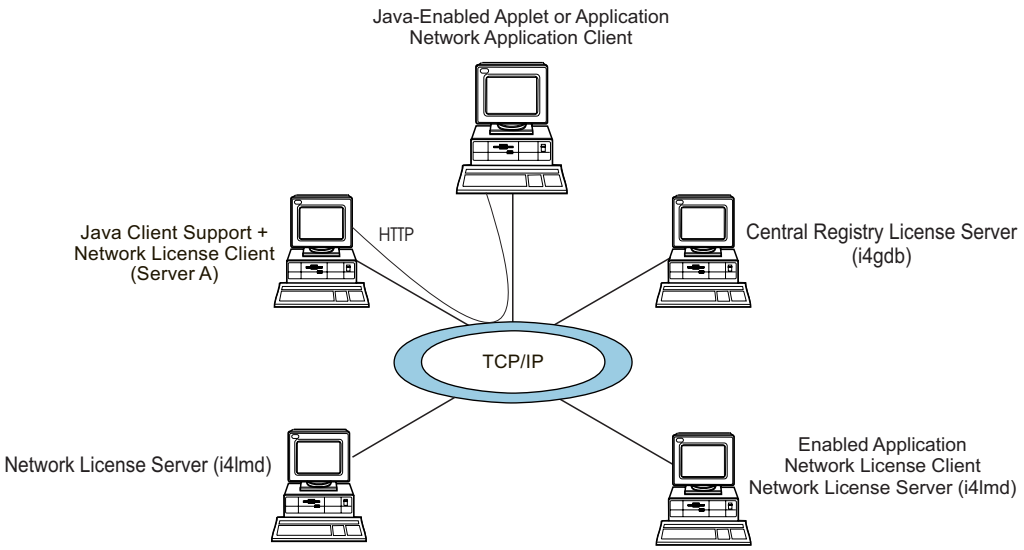


Figure 14. Direct Binding with Java Client Support

Before you begin configuring machines to use direct binding, be sure you have identified all the servers in your direct binding servers list. Then go to the configuration scenarios in "Setting Up Your Servers and Clients" on page 67. As you configure direct binding at each server and at each client, be sure you enter exactly the same list of servers.

Performance Notes:

It is important that the direct binding servers list include all the servers, and that it exclude machines that will not actually function as servers. If there are any extra machines in the list, there will be a noticeable effect on performance.

When a network license client configured for direct binding requests a license, and multiple servers have licenses for the product, the servers are checked for an available license in the sequence they were entered into the direct binding servers list during configuration of the client. Therefore, if you know how frequently specific network license clients will request licenses for specific products, you may be able to balance the workload of the servers by varying the sequence in which servers are defined at different clients.

If you are certain that all the licenses requested by a particular network license client will be supplied by a subset of the servers, when you configure the client you may configure direct binding with just those servers, rather than all servers in the direct binding servers list, to improve performance. If you configure the client in this way, make sure that it is configured to communicate with the necessary servers; otherwise, it will not be able to get licenses.

Planning for Java Applications and Applets

Planning the Central Registry

The *central registry license server* process provides a mechanism for storing licensing information in a database common to all the servers. It is used for the administration of customer-managed use products and products with reservable licenses. There must be one and only one central registry subsystem running in a direct binding servers list. This ensures that the data is accurate and complete.

1. You use the configuration tool to specify where to start the central registry.
2. Select the node where you will run the central registry carefully. After you place the central registry on a node, it cannot be moved.
3. The machine running the central registry must be up and running in order to perform administration tasks on network customer-managed products or on products using reservable licenses.

Planning for Java Applications and Applets

If you plan to use license-enabled Java applications or applets, you will need to set up one or more Web server machines. License-enabled Java applications and applets request licenses from the Web server. The Web server machine, in turn, serves as the network license client.

On the Web server machine, you must:

Run one of the following operating systems:

AIX 4.3.1, 4.3.2, or 4.3.3
Windows NT 4.0 (x86 processor)
OS/2 Warp 4.0
Solaris 2.6 (with the native threads package) or 2.7

Install a Web server and IBM WebSphere. For details about software requirements, see “Installing LUM Java Client Support” on page 63.

Install License Use Runtime Communications component Version 4.5.5 and LUM Java Client Support.

Configure the machine as a network license client, to communicate with network license servers where the licenses for the Java applications and applets are stored.

On each machine where license-enabled Java applications run, in the Java home directory, you must create a file named LicenseClient.properties. The contents of this file must be url=http://hostname, where hostname is the TCP/IP hostname of the Web server machine. This file identifies the Web server to which license requests are to be directed.

Planning Clusters

To take advantage of License Use Runtime high-availability licensing, you set up *clusters* of network license servers connected through TCP/IP. For concurrent licenses with vendor-managed use control only, the software vendor generates passwords that are bound to the cluster rather than to a single server. Some of the servers in a cluster serve licenses, while others wait in reserve to take over in case a serving server goes down. The servers that are serving at any time share equally the responsibility for the licenses that are bound to the cluster, and keep one another informed about the status of the licenses.

You can create and administer a cluster, and administer high-availability licenses, from any machine. However, only AIX, HP-UX, IRIX, Solaris, Windows NT, Windows NT Alpha, and Windows Terminal Server network license servers can be members of a cluster: no OS/2, Windows 95, or Windows 98 network license server can be a member of a cluster.

A network license server that is a member of a cluster can serve licenses that are bound to the server and participate as a member of a cluster at the same time.

High-availability licensing is recommended only for users who are already experienced with managing individual license servers and who already have a stable licensing environment working.

Restrictions on Cluster Size and Composition

For security reasons, it is necessary to impose strict rules on the size and composition of clusters. Be very careful when you decide how many and which servers to put in a cluster. You will not be able to change your decisions after the fact, and they will affect the size and composition of the cluster for as long as it exists.

When you create a cluster (using the GUI or the command-line interface; see “Scenario 11: Creating and Administering a Cluster” on page 114), you specify the initial number of servers in the cluster, and which servers they are. The initial number must be in the range 3 through 10. The first server assigned to the cluster is automatically *activated*; that is, it is available to participate in serving licenses as part of the cluster. You must explicitly activate the other members.

The initial number of servers dictates two important attributes of the cluster:

The minimum number of servers that must be activated in the cluster for the cluster to work

The maximum number of servers that can be added to the cluster in addition to the initial number

Planning Clusters

If you want to replace a server machine that is one of the initial minimum number, to upgrade the hardware or to replace failing hardware, add a new server to the cluster. The number of new servers you can add, even to replace other servers, is limited.

Attention: The initial minimum number of servers must always be in the cluster; they must not be deactivated. If any is deactivated, the cluster ceases to serve licenses.

Passwords that are bound to a cluster are usable on only that cluster. If you find it necessary to delete a cluster and create a new one, or to create additional clusters, you will not be able to use existing passwords on the new cluster.

To delete a cluster, deactivate all its members. When you deactivate a server, it must be up and running.

After a cluster has been created and its members have been activated, the number of activated members determines how many servers must be up and running for the cluster to function.

The relationships between these cluster attributes is shown in Table 3 on page 43.

The minimum number of servers up and running, as shown in the table, is the number of servers that serve licenses. All servers beyond that number are in reserve, waiting to take over if a serving server goes down.

Table 3. Number of Servers in a Cluster

Initial Number	Min No. Activated for Cluster to Work	Max No. Added after Cluster Creation	Actual No. Activated	Min No. Up & Running for Cluster to Work
3	2	1	2 3 4	2 2 3
4	4	2	4 5 6	3 3 4
5	4	1	4 5 6	3 3 4
6	6	2	6 7 8	4 4 5
7	6	1	6 7 8	4 4 5
8	8	2	8 9 10	5 5 6
9	8	1	8 9 10	5 5 6
10	10	2	10 11 12	6 6 7

Examples of Cluster Size Rules

Example 1: Initial number of servers is 3

The following rules apply:

During the life of the cluster, you can add only one server to the cluster. This means you can add a new server with upgraded hardware, and deactivate one of the original three servers, only once during the life of the cluster. This scenario has the effect of replacing a server with an upgraded machine.

Alternatively, you can add a fourth server to the cluster without deactivating any of the original three, thus increasing the cluster size to four servers. Again, you can add a server only once.

No matter whether the cluster has three or four members, at least two members must be activated for the cluster to work.

More than half of the activated servers must be up and running for the cluster to work.

Planning Clusters

Table 4 on page 44 shows how the servers are deployed, depending on how many are activated and how many are up and running.

Table 4. Example - Cluster with Three Initial Members

Number of Activated Members	Number of Members Up and Running	Number of Members Serving Licenses	Number of Servers In Reserve
2	2	2	0*
3	2	2	0*
	3	2	1
4	3	3	0*
	4	3	1

Note: * When the number of servers in reserve is 0, there is no high-availability advantage.

Example 2: Initial number of servers is 6

The following rules apply:

During the life of the cluster, you can add two servers to the cluster. This means you can add two new servers with upgraded hardware, in effect replacing two servers with upgraded machines.

Alternatively, you can add one or two servers to the cluster without deactivating any of the original six, thus increasing the cluster size to seven or eight servers.

Whether the cluster has six, seven, or eight members, at least six members must be activated for the cluster to work.

More than half of the activated servers must be up and running for the cluster to work.

Table 5 on page 45 shows how the servers are deployed, depending on how many are activated and how many are up and running.

Table 5. Example - Cluster with Six Initial Members

Number of Activated Members	Number of Members Up and Running	Number of Members Serving Licenses	Number of Servers In Reserve
6	4	4	0*
	5	4	1
	6	4	2
7	4	4	0*
	5	4	1
	6	4	2
	7	4	3
8	5	5	0*
	6	5	1
	7	5	2
	8	5	3

Note: * When the number of servers in reserve is 0, there is no high-availability advantage.

Cluster Membership Considerations

Be sure all network license clients that will use licenses bound to the cluster, and all servers that are members of the cluster, have all the servers of the cluster in their direct binding server list in order to exploit fully the high availability of licenses.

A server can be activated in only one cluster at any time. If you assign a server to a cluster and never activate it, or explicitly deactivate it, it can join a second cluster and be activated there. But in this case, the server cannot be activated in its original cluster, and no other server can be substituted in the original cluster. To reactivate the server in its original cluster, you must first deactivate it in the second cluster.

Do not disable remote administration on a server that is part of a cluster. If you do, you may have problems enrolling and removing licenses bound to the cluster.

Running the NetBIOS Socket Server

You need to run the NetBIOS socket server on any Windows NT machine that is configured to communicate with an OS/2 machine through NetBIOS. The OS/2 machine must run the NetBIOS socket server as well. NetBIOS is not supported on Windows 95 or Windows 98.

To start the NetBIOS socket server:

1. If your machine is configured as a server that provides licenses to OS/2 clients connected through NetBIOS, in i4ls.ini set:

   HasOS2Clients=yes

Verifying Connections

2. If your machine needs to bind an OS/2 License Use Management server connected through NetBIOS, on the Direct binding page of the configuration tool:
   a. In the **Name** field, add the name of the OS/2 server in uppercase.
   b. Select the **OS/2 server** check box.

Verifying Network Connections

License Use Runtime provides the i4tv tool to verify that license servers are running properly. For a detailed explanation of how to start the tool and how to use it, see Chapter 5, “License Use Runtime Commands” on page 127.

Chapter 3. Installing License Use Runtime

This chapter is intended for:

- The software vendor who needs to install License Use Runtime on a Windows NT, Windows NT Alpha, Windows Terminal Server, Windows 95, or Windows 98 machine for use in license-enabling a product
- The customer of a license-enabled product, if the vendor did not incorporate the License Use Runtime components into the product

This chapter explains:

- The License Use Runtime components you can select (“Selecting License Use Runtime Components” on page 48)
- License Use Runtime hardware and software requirements (“Hardware Requirements” on page 49 and “Software Requirements” on page 49)
- How to get the License Use Runtime code (“Obtaining the License Use Runtime Code” on page 50)
- How to install License Use Runtime interactively, using the installation GUI (“Installing License Use Runtime Interactively” on page 51)
- How to install License Use Runtime noninteractively, using a response file (“Installing License Use Runtime Noninteractively” on page 58)
- How to install License Use Runtime from a code server (“Installation Using a Code Server” on page 62)
- How to uninstall License Use Runtime components (“Uninstalling License Use Runtime” on page 63)
- How to install and uninstall LUM Java Client Support on Windows NT (“Installing LUM Java Client Support” on page 63)

Before You Install

Read this section to be sure you have everything you need and that you have made the necessary decisions before installing.

To install License Use Runtime, you must log in as a user with administrator privileges. This ensures that the path and registry information is updated during the installation. If you try to install License Use Runtime from another user ID, the installation program will issue the message:

Installation stopped. Only the user administrator can install License Use Runtime.

© Copyright IBM Corp. 1995, 1999

To determine whether License Use Runtime Version 4.5.5 is already installed on your machine, refer to the following files:

Communications component *c*:\ifor\ls\conf\comver
Runtime component *c*:\ifor\ls\conf\runver

where *c* is the drive letter.

Selecting License Use Runtime Components

License Use Runtime consists of these components:

Runtime
Communications
Documentation
LUM Java Client Support (which has a separate installation procedure)

Before you install on any machine, you must decide which components to install. Your selection of components depends on the role the machine will play in your licensing environment, as shown in Table 6. If your machine will play more than one role in the licensing environment, be sure you install the components required for all the roles. See "Setting Up Your Servers and Clients" on page 67 for information about configuring your machines to act in these roles.

Attention: If you are upgrading License Use Runtime from an earlier version, release, or modification level of License Use Runtime for which the Documentation component is installed, either uninstall the old Documentation component or install the new Documentation component. Otherwise, your License Use Runtime documentation will be incorrect for the installed product.

Table 6. Required Components by Machine Role

Role of the Machine	License Administration	Required Components
Nodelocked license server	Local administration only	Runtime
Nodelocked license server or Network license server or Network license client or Central registry license server	Local and remote administration	Runtime and Communications
Network license client	No administration	Communications
Network license client and Web server support for license-enabled Java applications and applets	No administration	Communications and LUM Java Client Support (Windows NT only)

Install the Documentation component on any machine to give you access to the online *Using License Use Runtime, Command Reference,* and *Message Reference* books, in .HTM format for viewing with a Web browser.

Hardware Requirements

The hardware requirements for the various Windows platforms are shown in Table 7.

Table 7. Hardware Requirements

Platform	Minimum		Recommended	
	Processor	**RAM**	**Processor**	**RAM**
Windows 95 Windows 98	33MHz 486	16 MB	Pentium 100MHz or higher	32 MB
Windows NT (x86) Windows Terminal Server (x86)	Pentium 100MHz	32 MB	Pentium 100MHz or higher	64 MB
Windows NT Alpha Windows Terminal Server for Alpha	Digital Alpha 433A	64 MB	Digital Alpha 433A	64 MB

Before installing License Use Runtime, allocate enough disk space for the components, as shown in Table 8. proc=apa.

Table 8. Disk Space Requirements

Component	Windows NT (x86) Windows Terminal Server (x86) Windows 95 Windows 98	Windows NT Alpha Windows Terminal Server for Alpha
Runtime	10 MB	5 MB
Communications	4 MB	6 MB
Documentation	2 MB	2 MB

Software Requirements

One of the following operating systems:

- Windows NT 4.0 (x86) with Service Pack 3, 4, or 5 (Service Pack 4 recommended)
- Windows NT 4.0 Alpha with Service Pack 3, 4, or 5 (Service Pack 4 recommended)
- Windows NT 4.0 Terminal Server Edition (x86) with Service Pack 3 or 4 (Service Pack 4 recommended)
- Windows NT 4.0 Terminal Server Edition for Alpha with Service Pack 3 or 4 (Service Pack 4 recommended)
- Windows 95
- Windows 98

For all supported Windows platforms:

Obtaining the License Use Runtime Code

- A device driver that complies with NDIS 4
- The latest level of the device driver. To determine the latest level of IBM device drivers, go to:

  http://www.networking.ibm.com
- The i4target executable. if License Use Runtime is not yet installed on the machine, you can download i4target from the License Use Management Web site:

  http://www.software.ibm.com/is/lum

  or the vendor of a license-enabled product can supply it to customers.

Depending on your operating system or operating environment, one or more of the transport protocols shown in Table 9:

Table 9. Transport Protocols

Platform	TCP/IP	NetBIOS	IPX
Windows NT 4.0 (x86)	√	√	√
Windows NT 4.0 Terminal Server Edition (x86)	√	√	√
Windows NT Version 4 Alpha	√		
Windows NT 4.0 Terminal Server Edition for Alpha	√		
Windows 95	√		√
Windows 98	√		

If the prerequisites are not met, the hardware-based target ID is not generated.

Protocol Configuration Requirements

After installation only the TCP/IP protocol is available. If you use the configuration tool command-line interface to configure, the tool handles protocol configuration. If you use the GUI to configure, then to use the protocols other than TCP/IP, edit the Registry and change the values of the following parameters from False to True:

HKEY_LOCAL_MACHINE\SOFTWARE\Gradient\DCE\Configuration\IPXSupport
HKEY_LOCAL_MACHINE\SOFTWARE\Gradient\DCE\Configuration\NetBIOSSupport

If you use NetBIOS as the transport protocol, be sure that the NetBEUI protocol is installed, and be sure you have done the following:

In the Services page of the Network window select the NetBIOS Interface and modify its properties so that the LAN adapter number of the Network Route starting with Nbf is set to 0.

Obtaining the License Use Runtime Code

You can get the License Use Runtime code:

From the product CD-ROM.

You can use the CD-ROM directly at the machine where you are installing License Use Runtime, or you can use it to put the License Use Runtime code images on a code server and then install your target machines from the code server (see "Installation Using a Code Server" on page 62).

By downloading License Use Runtime from the Web. Download the self-extracting EXE file from http://www.software.ibm.com/is/lum. Enter the name of the file as a command, using the /d option. This will:

- Extract the License Use Runtime code images, and
- Create the same directory structure that exists on the CD-ROM.

Deciding How to Install

Before you install, you need to decide:

Whether to install directly at the target machine or to put the License Use Runtime code images on a code server ("Installation Using a Code Server" on page 62) so that multiple users can use them for installation

Whether to install interactively, using the installation GUI ("Installing License Use Runtime Interactively") or noninteractively, using a response file ("Installing License Use Runtime Noninteractively" on page 58)

Installing License Use Runtime Interactively

To install License Use Runtime interactively, follow these steps:

1. In Windows NT, log on as Administrator.
2. Shut down any currently running programs, such as virus scanning programs.
3. Start the installation program:
   a. If you are installing License Use Runtime on a non-Windows Terminal Server machine, switch to the \runtime directory on the CD-ROM drive or to the directory on the disk drive to which you have downloaded the code, and enter:

      setup

      Alternatively, to save to a response file the decisions you make during installation, enter:

      setup -r

      For more information about response files, see "Saving a Response File During an Interactive Installation" on page 61.

   b. If you are installing License Use Runtime on a Windows Terminal Server machine:

      1. Select **Settings** → **Control Panel** → **Install** → **Next**.
      2. In the **Command line for installation program** field, type the drive identifier, path, and name of the installation program (**setup**).

3 Select the **Finish** pushbutton.

The Welcome window appears (Figure 15).

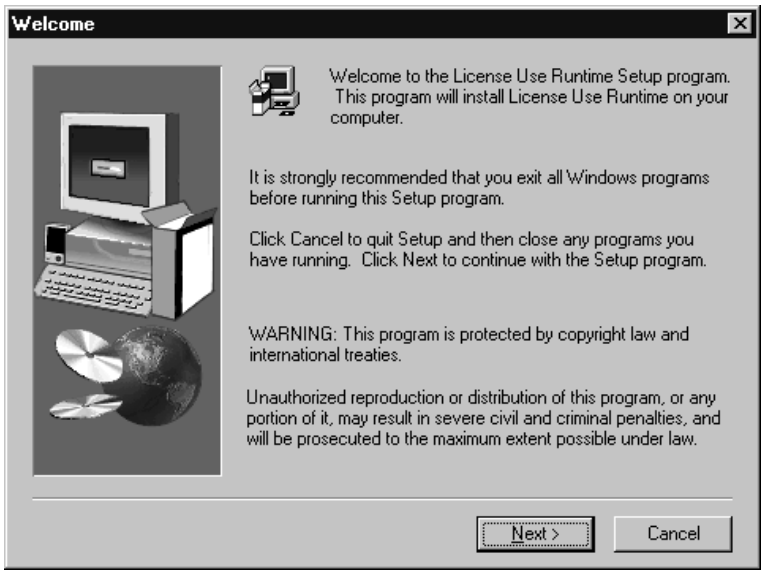


Figure 15. Welcome Window

4 In the Welcome window, select **Next**>.

5 The READ.ME file for License Use Runtime is displayed (Figure 16 on page 53).

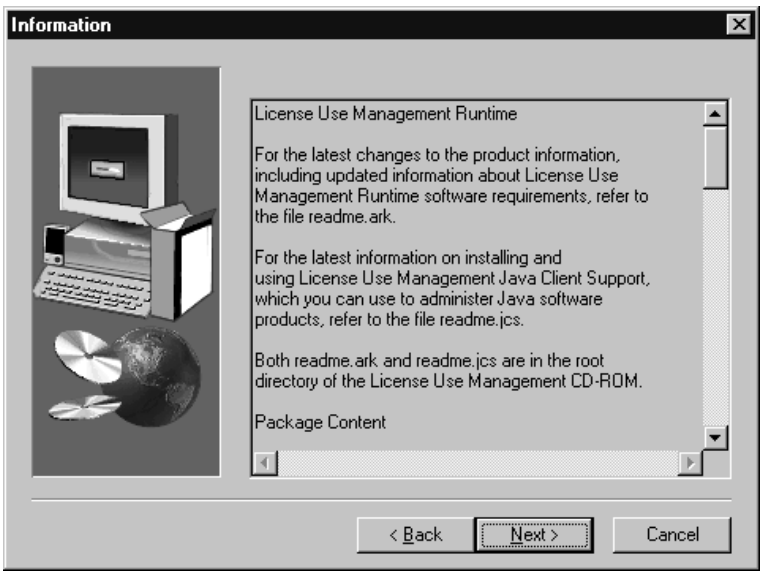


Figure 16. READ.ME Window. The text displayed on your monitor may differ from that shown here.

This file contains an overview of the installation procedure.

Availability of Java Client Support: The second paragraph of the displayed document refers to Java Client Support, which is available only under AIX, OS/2, Solaris, and Windows NT; it is not available on the Windows Terminal Server, Windows NT Alpha, Windows 95, or Windows 98 platforms.

When you have read the document, select **Next**>.

6 Select **Next**>. The Choose Destination Location window appears (Figure 17).

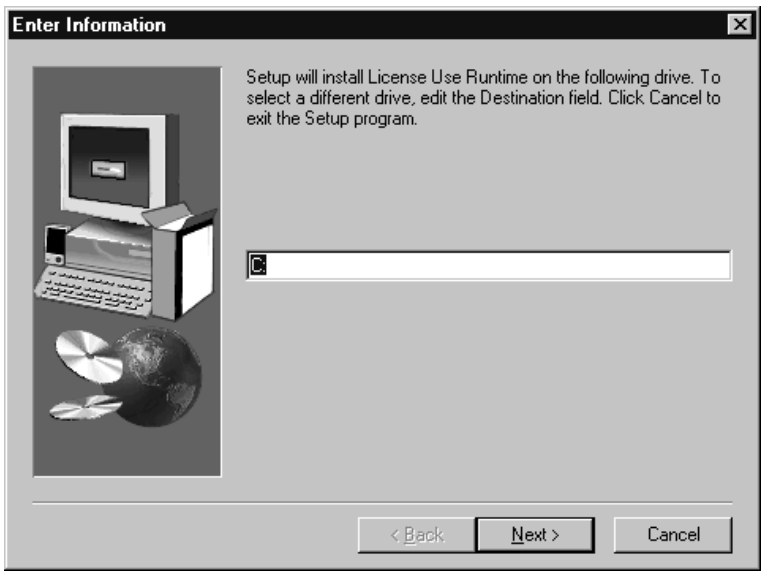


Figure 17. Choose Destination Location Window

Specify a destination drive for License Use Runtime files in the **Destination** field. The default is **C:**.

You can specify the drive only. The product is always installed in the ifor directory.

If Application Developer's Toolkit is installed on the machine, install License Use Runtime on the same drive.

7 Select **Next**>. The Select Components window appears (Figure 18).

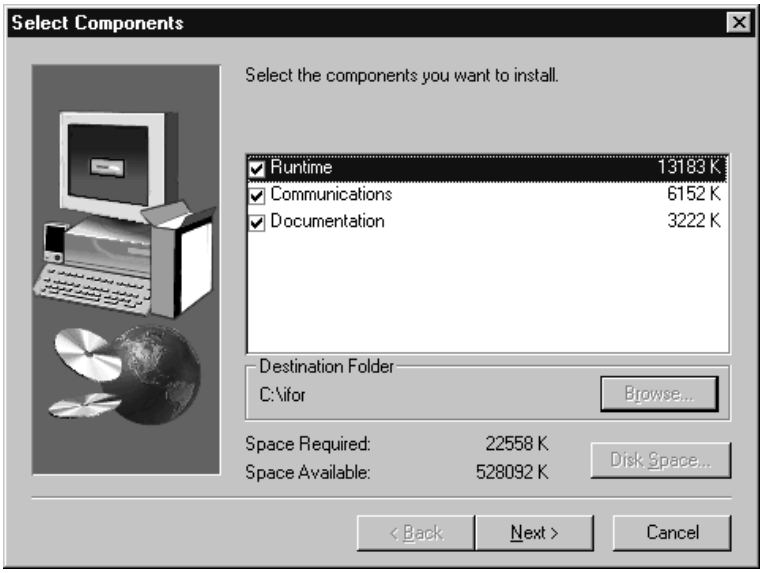


Figure 18. Select Components Window

8 Select the License Use Runtime components you want to install.

9 Select **Next**>. The Setup window appears, as shown in Figure 19.

Figure 19. Setup Window

License Use Runtime starts to copy files. You can monitor the progress by watching the progress bar.

If you get the message:

Unable to update PATH system variable.
Add the following path manually:
c:\ifor\WIN\BIN;c:\ifor\WIN\BIN*language*

where:

c Is the installation drive
language Is the installation language

Windows NT has not updated the PATH environment variable, because the sum of the length of the current environment variable plus the length of the License Use Runtime environment variable would exceed 1024 bytes. See Verifying and Updating the PATH Environment Variable for information about how to manually add the License Use Runtime path information.

10 The Setup Complete window is displayed. The contents of the window, and the actions you must take, depend on which operating system you are using:

If you have installed License Use Runtime on a Windows 95 or Windows 98 machine, the Setup Complete window looks like that shown in Figure 20.

Figure 20. Setup Complete Window

a Choose either of these options:

- Yes, I want to restart my computer now.
- No, I will restart my computer later.

In either case, you must restart your computer before you can use License Use Runtime.

b Select **Finish**.

License Use Runtime is installed on your computer, and a folder called License Use Runtime is displayed on the Program menu.

c Shut down and restart your machine.

If you have installed License Use Runtime on a Windows NT or Windows Terminal Server machine (x86 or Alpha), the Setup Complete window looks like that shown in Figure 21 on page 57.

Figure 21. Setup Complete Window

a Select **Finish**.

If you have installed License Use Runtime on a Windows NT machine, setup is now complete. You can start using License Use Runtime immediately. If you have installed License Use Runtime on a Windows Terminal Server machine, continue to step 10b.

b If you have installed License Use Runtime on a Windows Terminal Server machine (x86 or Alpha), run the **reglum.cmd** script, to register the License Use Runtime DLLs.

After installation has completed, you must configure License Use Runtime, as explained in Chapter 4, “Getting Started with License Use Runtime” on page 67.

Installing after an Uninstallation

If you uninstall License Use Runtime 4.*x.y*, your existing license databases are not deleted. They remain in the directory ifor\ls\conf. If you then install License Use Runtime Version 4.5.5, the installation process renames the databases to have an extension of .old. If you want to use the databases with License Use Runtime Version 4.5.5, enter the following sequence of commands:

Installing Noninteractively

```
i4cfg -stop
cd ifor\ls\conf
copy crpdb.dat.old crpdb.dat
copy crpdb.idx.old crpdb.idx
copy licdb.dat.old licdb.dat
copy licdb.idx.old licdb.idx
copy llmdb.dat.old llmdb.dat
copy llmdb.idx.old llmdb.idx
i4mig -r
i4mig -r -c
i4mig -r -n
```

Installing License Use Runtime Noninteractively

You can use the **setup** command to install License Use Runtime without being prompted.

You can create a response file in advance, as explained in "Creating a Response File" on page 60. The response file contains your responses to the decisions that are made during an interactive installation. You need to be present only to initiate the installation and to restart your workstation, if necessary, when the installation is complete. No questions are asked while the installation is running.

If you get the message:

Unable to update Path system variable.
Add the following path manually:
c:\ifor\WIN\BIN;c:\ifor\WIN\BIN*language*

where:

c Is the installation drive
language Is the installation language

Windows NT has not updated the PATH environment variable, because the sum of the length of the current environment variable plus the length of the License Use Runtime environment variable would exceed 1024 bytes. See Verifying and Updating the PATH Environment Variable for information about how to manually add the License Use Runtime path information.

1. From the CD-ROM drive or the directory where you have downloaded the code, switch to the \runtime directory.
2. Run the installation program:

   If you are installing License Use Runtime on a non-Windows Terminal Server machine, from the CD-ROM drive or the directory to which you have downloaded the code, switch to the \runtime directory and enter the **setup -s** command, according to the syntax described in "Using the Installation Command" on page 59.

   If you are installing License Use Runtime on a Windows Terminal Server machine:

a. Select **Settings** → **Control Panel** → **Install** → **Next**.

b. In the **Command line for installation program** field, type the drive identifier, path, and name of the installation program (**setup -s**), according to the syntax described in "Using the Installation Command."

c. Select the **Finish** pushbutton.

3. If you installed the Runtime component under Windows 95 or Windows 98, shut down and restart the machine.

4. If you installed License Use Runtime on a Windows Terminal Server machine, run the **reglum.cmd** script to register the License Use Runtime dynamic link libraries.

5. Follow the instructions in Chapter 4, "Getting Started with License Use Runtime" on page 67.

Using the Installation Command

The following syntax shows the parameters accepted by the **setup** installation command:

setup -s -f1<response_file> **-f2**<log_file>

where:

-s
Specify this to execute a silent installation.

If you omit this parameter, the installation starts interactively with all windows displayed.

-f1<**response_file**>
The full path and name of the response file containing the installation parameters. It is mandatory if you specify the -s parameter.

-f2<**log_file**>
The full path and name of the log file created by the silent installation. If you do not specify it, by default a SETUP.LOG log file is created and stored in the same directory as that of SETUP.EXE. The SETUP.LOG file for a successful silent installation of the *Examp* product is shown below.

Creating a Response File

```
[InstallShield Silent]
Version=v3. .
File=Log File

[Application]
Name=Examp
Version=3. .

Company=Stirling

[ResponseResult]
ResultCode=
```

To check if the installation was successful, check ResultCode in SETUP.LOG. If ResultCode is 0, the installation was successful; if it has any other value, the installation was unsuccessful.

Note that for -f1 and -f2, you must not leave a blank between the parameter and the path you specify after it.

After the installation is complete:

1. If you installed License Use Runtime on a Windows NT machine (x86 or Alpha), go to step 2. Otherwise:

   If you installed License Use Runtime under Windows 95 or Windows 98, shut down and restart your machine.

   If you installed License Use Runtime on a Windows Terminal Server, run the **reglum.cmd** script to register License Use Runtime dynamic link libraries.

2. Configure your machine as explained in Chapter 4, “Getting Started with License Use Runtime” on page 67.

Creating a Response File

This section explains two ways to create a response file: editing the sample response file provided with License Use Runtime, and saving the results of an interactive installation to a response file for subsequent use on other machines.

Editing the Sample Response File

The response file is divided into several sections, each started by a name included in brackets. There are two types of line in each section:

Comment lines Have a semicolon as the first nonblank character on the line.

Response lines Determine the options and configurations to install on the target system. Response lines have the following syntax:

keyword=value

Keywords are case-sensitive.

Creating a Response File

An example of how to install License Use Runtime using a response file follows:

```
setup -s -f1c:\mydir\setup.iss
```

where **setup.iss** is the response file. This sample file is included in the License Use Runtime package in the same directory as the setup.exe command.

Use this response file to customize License Use Runtime installation. You can change the values of the keywords identified by the string:

| ;

They are:

| **szText**
The target drive for License Use Runtime. Example:

| szText=C:

szDir
| The target drive for License Use Runtime. Repeat here the same value specified
| for szText.

Component-count
The number of components you want to install. Valid values are from 1 to 3.

Component-*n*
The names of the components you want to install. Do not use quotation marks around the component name. Specify as many components as the number you specify in Component-count. For example, if you want to install all three License Use Runtime components, set Component-count to 3, and then:

```
Component- =Runtime
Component-1=Communications
Component-2=Documentation
```

If you want to install only the Communications and Documentation components, set Component-count to 2, and then:

```
Component- =Communications
Component-1=Documentation
```

Do not modify the other keyword values.

Saving a Response File During an Interactive Installation

When you do an interactive installation, you can start the installation by entering the command **setup -r**. If you do so, the installation process saves the decisions you make in a response file called **setup.iss** in the WINDOWS directory. You can then use the response file to replicate the same installation on other machines.

Using a Code Server

Verifying and Updating the PATH Environment Variable

In Windows NT, check whether the PATH environment variable contains the following information:

c:\ifor\WIN\BIN;*c*:\ifor\WIN\BIN*language*

where:

c	Is the installation drive
language	Is the installation language; for example, EN_US for US English

If the License Use Runtime path information has not been added, update the PATH environment variable manually, as follows:

1. Go to Settings → Control Panel → System → Environment.
2. If necessary, delete unused path information. The License Use Runtime path information requires 38 bytes. Remember to add any necessary delimiting semicolon to the end of the original PATH statement.
3. Add to the end of the PATH statement:

   ;*c*:\ifor\WIN\BIN;*c*:\ifor\WIN\BIN*language*
4. Press the **Set** or **OK** pushbutton.

Installation Using a Code Server

A *code server* is a workstation that contains the files (CD-ROM images) to be used during the installation.

To install License Use Runtime on a non-Windows Terminal Server machine, connect to the code images directory of the code server. Then switch to the directory x:\runtime, where x: is the drive letter of the redirected drive and \runtime is the License Use Runtime directory name, and type SETUP. Type the command without parameters for an interactive installation, or with parameters as indicated in the section “Using the Installation Command” on page 59 for a noninteractive installation.

To install License Use Runtime on a Windows Terminal Server machine:

1. Select **Settings** → **Control Panel** → **Install** → **Next**.
2. In the **Command line for installation program** field, type the drive identifier, path, and name of the installation program (**setup**).
3. Select the **Finish** pushbutton.

Proceed with the installation as explained in “Installing License Use Runtime Interactively” on page 51 or “Installing License Use Runtime Noninteractively” on page 58. When you have successfully installed License Use Runtime, a message indicates that the installation is complete.

If you have installed the Runtime component of License Use Runtime under Windows 95 or Windows 98, shut down and restart the machine.

If you installed License Use Runtime on a Windows Terminal Server machine, run the **reglum** script to register the License Use Runtime dynamic link libraries.

For non-Windows Terminal Server and for Windows Terminal Server installation, follow the instructions in Chapter 4, “Getting Started with License Use Runtime” on page 67.

Uninstalling License Use Runtime

You can remove License Use Runtime from your computer only in its entirety. You cannot selectively uninstall components.

To remove License Use Runtime from your machine, go to Settings → Control Panel → Add/Remove Programs and select License Use Runtime.

Note, however, that the uninstallation process does not remove configuration files or other files created after installation. To remove such files, after the uninstallation process has finished, delete the ifor directory and all its contents.

Installing LUM Java Client Support

LUM Java Client Support is a separately installable component, which you can install only on Windows NT.

Before You Install

This section presents disk space and software requirements.

Disk Space Requirements

LUM Java Client Support requires approximately 200 KB of disk space.

Software Requirements

The following software is required:

One of the following Web servers:

- Lotus Domino Go Webserver 4.6.*x*
- Netscape Enterprise Server 3.0.1 or 3.5.1; 3.5.1 is recommended
- Netscape FastTrack Server 2.0.1 or 3.0.1
- Microsoft Internet Information Server 2.*x*, 3.*x*, or 4.0

Installing LUM Java Client Support

Either Java Development Kit (JDK) 1.1.4 or 1.1.6, or Java Runtime Environment (JRE) 1.1.4 or 1.1.6. JDK 1.1.6 is shipped with the IBM WebSphere Application Server.

IBM WebSphere Application Server 1.1 or 2.0

The Communications component of License Use Runtime Version 4.5.5

TCP/IP

Obtaining LUM Java Client Support Code

You can get the LUM Java Client Support code:

From the product CD-ROM.

By downloading LUM Java Client Support from the Web. Download the self-extracting EXE file from http://www.software.ibm.com/is/lum. Enter the name of the file as a command, using the /d option. This will:

- Extract LUM Java Client Support code images, and
- Create the proper directory structure.

With an enabled application, if the vendor chose to redistribute the LUM Java Client Support code.

Installing LUM Java Client Support

To install LUM Java Client Support interactively, follow these steps:

1. In Windows NT, log on as Administrator.
2. From the CD-ROM drive or the directory where you have downloaded the code, switch to the \servlet directory and type **setup**.
3. Follow the panels that are displayed. LUM Java Client Support is installed in the directory APPSERVER_ROOT\Web\LUM\classes\com\ibm\licUseMgmt, where APPSERVER_ROOT\Web\ is a WebSphere subdirectory. One file is also stored in ifor\win\bin.

Uninstalling LUM Java Client Support

To remove LUM Java Client Support from your machine, go to Settings → Control Panel → Add/Remove Programs and select License Use Management LUM Java Client Support.

Chapter 4. Getting Started with License Use Runtime

The scenarios in this chapter describe how to set up your License Use Runtime environment and how to manage both nodelocked and network licensed products.

Setting Up Your Servers and Clients

After you install License Use Runtime, you must configure the program on each machine.

You configure using a configuration tool. On platforms other than Windows NT Alpha and Windows Terminal Server Alpha, this tool has a GUI and a command line interface. On the Windows NT Alpha and Windows Terminal Server Alpha platforms it has only a command line interface. The information you provide is saved in a configuration file. When you start License Use Runtime, it uses the information in this file to direct the behavior of the local system in the licensing environment. See Appendix A, “License Use Runtime Configuration File” on page 199 for reference information on the configuration file. The configuration details depend on the role the machine will play in your licensing environment and the types of licenses you need to handle.

Configuring to Handle Nodelocked Licenses

To handle only products with nodelocked licenses, you need only configure the machine as a nodelocked license server (“Scenario 1: Configuring a Standalone Nodelocked License Server” on page 73) and have the nodelocked license server up and running. Every machine is automatically configured as a nodelocked license server when License Use Runtime is installed. You do not have to do any configuration unless you want to change the default configuration.

If a machine configured as a nodelocked license server is in a network, instances of the Basic License Tool on other machines can administer licenses on the nodelocked license server. With a little additional configuration (“Scenario 2: Configuring a Nodelocked License Server in a Network” on page 75), you can run the Basic License Tool on the nodelocked license server machine and administer licenses stored on remote network license servers, nodelocked license servers, and the central registry.

Configuring to Handle Network Licenses

To handle products with network licenses, you must configure at least one network license server (“Scenario 3: Configuring a Network License Server” on page 79) and configure each client as a network license client (“Scenario 4: Configuring a Network License Client” on page 84). If you use products with customer-managed or reservable licenses, you must also configure one server as the central registry (“Scenario 5: Configuring the Central Registry License Server” on page 88).

© Copyright IBM Corp. 1995, 1999

Configuration

Determining the Configuration Required

You can configure a machine to play more than one role in your licensing environment. For example, if you configure the same machine as a nodelocked license server, a network license server, and the central registry license server, that machine can handle all types of licensed products.

If you know the types of licenses used by the license-enabled products in your environment, Table 10 on page 69 will help you to determine how to configure:

The license servers for the application
The machines that will request licenses for the application

Consult the documentation of the license-enabled products for the license types and other information about the enabling that might affect your configuration. In any case, the enrollment certificate file that you receive from the vendor shows the type of license.

Table 10. Configuration Required to Support All Types of Licenses

License Type	License Requester	License Server
Simple Nodelocked (Non-Runtime-Based Enabling)[1]	License-Enabled Application	None
Simple Nodelocked (Runtime-Based Enabling)[1]	License-Enabled Application	Nodelocked License Server on the Same Machine as the Application
Use-Once Nodelocked	License-Enabled Application	Nodelocked License Server on the Same Machine as the Application
Concurrent Nodelocked	License-Enabled Application	Nodelocked License Server on the Same Machine as the Application
Per-Server	License-Enabled Application	Nodelocked License Server on the Same Machine as the Application
Use-Once	Network License Client	Network License Server[2] + Central Registry License Server[3]
Concurrent	Network License Client	Network License Server[2] + Central Registry License Server[3]
Per-Seat	Network License Client + Nodelocked License Server	Central Registry License Server[3]
Reservable	Network License Client + Nodelocked License Server[4]	Network License Server[2] + Central Registry License Server[3]

Notes:

1. If the vendor enabled the product with simple nodelocked licenses and delivered the licenses to you in a compound network password, you must also:

   Configure a network license server, where you will install the compound password, and

   Configure the local machine (where the nodelocked license is to be installed) as a network license client of that server.

2. You can configure one or more network license servers.
3. You can configure only one central registry license server. For customer-managed use products, the central registry is required. It enables you to update the count of product licences, implement the hard stop or soft stop policy, or track the high-water mark. Note that because per-seat licenses are always customer-managed, they require the central registry. For reservable licenses, the central registry is required if you want to be able to reserve licenses for specific users.
4. For reservable licenses, the nodelocked license server is required for the end user to get a reserved license.

Configuration Scenarios

Before You Configure

Before you begin the configuration process, for every machine you are going to configure, you need to decide which roles it will play and how you plan to set up direct binding. You might also decide to override some of the configuration defaults. The actions you can take during configuration are summarized in Table 11. Check the table for all the roles your machine will play, and make all the indicated decisions before you start configuration.

Table 11. Configuration Options

Configuration Options	Nodelocked License Server	Network License Client	Network License Server	Central Registry License Server
Customize selection of information logged or accept default? ("Customizing Log Information")	√		√	√
Change log path or accept default? ("Customizing Log Information")	√		√	√
Start license servers at system startup (default is no)? ("Automatically Starting License Servers" on page 71)	√		√	√
Disable remote administration of network license servers (default is no)? ("Disabling Remote Administration" on page 71)			√	
Disable remote administration of nodelocked license servers (default is no)? ("Disabling Remote Administration" on page 71)	√			
Set up direct binding ("Configuring Direct Binding" on page 71)				
Prepare a list of nodelocked license servers for remote administration	√1	√	√	√
Have your direct binding servers list ready	√1	√	√	√
Know which machine is the central registry	√1	√	√	√
Change default direct binding ports?	√1	√	√	√

Notes:

1. Applicable if you plan to run the Basic License Tool and administer licenses on other machines from this nodelocked license server.

Customizing Log Information

For any license server, you can customize the selection of events that are logged, and the location of the log file. Note that if you want to change the location of the log from the default to a path you choose, the directory you specify must already exist. Otherwise, you will lose the logging function.

The following events can be logged:

All events
Includes all the events in the list.

Errors
Describes server errors that do not stop the server, but return a status code and a message. This is logged by default.

License timeout
Tells you that the server has canceled the request for a license because the check period expired. This is not logged by default.

License wait
Tells you when a license request cannot be satisfied because no licenses are available, and the user is added to a queue. This is not logged by default.

License checkin
Tells you when a licensed product has sent a check-in call to the server to notify that the product is running. This is not logged by default.

License grant/release
Tells you when a license was granted or released. This is not logged by default.

Vendor added/deleted
Tells you when a product of a new vendor was registered or deleted. This is logged by default.

Vendor messages
Provides the log messages the vendor inserted in the enabled product. This is logged by default.

Product added/deleted
Tells you when a new product was registered or deleted. This is logged by default.

Server start/stop
Logs the successful start or stop of the license server. This is not logged by default.

Automatically Starting License Servers

During configuration of any license server, you can specify that license servers should start automatically when you start the machine. Otherwise, you must remember to start the services manually after configuration and before using the Basic License Tool or any enabled applications.

Disabling Remote Administration

When you configure a network or nodelocked license server, you can specify that licenses stored on that server cannot be administered from any other license server.

Configuring Direct Binding

When you configure a network license server, network license client, or central registry license server that is to be part of a direct binding environment, you must have your direct binding servers list ready. (See “Planning Direct Binding” on page 37.)

When you configure a nodelocked license server, network license server, or central registry license server that is to be part of a direct binding environment, you must also

have ready a list of nodelocked license servers whose licenses you want to administer remotely from this machine.

You will enter the hostnames or network addresses of all the servers in the list (other than the nodelocked license server on the local machine, which is added to the list automatically). You will also designate which server, if any, is the central registry.

When you configure the servers in the direct binding servers list, and the clients that will use them, be sure you define exactly the same set of servers on each.

You can change the default port numbers for nodelocked license servers, network license servers, and the central registry license server. Do not change the defaults unless they are already in use by other applications.

If you are running License Use Management Java Client Support on the same machine and want to change the direct binding list:

1. Stop License Use Management and the Web server.
2. Change the direct binding list.
3. Restart License Use Management and the Web server.

Using the Configuration Tool GUI

On platforms other than Windows NT Alpha, you can use the License Use Runtime GUI to configure your machine. If you use Windows NT Alpha, see "Using the Configuration Tool Command-Line Interface" on page 73.

In the License Use Runtime folder, click on **Configuration Tool**.

Alternatively, start the Configuration Tool by entering the `i4cfg` command with no parameters.

A configuration tool notebook appears. Follow the steps of one or more of the configuration scenarios in this section, depending on the role of your machine in your licensing environment.

Each of the configuration scenarios in this chapter shows how to configure a machine to play only one role in the licensing environment. If the machine is to play two or more roles, check *all* the applicable roles on the **Configure As** page of the configuration tool notebook.

If you configure a specific page of the notebook more than once, the most recent configuration takes effect.

Using the Configuration Tool Command-Line Interface

You can use the i4cfg command to accomplish the same configuration tasks explained in the scenarios in this chapter. At the end of each scenario, a section named "Command-Line Equivalent" shows the i4cfg commands that correspond to the GUI scenario.

See "i4cfg - Configuration Tool" on page 153 for details about the i4cfg command.

Scenario 1: Configuring a Standalone Nodelocked License Server

Use this scenario to configure a nodelocked license server without setting up any network connections to other License Use Runtime servers. Use "Scenario 2: Configuring a Nodelocked License Server in a Network" on page 75 if you want to run the Basic License Tool and administer licenses on other servers from this machine,

A nodelocked license server can use licenses of the types shown in Table 10 on page 69. By default, every machine is configured as a nodelocked license server. You need to perform this configuration only if you want to change the default configuration.

To configure a machine as a standalone nodelocked license server after installation, perform the following steps:

1. Double-click on **Configuration Tool** in the License Use Runtime folder. The Configuration Tool notebook is displayed.
2. On the **Configure As** page, select **Nodelocked License Server** and **Advanced Configuration.** The notebook is shown in Figure 22 on page 74.

Configuring a Standalone Nodelocked License Server

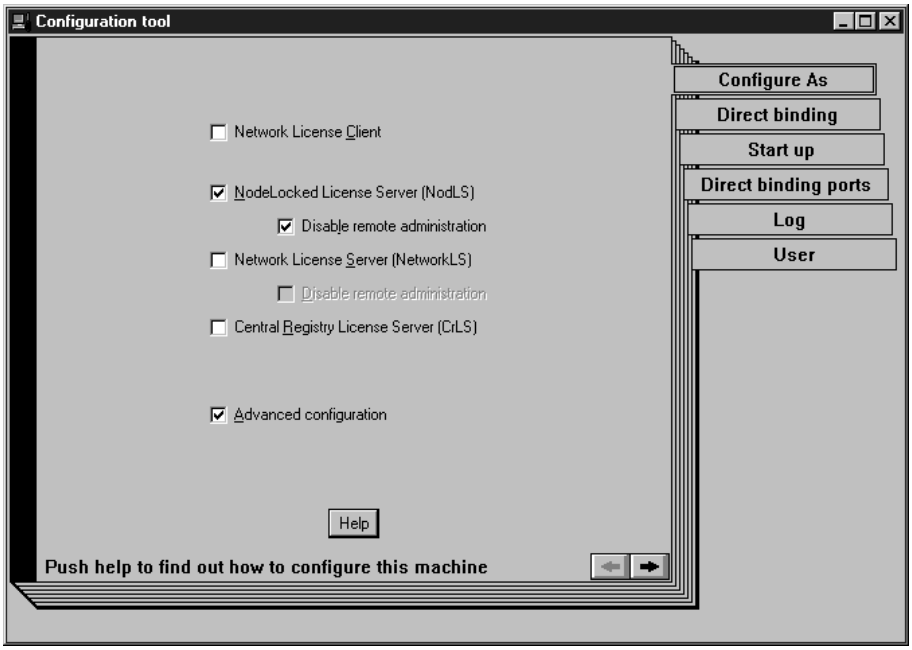


Figure 22. Configuration Tool Notebook - Standalone Nodelocked License Server

For the type of configuration you are doing, ignore the **Direct Binding, Direct Binding Ports,** and **User** pages (if they appear in the notebook), and the **Disable Remote Administration** check box.

3 On the **Start up** page, select **Start services at system startup** to start the nodelocked license server when you power on the machine. (This option is available on Windows NT only.)

4 On the **Log** page, select the events you want to be logged and specify where you want the log to be kept, as shown in Figure 23 on page 75. If you change the location of the log from the default directory to a directory of your own choice, that directory must already exist. Otherwise, the logging function will be lost.

The logged events are stored in the files *llmlgnn_*, where *nn* assumes values from 00 to 99. When a file is full, according to a maximum number of lines fixed in the configuration file, a new log file is started.

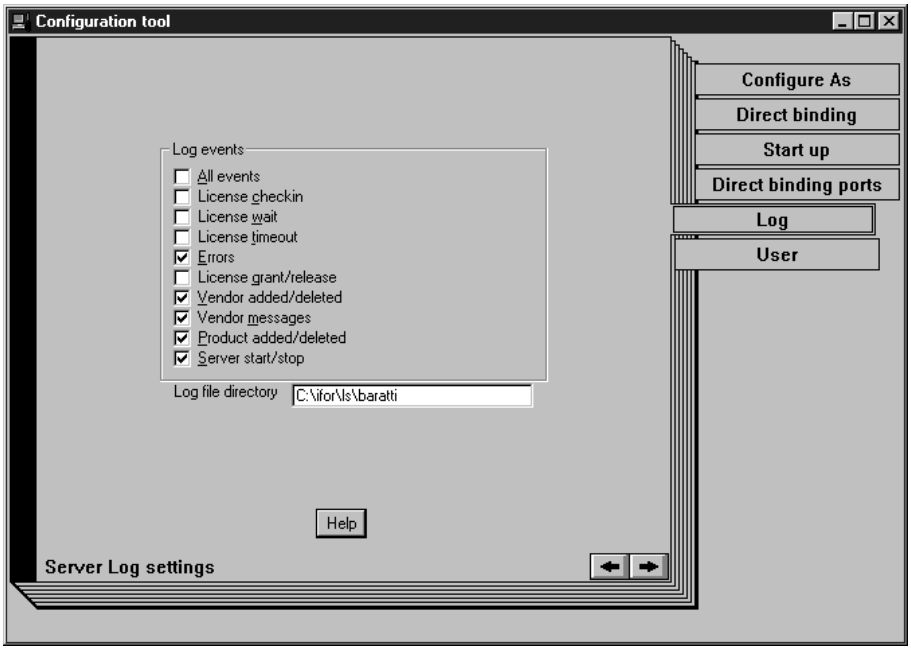


Figure 23. Configuration Tool Notebook - Log Page

5 Select **Close** from the system menu at the upper left corner of the notebook. A message is displayed to confirm that you are ready to save your choices.

Command-Line Equivalent

To configure the standalone nodelocked license server:

```
i4cfg -a n -S a -e evmps -l c:\ifor\ls\baratti
```

Scenario 2: Configuring a Nodelocked License Server in a Network

Use this scenario to configure a nodelocked license server, making it part of a direct binding server list. Configuring in this way, you can:

Run the Basic License Tool and administer licenses on other license servers from this machine

Allow instances of the Basic License Tool on other machines to administer nodelocked licenses on this machine

A nodelocked license server can use licenses of the types shown in Table 10 on page 69.

To configure a machine as a nodelocked license server with remote administration, after installation, perform the following steps:

1 Double-click on **Configuration Tool** in the License Use Runtime folder. The Configuration Tool notebook is displayed.

Configuring a Nodelocked License Server in a Network

2 On the **Configure As** page, select **Nodelocked License Server** and **Advanced Configuration.**

3 Leave the **Disable remote administration** box unchecked, to enable instances of the Basic License Tool on other machines to administer nodelocked licenses on this machine.

The completed **Configure As** page is shown in Figure 24.

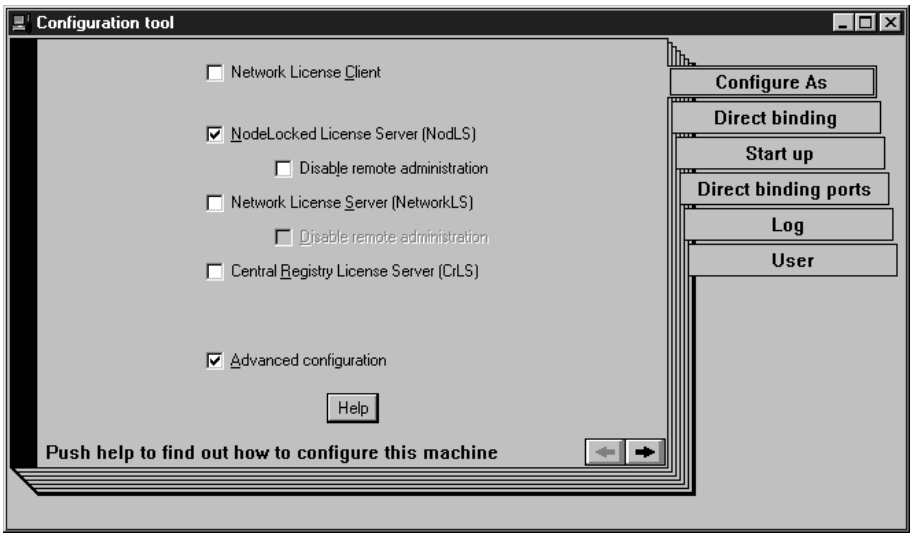


Figure 24. Configuration Tool Notebook - Nodelocked License Server in a Network

4 Complete the **Start up** and **Log** pages, as explained in "Scenario 1: Configuring a Standalone Nodelocked License Server" on page 73, if you want to change the defaults.

5 Select the **User** tab. The **User** section is displayed, as shown in Figure 25 on page 77.

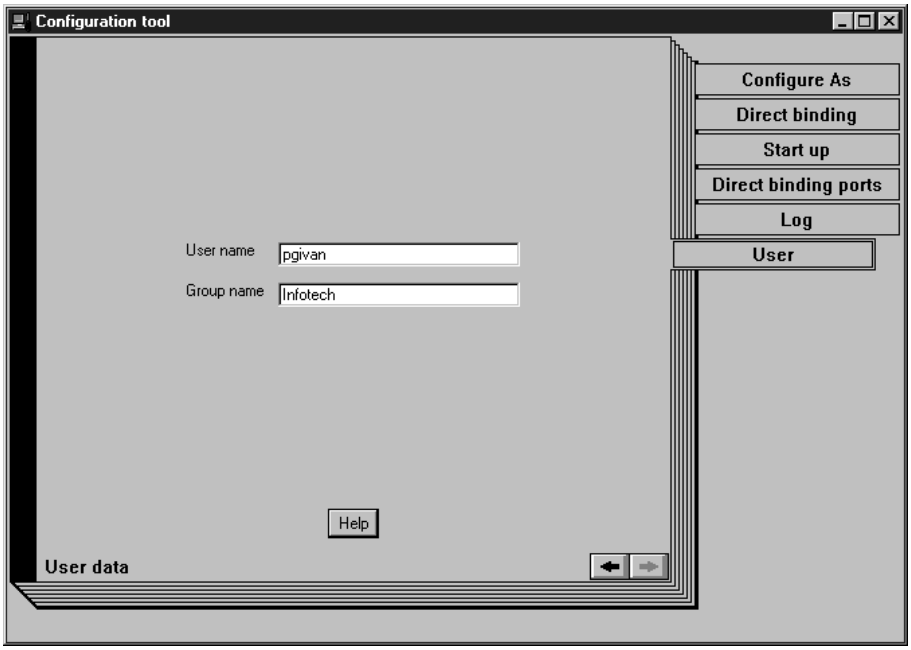


Figure 25. Configuration Tool Notebook - User Section

The user name is the logon user ID in effect when you do the configuration. Enter a group name if you want to change the default.

Note that the License Use Runtime group name is not the same as the Windows NT group name.

License Use Runtime treats the user name as the name of the user who requests and uses licenses on this machine. The user name is displayed as part of various reports about license usage. It is also used to reserve reservable licenses for a specific user. The group name is used to reserve reservable licenses for members of a specific group.

Use the remaining pages of the notebook to establish remote connections between this server and other License Use Runtime license servers. From this machine, you will be able to use the Basic License Tool to administer licenses on those servers.

6 Select the **Direct binding** tab.

The Direct binding section is displayed, as shown in Figure 26 on page 78.

Configuring a Nodelocked License Server in a Network

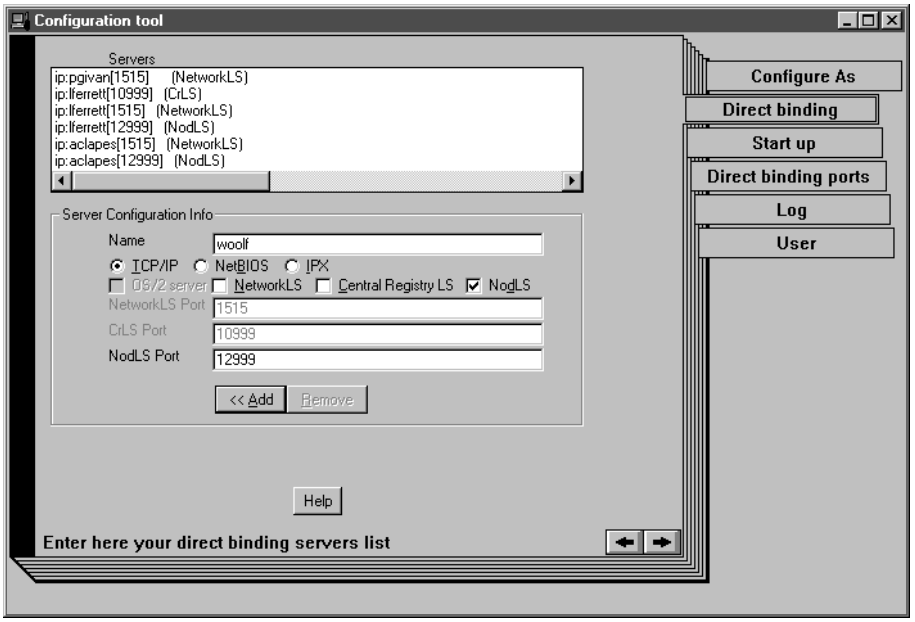


Figure 26. Configuration Tool Notebook - Direct Binding Section

On this page you specify all the network license servers and nodelocked license servers whose licenses you will administer remotely from this machine. Do not include this nodelocked license server itself. If the central registry license server, a network license server, and a nodelocked license server run on the same machine, include all the servers whose licenses you want to administer.

In this example, the administrator performs the following steps for each server:

a Select the **TCP/IP** radio button. On the Windows NT Alpha platform, this is the only choice.

b In the **Name** field, enter the TCP/IP host name of the machine you are adding to the **Servers** list. Note that the server name is case-sensitive.

c Check **NodLS, NetworkLS**, or **Central Registry LS**, or any combination, depending on the roles the machine plays in the network.

d Leave the default values in the **NetworkLS Port, CrLS Port**, and **NodLS Port** fields.

For platforms other than Windows NT Alpha, Windows 95, or Windows 98, if NetBIOS is selected, the default values are, respectively, 115, 109, and 129. If you change them, use values between 32 and 254.

If NetBIOS or IPX is selected, the protocol must be configured as explained under “Protocol Configuration Requirements” on page 50.

e Select the <<**Add** push button to add the server to the **Servers** list.

7 If you want to change the default direct binding port numbers, use the **Direct binding ports** page. If you set the direct binding ports for a particular protocol to 0, the server does not listen on that protocol.

8 To run the configured processes, see “Starting and Listing Your Processes” on page 90.

Now the workstation can be used as a nodelocked license server. The administrator can run the Basic License Tool on the workstation and can administer licenses on remote License Use Runtime servers.

Command-Line Equivalent

To configure the nodelocked license server in a network:

```
i4cfg -a n -S a,n -e evmps -l c:\ifor\ls\baratti
-b "'network ip:pgivan ip:lferrett ip:aclapes'
'nodelocked ip:lferrett ip:aclapes ip:woolf' 'registry ip:lferrett'"
```

Scenario 3: Configuring a Network License Server

This scenario shows how the administrator configures License Use Runtime as a network license server (in this example, *pgivan*), making it part of a direct binding server list.

When you configure a machine as a network license server, you can use licenses of the types shown in Table 10 on page 69. You can also use the Basic License Tool to administer licenses on remote license servers in the network.

To configure the network license server, after installation of License Use Runtime:

1 Double-click on **Configuration Tool** in the License Use Runtime folder. The Configuration Tool notebook is displayed.

2 On the **Configure As** page, select **Network License Server**. Note that **Network License Client** is then automatically checked. Leave **Disable Remote Administration** unchecked. Check **Advanced Configuration**, which adds the **Direct binding ports**, **Log**, and **User** pages to the notebook.

The **Configure As** page is shown in Figure 27 on page 80.

Configuring a Network License Server

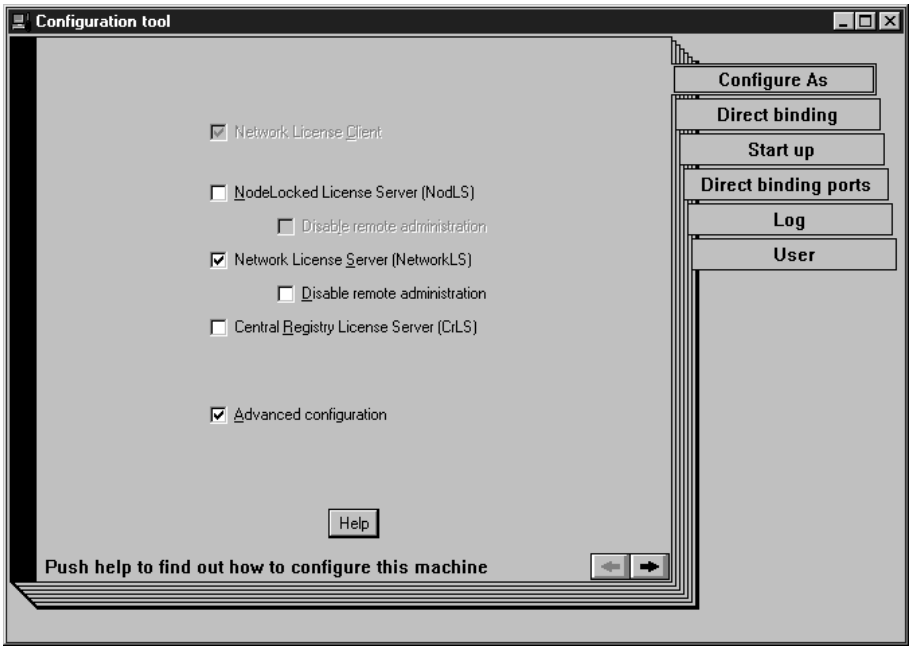


Figure 27. Configuration Tool Notebook - Network License Server

3 On the **Start up** page, select **Start services at system startup** to start the network license server when you power on the machine. (This option is available on Windows NT only.)

4 On the **Log** page, select the events you want to be logged and specify where you want the log to be kept, as shown in Figure 28 on page 81.

The logged events are stored in the files *logdbnn_*, where *nn* assumes values from 00 to 99. When a file is full, according to a maximum number of lines fixed in the configuration file, a new one is started.

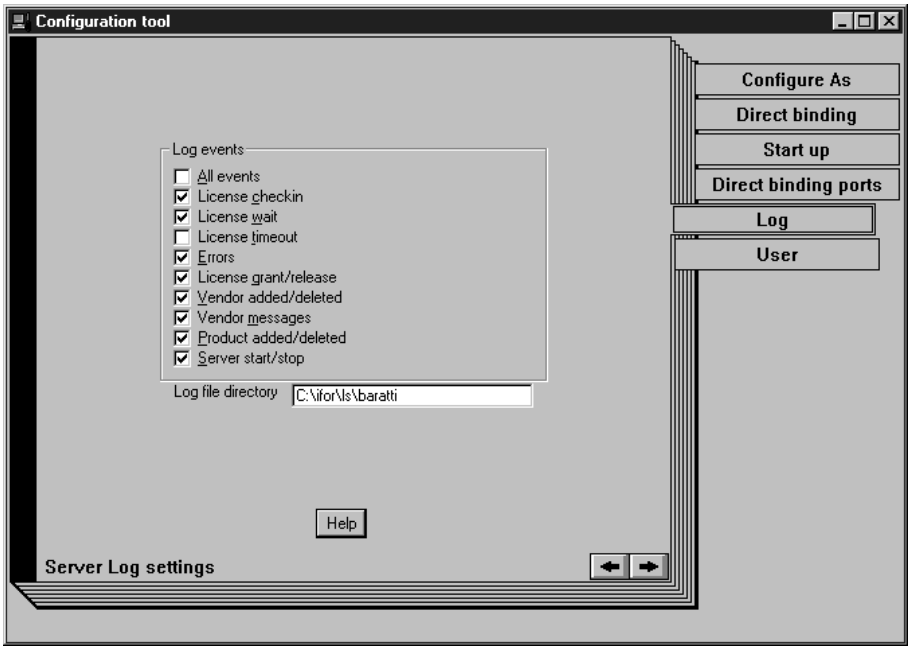


Figure 28. Configuration Tool Notebook - Log Page

5 Select the **Direct binding** tab.

The Direct binding section is displayed, as shown in Figure 29 on page 82.

Configuring a Network License Server

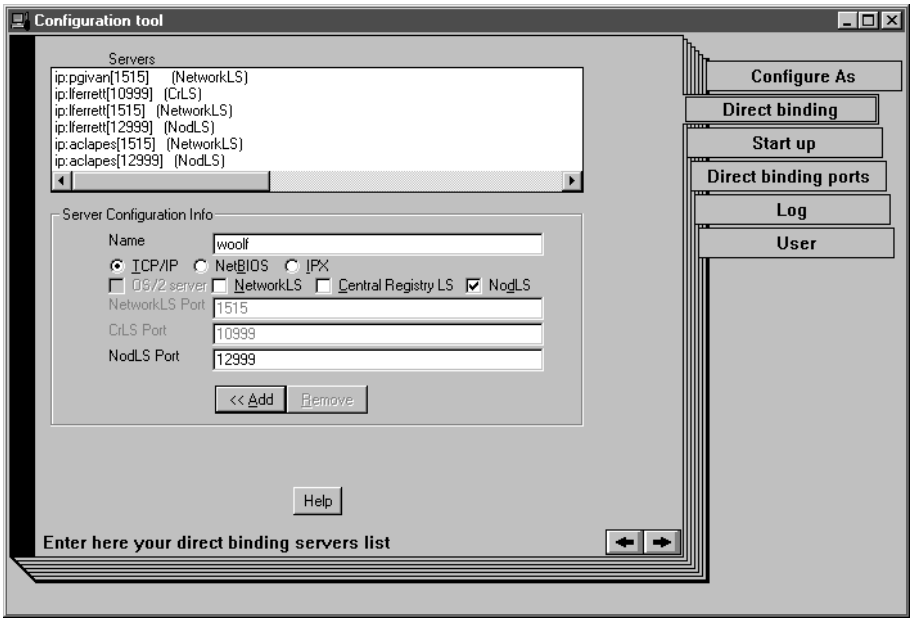


Figure 29. Configuration Tool Notebook - Direct Binding Section

On this page you specify all the network license servers in the direct binding servers list. You must include this network license server that you are configuring, and the central registry license server, if any. If the central registry license server and a network license server run on the same machine, be sure you include *both* servers in the list.

As you configure the servers in the direct binding servers list, be sure you define exactly the same set of servers on each.

In addition to specifying the direct binding servers list, use this page to specify any remote nodelocked license servers whose licenses you want to administer from this machine. Do not include the nodelocked license server on this machine. (You can administer local nodelocked licenses automatically, without specifying direct binding.)

In this example, the administrator performs the following steps for each server:

a Select the **TCP/IP** radio button. On the Windows NT Alpha platform, this is the only choice.

b In the **Name** field, enter the TCP/IP host name of the machine you are adding to the **Servers** list. Note that the server name is case-sensitive.

c If the server being added is a network license server, leave only the **NetworkLS** check box selected.

d Check **NodLS, NetworkLS,** or **Central Registry LS,** or any combination, depending on the roles the machine plays in the network.

e Leave the default values in the **NetworkLS Port, CrLS Port**, and **NodLS Port** fields.

For platforms other than Windows NT Alpha, Windows 95, or Windows 98, if NetBIOS is selected, the default values are, respectively, 115, 109, and 129. If you change them, use values between 32 and 254.

If NetBIOS or IPX is selected, the protocol must be configured as explained under "Protocol Configuration Requirements" on page 50.

f Select the <<**Add** push button to add the server to the **Servers** list.

6 If you want to change the default direct binding port numbers, use the **Direct binding ports** page. If you set the direct binding ports for a particular protocol to 0, the server does not listen on that protocol.

7 Select the **User** tab. The **User** section is displayed, as shown in Figure 30.

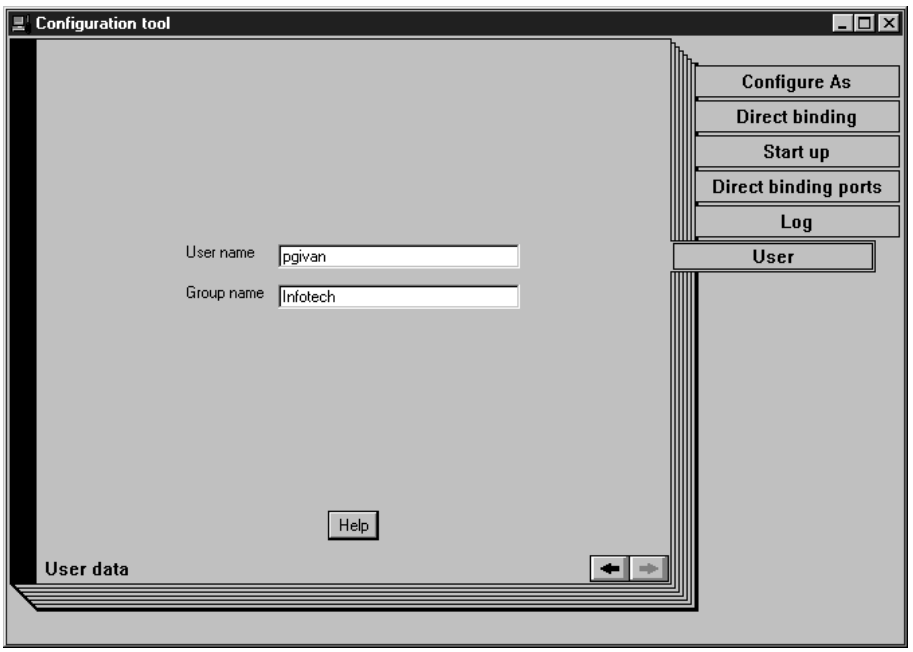


Figure 30. Configuration Tool Notebook - User Section

The user name is the logon user ID in effect when you do the configuration. Enter a group name if you want to change the default.

Note that the License Use Runtime group name is not the same as the Windows NT group name.

Configuring a Network License Client

License Use Runtime treats the user name as the name of the user who requests and uses licenses on this machine. The user name is displayed as part of various reports about license usage. It is also used to reserve reservable licenses for a specific user. The group name is used to reserve reservable licenses for members of a specific group.

8 Select **Close** from the system menu in the upper left corner of the notebook. A message is displayed to confirm that you are ready to save your choices.

9 To run the configured processes, see “Starting and Listing Your Processes” on page 90.

Now the workstation can be used as a network license server, to install and grant product licenses and monitor their usage.

Command-Line Equivalent

To configure the network license server:

```
i4cfg -a s -S a,s -e cwegvmps -l c:\ifor\ls\baratti -g Infotech
-b "'network ip:pgivan ip:lferrett ip:aclapes'
'nodelocked ip:lferrett ip:aclapes ip:woolf' 'registry ip:lferrett'"
```

Scenario 4: Configuring a Network License Client

This scenario shows how the administrator or the end user configures License Use Runtime as a client of the network license servers configured in “Scenario 3: Configuring a Network License Server” on page 79 and of the central registry license server configured in “Scenario 5: Configuring the Central Registry License Server” on page 88. Configuring a machine as a network license client makes it possible to use licenses of the types shown in Table 10 on page 69.

This scenario shows the steps the end user follows to configure the network license client after installation of License Use Runtime.

1 Double-click on **Configuration Tool** in the License Use Runtime folder. The Configuration Tool notebook is displayed, as shown in Figure 31 on page 85. Check **Advanced Configuration** to add the **User** tab to the notebook.

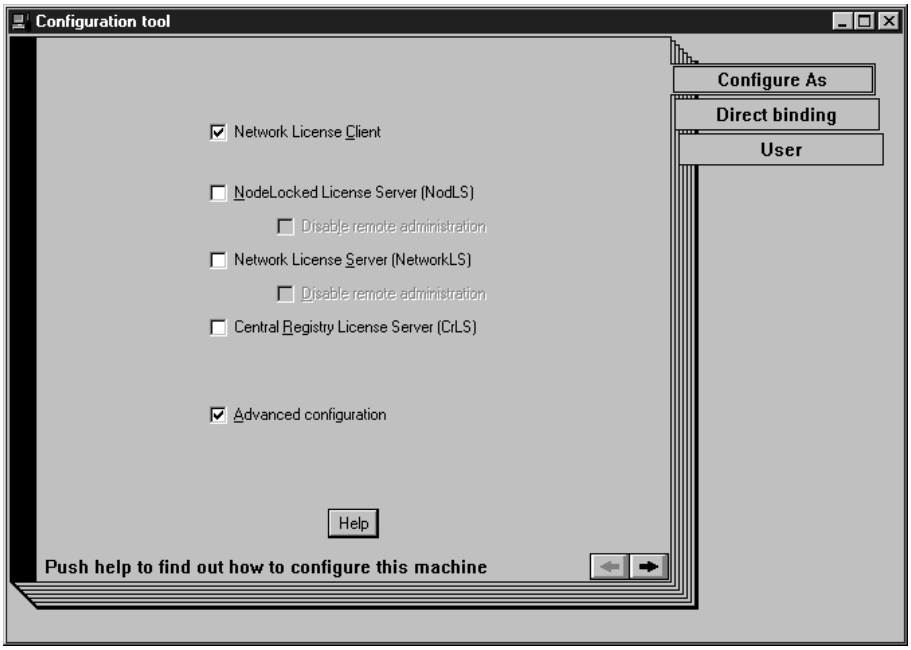


Figure 31. Configuration Tool Notebook - Network License Client

2 Select the **Direct binding** tab. The Direct binding section is displayed, as shown in Figure 32 on page 86.

Configuring a Network License Client

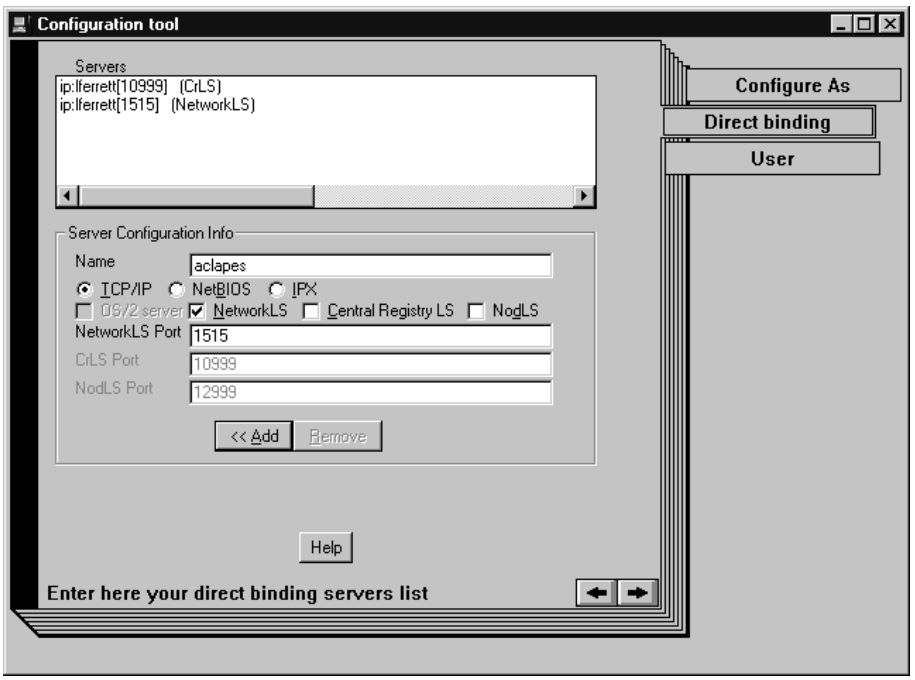


Figure 32. Configuration Tool Notebook - Direct Binding Section

On this page you specify all the license servers with which this client will communicate. Before you begin specifying the servers, be sure you have read the performance notes under “Planning Direct Binding” on page 37.

In this example, for each license server, the administrator does the following:

a Select the **TCP/IP** radio button. On the Windows NT Alpha platform, this is the only choice.

b Enter the TCP/IP host name of the license server in the **Name** field. (Note that the server name is case-sensitive.)

c Check **NodLS**, **NetworkLS**, **Central Registry LS**, or any combination, depending on the roles the server plays in the network.

d Leave the default values in the **NetworkLS Port**, **CrLS Port**, and **NodLS** fields.

For platforms other than Windows NT Alpha, Windows 95, or Windows 98, if NetBIOS is selected the default values are, respectively, 115, 109, and 129. If you change them, use values included between 32 and 254.

If NetBIOS or IPX is selected, the protocol must be configured as explained under “Protocol Configuration Requirements” on page 50.

e Select the <<**Add** push button to add the server to the **Servers** list.

3 Select the **User** tab. The User section is displayed, as shown in Figure 33.

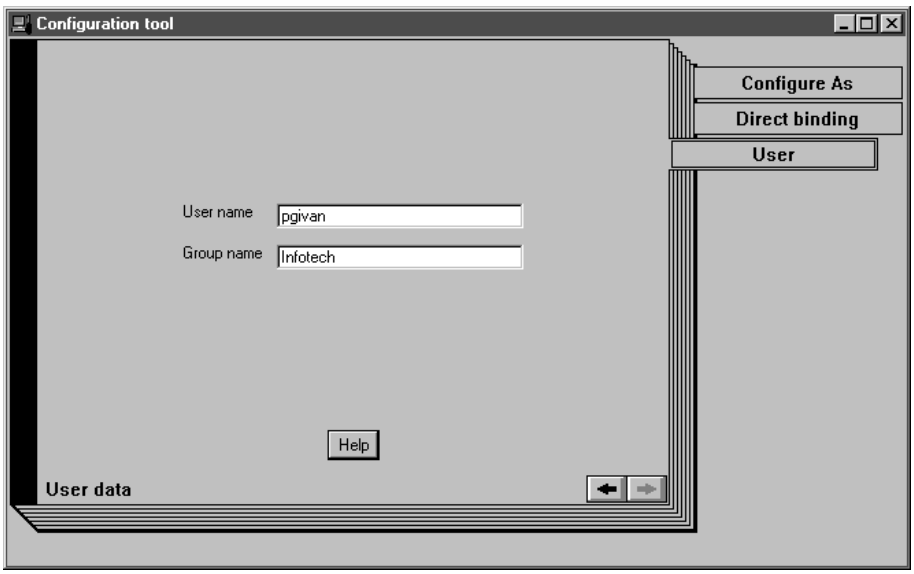


Figure 33. Configuration Tool Notebook - User Section

The user name is the logon user ID in effect when you do the configuration. Enter a group name if you want to change the default.

Note that the License Use Runtime group name is not the same as the Windows NT group name.

License Use Runtime treats the user name as the name of the user who requests and uses licenses on this machine. The user name is displayed as part of various reports about license usage. It is also used to reserve reservable licenses for a specific user. The group name is used to reserve reservable licenses for members of a specific group.

4 Select **Close** from the system menu at the upper left corner of the notebook. A message is displayed to confirm that you are ready to save your choices.

The end user can now use products that have network licenses.

Command-Line Equivalent

To configure the network license client:

```
i4cfg -a c -g Infotech
-b "'network ip:lferrett ip:aclapes' 'registry ip:lferrett'"
```

Configuring the Central Registry License Server

Scenario 5: Configuring the Central Registry License Server

This scenario shows how the administrator configures the central registry license server (in this example, *pgivan*), making it part of a direct binding servers list.

Be sure you configure only one central registry license server. Plan carefully where to configure it; once you start it, you cannot move it. (See “Planning the Central Registry” on page 40.)

Configuring a machine as the central registry license server makes it possible to use licenses of the types shown in Table 10 on page 69. It also makes it possible to use the Basic License Tool to administer licenses on remote license servers in the network.

To configure the central registry license server after installation of License Use Runtime:

1. Double-click on **Configuration Tool** in the License Use Runtime folder. The Configuration Tool notebook is displayed.
2. On the **Configure As** page, select **Central Registry License Server**. Note that **Network License Client** is then automatically checked.

   Check **Advanced Configuration,** which adds the **User**, **Log**, and **Direct binding ports** pages to the notebook

   The **Configure As** page is shown in Figure 34.

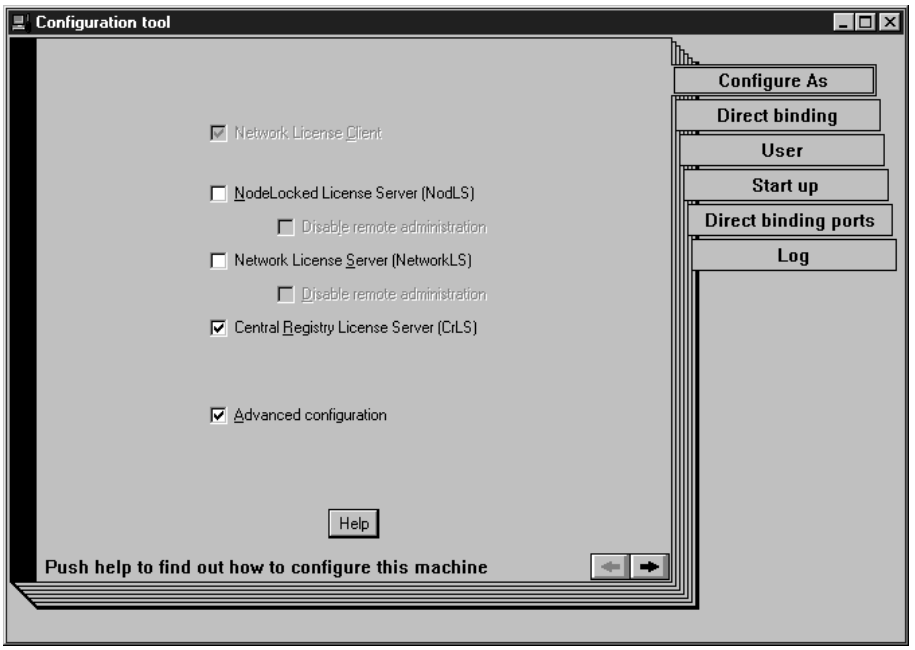


Figure 34. Configuration Tool Notebook - Central Registry License Server

3 On the **Start up** page, select **Start services at system startup** to start the central registry license server when you power on the machine. (This option is available on Windows NT only.)

4 Select the **Direct binding** tab. The Direct binding section is displayed, as shown in Figure 35.

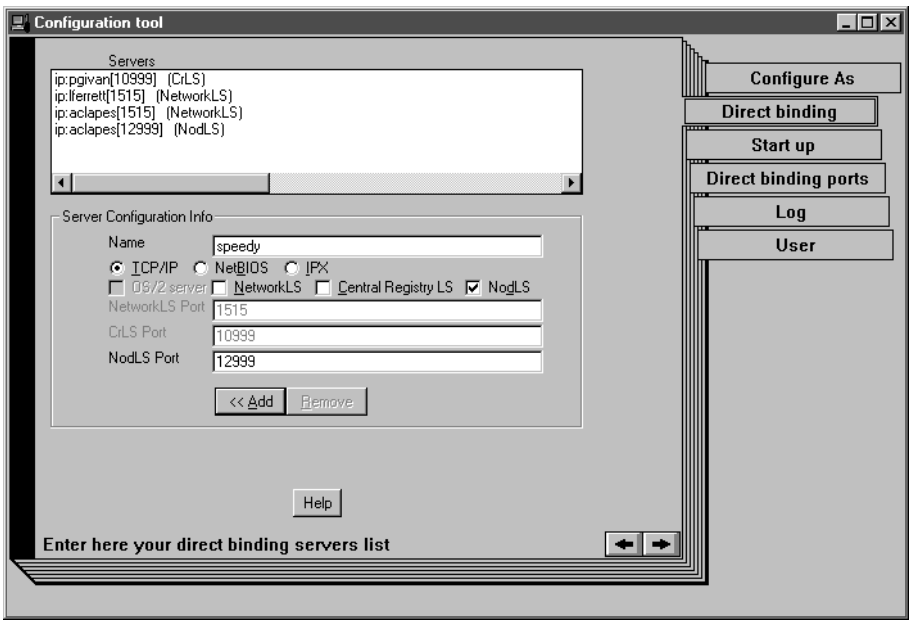


Figure 35. Configuration Tool Notebook - Direct Binding Section

On this page you specify all the license servers in the direct binding servers list. You must include the central registry license server that you are configuring. If the central registry license server and a network license server or nodelocked license server run on the same machine, include *all* servers in the list.

As you configure the servers in the direct binding servers list, be sure you define exactly the same set of servers on each.

In addition to specifying the direct binding servers list, use this page to specify remote nodelocked license servers whose licenses you want to administer from this machine. Do not include the nodelocked license server on this machine. (You can administer local nodelocked licenses automatically, without specifying direct binding.)

In this example, the administrator performs the following steps for each server with which this server will communicate:

a Select the **TCP/IP** radio button. On the Windows NT Alpha platform, this is the only choice.

b Check **NodLS, NetworkLS,** or **Central Registry LS,** or any combination, depending on the roles the machine plays in the network.

c Leave the default values in the **NetworkLS Port, CrLS Port**, and **NodLS Port** fields.

For platforms other than Windows NT Alpha, Windows 95, or Windows 98, if NetBIOS is selected the default values are, respectively, 115, 109, and 129. If you change them, use values between 32 and 254.

d Select the <<**Add** push button to add the server to the **Servers** list.

5 Select the **User** tab. The **User** section is displayed.

The user name is the logon user ID in effect when you do the configuration. Enter a group name if you want to change the default.

Note that the License Use Runtime group name is not the same as the Windows NT group name.

License Use Runtime treats the user name as the name of the user who requests and uses licenses on this machine. The user name is displayed as part of various reports about license usage. It is also used to reserve reservable licenses for a specific user. The group name is used to reserve reservable licenses for members of a specific group.

6 Select **Close** from the system menu in the upper left corner of the notebook. A message is displayed to confirm that you are ready to save your choices.

7 To run the configured processes, see “Starting and Listing Your Processes.”

Now the workstation can be used as the central registry license server.

Command-Line Equivalent

To configure the central registry license server:

```
i4cfg -a r -S a -g Infotech -b "'network ip:lferrett ip:aclapes'
'nodelocked ip:speedy ip:aclapes' 'registry ip:pgivan'"
```

Starting and Listing Your Processes

In Windows NT, you can start and list your processes in two ways: treating them as Windows NT services (“License Use Runtime Processes Defined as Windows NT Services”) or using Service Manager Tool (“Managing License Use Runtime Processes with Service Manager Tool” on page 91).

License Use Runtime Processes Defined as Windows NT Services

During installation, your License Use Runtime processes are defined as Windows NT services. You can view them in the Settings --> Control Panel --> Services window. If

you are logged on with administrator authority, you can also start and stop the services in the same window.

During configuration, the processes you configure are defined as manual services, which means they are not started when you start the machine. You can change them to automatic services in the Services window, or by checking the Start services at system startup box on the Start up page of the configuration tool. Services you do not configure are defined as disabled services.

If you uninstall License Use Runtime or one or more of its components, go to the same window and disable any services you have designated as automatic. In this way you will avoid errors at system startup.

Note that errors during start or stop of the License Use Runtime processes are logged in the Windows NT log. To view them, go to the Programs --> Administrative Tools (Common) --> Event Viewer window, and select Log --> Application.

Managing License Use Runtime Processes with Service Manager Tool

When you finish your configuration, double-click on **Service Manager Tool** in the License Use Runtime folder to start the processes. The Service Manager Tool window is displayed, as shown in Figure 36.

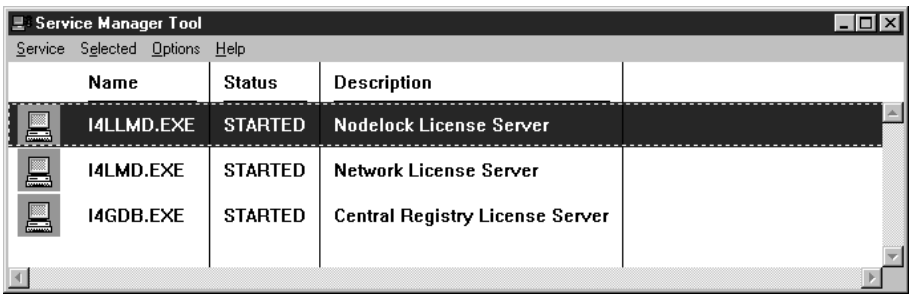


Figure 36. Service Manager Tool Window

In the Service Manager Tool window select **Service** from the menu bar and **Start** from the pull-down menu.

If you prefer to use the command-line interface, to start the processes you have configured on your machine, issue the command:

```
i4cfg -start
```

To verify that they are up and running, issue the command:

```
i4cfg -list
```

Administering License Use

Verifying Connections to Servers

To verify that license servers are running properly, use the i4tv (test verification) tool, or use the `i4blt -ln` command to get a list of active servers (network license servers, nodelocked license servers, and the central registry license server). For more information about these commands, see Chapter 5, “License Use Runtime Commands” on page 127.

Administering License Use

The rest of this chapter consists of scenarios that illustrate how the administrator performs the daily activities of managing license-enabled products.

The scenarios assume that the administrator has configured the nodelocked license server, a network license server, and the central registry license server on the server named *PGIVAN*, and that they are all up and running. The Basic License Tool is run from the server named *PGIVAN*.

The scenarios use three sample license-enabled products from three fictitious IBM vendors:

SMARTJava Version 2.3, a product of the vendor IBM Software Group. SMARTJava has concurrent licenses, which the vendor delivers via a compound password. It is a customer-managed use product, and the vendor enabled it to allow the customer to exercise the hard stop/soft stop policy. Its enrollment certificate is named *smrtjava.lic*.

DataMaster Version 2.1a, a product of the vendor IBM Corporation. DataMaster is a vendor-managed use product with reservable licenses. Its enrollment certificate is named *datamst.lic*.

e-MailVision Version 1.2, a product of the vendor IBM Software Solutions. e-MailVision has per-server/per-seat licenses. The enrollment certificate for the per-server license is *emailvps.lic*, and for per-seat it is *emailvpt.lic*.

Of course, when you perform the activities illustrated in the scenarios you must substitute your own values for variables such as server name, product name, vendor name, enrollment certificate name, product version, and user name.

Using the Basic License Tool GUI

The scenarios, except for “Scenario 10: Restricting User Access” on page 113, use the Basic License Tool graphical user interface, which features a graphical summary of information about all the products with licenses on the servers you select, a notebook of details about each product, and a graphical summary of information about clusters of network license servers.

Starting the Basic License Tool GUI

You must start the Basic License Tool GUI before you can run any of these scenarios. To start the GUI, double-click on the **Basic License Tool** icon in the License Use Runtime folder.

Alternatively, start the Basic License Tool GUI by entering the i4blt command with no parameters. The Basic License Tool window is displayed (Figure 37 on page 93).

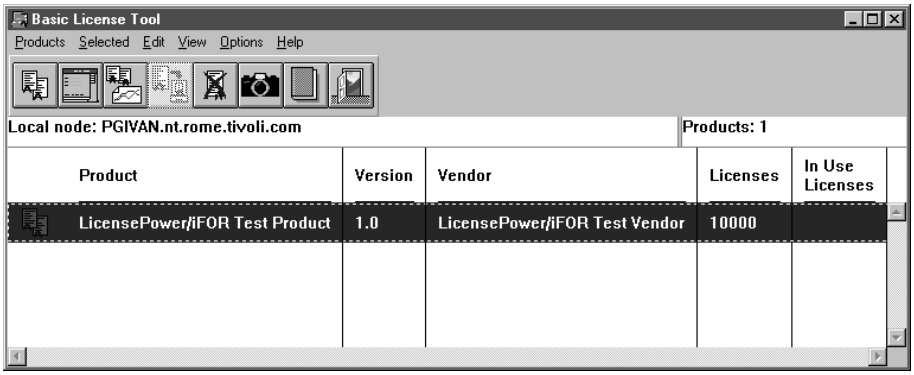


Figure 37. Basic License Tool Window

The products in the list displayed belong to all the active network license servers, nodelocked license servers, and the central registry.

In the Basic License Tool window, the heading *Licenses* indicates licenses that are available to end users (including those that are in use) *unless* the current date is before the start date of the licenses or the licenses have expired.

Refreshing License Information

In general, before using the Basic License Tool to view information about concurrent, reservable, per-server, and concurrent nodelocked licenses in use, highlight the product you are interested in and select **Clean up stale licenses** from the **Selected** pull-down. After the cleanup finishes, press F5 to refresh the window.

Selecting Servers

Use the **Settings...** option of the **Options** pull-down menu to specify types of licenses (nodelocked, network, or both) to be gathered as of the next refresh. You also have the option (exercised from the **Include...** option of the **View** pull-down menu) to select a subset of the active nodelocked and network license servers. Licenses of the types specified in **Settings...,** on the servers specified in **Include...,** are available to the administrator.

Selection of a server is effective as long as the server is active. If a server goes down and is restarted, it is no longer selected.

If a selected server shuts down, and you subsequently request a report or a display of information gathered from all selected servers, the request fails and error messages notify you that it was not possible to communicate with the server. In this case, use the **Refresh Now** option in the **View** pull-down, or deselect or restart the failing server, and try again.

Managing a Licensed Product

Using the Basic License Tool Command-Line Interface

You can achieve the same results using the Basic License Tool GUI or using the corresponding command-line interface. At the end of each scenario, a section called "Command-Line Equivalent" shows how to accomplish the same results using the command-line interface.

Performing Basic Administration

The scenarios in this section demonstrate how to:

Enroll a licensed product ("Scenario 6: Managing a Licensed Product").

Distribute licenses from a compound password ("Scenario 6: Managing a Licensed Product").

Get a report on the use of licensed products ("Scenario 6: Managing a Licensed Product").

Check the current users of licensed products ("Scenario 6: Managing a Licensed Product").

Reserve reservable licenses for specific users and monitor the use of reservable licenses ("Scenario 7: Managing Reservable Licenses" on page 102).

Scenario 6: Managing a Licensed Product

In this scenario, the administrator enrolls and manages the SMARTJava product. This scenario shows you how to:

Enroll the SMARTJava product
Enroll 20 licenses for SMARTJava
Distribute five of the SMARTJava licenses to a network license server
Request a report on usage of SMARTJava licenses during a one-month period
Check the number of concurrent users of SMARTJava

Enrolling the Product

To enroll the SMARTJava product:

1. In the Basic License Tool window (Figure 37 on page 93), select **Products** from the menu bar.
2. Select **Enroll product...** from the pull-down menu.

   The Enroll Product window is displayed.
3. Select **Import...**. The Import window is displayed. From the **File name** list, select the enrollment certificate **smrtjava.lic**, as shown in Figure 38 on page 95.

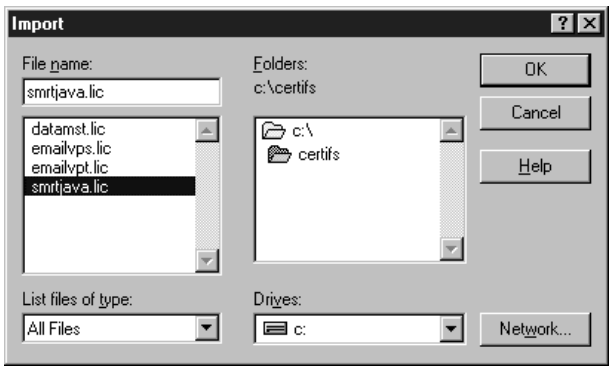


Figure 38. Import Window

Click on **OK**.

The Enroll Product window is redisplayed, filled in with the information from the enrollment certificate file, as shown in Figure 39.

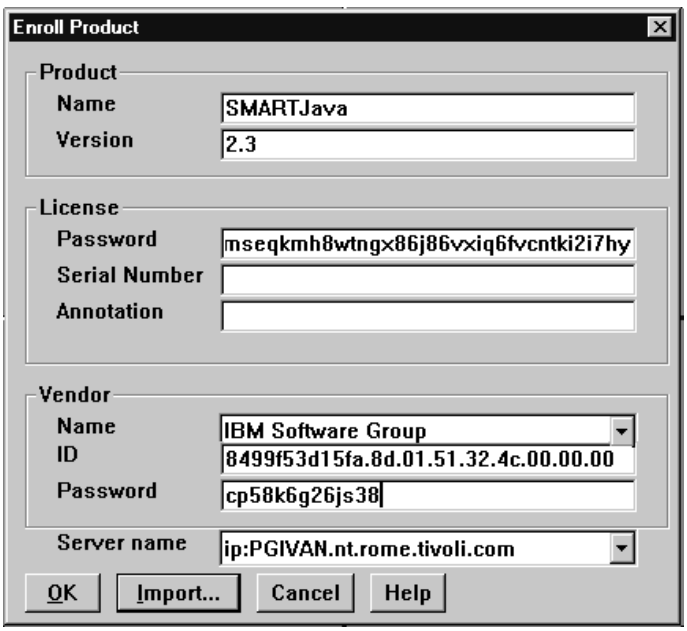


Figure 39. Enroll Product Window

4 In the **Server name** field, select the server on which the licenses are to be installed. In this example, because the product is customer-managed and has network licenses, the licenses are enrolled on the central registry license server (*PGIVAN*).

If the product is vendor-managed, and has a specific target ID set in the enrollment certificate, the licenses must be enrolled on the network license server or nodelocked license server of that target machine.

If the product is vendor-managed and the target ID in the enrollment certificate file is set to ANY, select a network license server or a nodelocked license server, depending on license type.

If the product is customer-managed and has nodelocked licenses, select a nodelocked license server.

If the password is bound to a cluster rather than an individual server, select any network license server that is an activated member of the cluster.

Click on **OK**.

Because the product is customer-managed, the Enroll Licenses window is displayed.

5 Fill in the Enroll Licenses window with your user information and the number of licenses you want to enroll (in this example, 20) as shown in Figure 40.

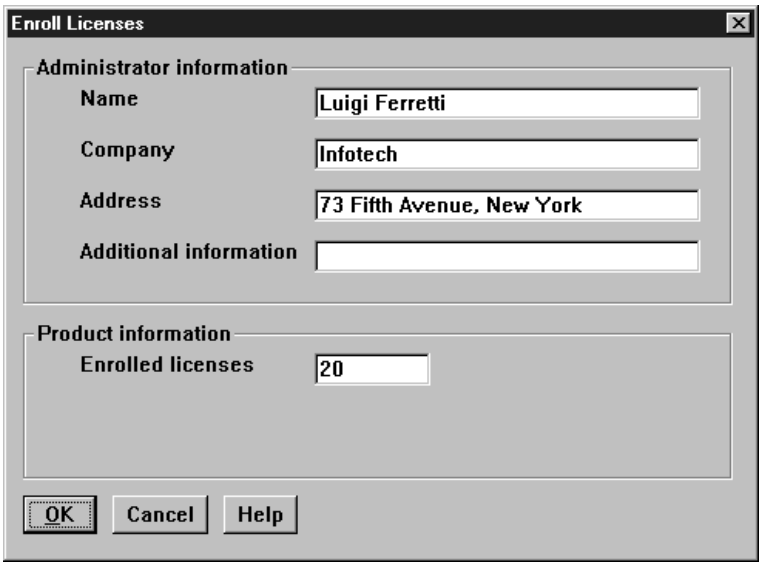


Figure 40. Enroll Licenses Window

Select **OK**. A message is displayed indicating that the product has been enrolled.

The Basic License Tool window is displayed again, as shown in Figure 41 on page 97. Note that there is a new line corresponding to SMARTJava, and that the icon for the product (a wrapped box) indicates that the product has a compound password.

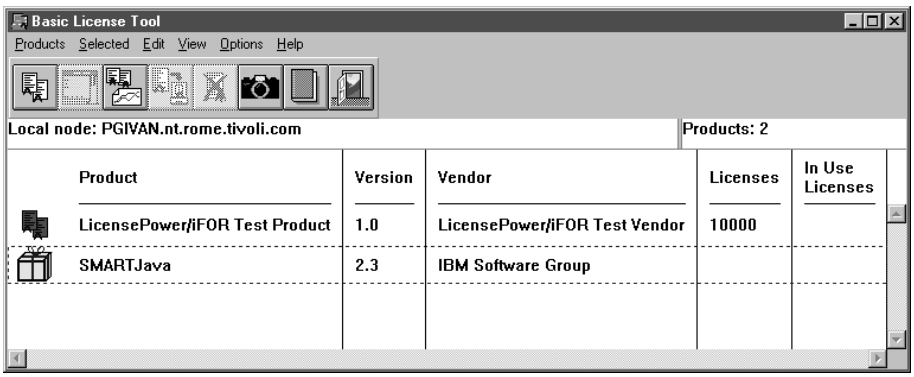


Figure 41. Basic License Tool Window with SMARTJava Enrolled

Distributing the Licenses

Before the network license clients can be granted licenses to use the product, the administrator must distribute the licenses to a network license server.

Distribution of licenses is required in the case of network licenses delivered with a compound password. Simple passwords (for example, per-server, per-seat, and concurrent nodelocked licenses) must not be distributed.

To distribute five licenses to the network license server *PGIVAN*, the administrator performs the following steps:

1. Select the line corresponding to the SMARTJava product in the Basic License Tool window (Figure 41).
2. Select **Selected** from the menu bar and **Distribute licenses...** from the pull-down menu. The Distribute Licenses window is displayed, as shown in Figure 42.

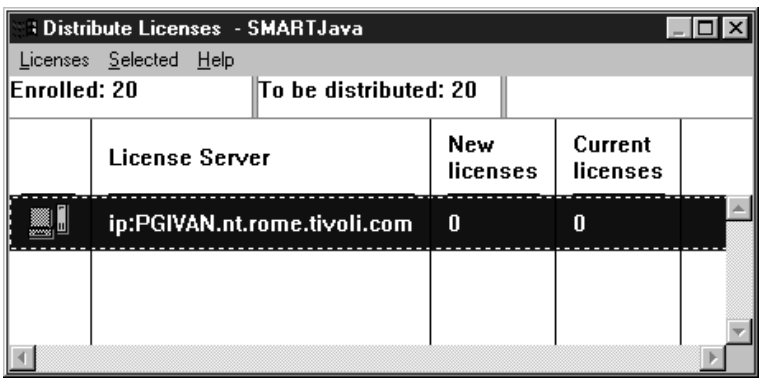
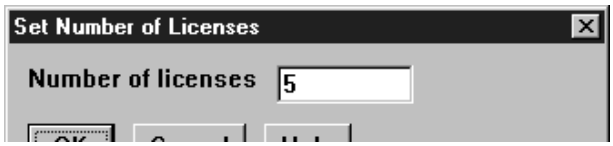


Figure 42. Distribute Licenses Window

3 In the Distribute Licenses window, select the network license server. Click on the selected server with the right mouse button. A pop-up menu is displayed. Select **Set number of licenses...**. The Set number of licenses window is displayed.

4 Enter **5** in the **Number of licenses** field, as shown in Figure 43, and click on **OK**.