IBM License Use Management Runtime

IBM

Using License Use Management Runtime Fix Pack 12

Version 4.6.8.12

IBM License Use Management Runtime

Using License Use Management Runtime Fix Pack 12

Version 4.6.8.12

Sixth Edition (May 2008)

This edition replaces SH19-4553-04. The major changes are described in Summary of Changes for Versions 4.6.8 to 4.6.8 fix pack 12.

Sixth Edition (May 2008)

This major revision obsoletes and replaces SH19-4553-04. The major changes are described in "Summary of Changes for Versions 4.6.8 to 4.6.8 fix pack 12" on page xvii.

This edition applies to Version 4.6.8.12 of IBM License Use Management Runtime, Program Numbers 5765-G61, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure that 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.

Fix Pack 12

Note: Changes made to this reference manual, other than minor or obvious ones, are indicated in vertical bars to the left of the changes.

IBM welcomes your comments.

Address your comments to:

License Use Management Information Development
SWG Krakow Laboratory
IBM Polska Sp. z o.o.
ul. Armii Krajowej 18
30-150 Krakow
Poland

Fax Number: +48 (12) 628 9993

Internet ID: LUMSERV@PL.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.

Contents

Figures

Tables

About This Book

This book describes IBM® License Use Management Runtime 4.6.8, fixpack 12 for AIX®, HP-UX, Solaris, and Windows®. It contains information about how to set up:

- The servers where product licenses are to be installed
- The clients that use the products
- An environment that can support network licensing

It also explains how to manage the licenses for the products you install. Before reading this guide, read the documentation supplied with the licensed product.

Who Should Read This Book

This book is intended for the following types of users.

- The system administrator who is responsible for setting up and administering the license management environment. According to the platform where the product is installed, this person, depending on the platform being used, must have general knowledge of the following products and environments:
  - AIX, Windows, HP-UX or Solaris
  - The UDP protocol
  - The network environment
- The end users who run the software products and the License Use Management Runtime component. End users require only general knowledge of the AIX, Windows, HP-UX or Solaris operating environment.

How This Book is Organized

This book is organized into the following sections:

Chapter 1, "Introduction" 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 for the use of licensed products.

Chapter 2, "Planning Network Licensing" provides the basic NCS concepts you need to set up your network and manage licenses.

Chapter 3, "Installing License Use Management Runtime" explains how to install, uninstall, and upgrade License Use Runtime components.

Chapter 4, "Configuring License Use Management Runtime" provides scenarios for configuring License Use Runtime, administering product licenses, and implementing customer-controlled policies, using the License Use Runtime tools.

Chapter 5, "Administering License Use Management Runtime" describes daily tasks such as administering product licenses, and implementing customer-controlled policies.

Chapter 6, "License Use Management Runtime Command Line Interface" documents the License Use Runtime command line interface.

Chapter 7, "Managing the licensing environment effectively" explains how to take better advantage of License Use Management Runtime.

Chapter 8, "Troubleshooting" explains how to improve performance and handle problems, should they arise when you use license-enabled products.

Appendix A, "License Use Management 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, "Operating Systems and License Use Management Support" identifies the operating systems supported by License Use Management and indicates, where appropriate, the conditions of their support.

Appendix D, "Accessibility" provides a summary of the implementations that make the use of the interface accessible to people with some physical disabilities.

The Glossary explains terms used in this manual.

Prerequisite and Related Information

This section lists other sources of information for License Use Management Runtime.

README File

For changes to License Use Management Runtime that were made after the book was finished, see the README.ARK file in the download package from the License Use Management Web site.

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

Depending on the operating system you are working on, you can find the README.ARK file in the directories listed below:

AIX

/usr/opt/ifor/ls/os/aix/doc

On AIX, the file is written in small case *readme.ark*.

HP-UX

/opt/lum/ls/os/hpux/doc

Solaris

/opt/lum/ls/os/solaris/doc

Windows

C:\ifor

where C: is the drive where License Use Management Runtime was installed.

MAN Pages for Command Syntax

To display the man pages, you might need to add an entry to the MANPATH environmental variable in the .profile. Depending on the operating system, new entry must specify the directory.

AIX

`/usr/opt/ifor/ls/os/aix/doc/en_US`

Man pages are installed with **ifor_ls.html.en_US** packages and if the **ifor_ls.base** package is installed there are no man pages in License Use Management Runtime.

HP-UX

`/opt/lum/ls/os/hpux/doc`

Solaris

`/opt/lum/ls/os/solaris/doc`

To get help with the syntax of a particular command, go to the *language* directory and type:

`man command_name`

This displays the command syntax. For example, to get help with the command i4blt, type:

`man i4blt`

HTML Files

To view an HTML file, open it in your Web browser.

If you are working on a non-IBM UNIX® (HP-UX, Solaris) platform, the files are in the following directories:

HP-UX

`/opt/lum/ls/os/hpux/doc`

Solaris

`/opt/lum/ls/os/solaris/doc`

If you are working on an AIX platform, the files are in one of the following directories:

- If only the ifor_ls.base package is installed:

  `/usr/opt/ifor/ls/os/aix/doc`

- If both the ifor_ls.base and the ifor_ls.html.en_US packages are installed:

  `/usr/opt/ifor/ls/os/aix/doc/en_US`

If you are working on Windows platform, the files are in the directory:

`c:\ifor\win\bin\en_US`

There are HTML versions of:

- License Use Management Runtime Message References for non-IBM UNIX (HP-UX, Solaris), AIX and Windows:

  `lummsg.htm`

- This book (*Using License Use Management Runtime*):

  `lumusg.htm`

  A PDF version of this book is available for download from the IBM License Use Management Web site.

- *Using Application Developer's Toolkit* (if Application Developer's Toolkit is installed in your environment):

  `lumtkt.htm`

  The location of this file depends on Application Developer's Toolkit installation path.

  This book describes how to license-enable software products for use with License Use Management.

License Use Management Web Site

For information and news about License Use Management, and to download License Use Management Runtime publications and code, visit the License Use Management Web site at:

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

Notational Conventions

This book uses the following notation in text:

Bold   Bold print indicates something you click, select, or type, such as a menu option, field, or push button.

Italics   Italics are used for variables, for new terms in the text, for directories and for emphasis.

`Monospacing`
   Monospacing is used for system messages and examples.

When you see an operating system icon, the text applies only to the operating systems indicated by that icon until the next icon appears. Icons used in this book are described in the following table.

ALL	This icon identifies text that is common to all operating systems.
UNIX	This icon identifies text that refers to all NON-IBM UNIX operating systems (HP-UX, Solaris).
AIX	This icon refers to text for AIX systems only
Windows	This icon refers to text for Windows systems only
	This icon refers to text for AIX and non-IBM UNIX systems only
	This icon refers to text for AIX and Windows systems only
	This icon refers to text for NON-IBM UNIX and Windows systems only

Default Windows Installation

In all Windows examples and commands, this book uses the default installation drive (c:\) for directory notation. You can use any drive for installation, but remember to substitute the driver letter shown in this book from c:\ to the one you used for installation.

Summary of Changes for Versions 4.6.8 to 4.6.8 fix pack 12

Fix Pack 12

This edition of the book incorporates changes made in Versions 4.6.8 through 4.6.8 Fix Pack 12 of License Use Management Runtime, which were documented in the README.ARK files for those versions.

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

Additional operating systems supported

The License Use Management Runtime supports the following additional operating systems:

- Windows Vista for 32–bit and 64–bit
- AIX 6.1.
- Solaris 9, 10 (SPARC)
- HP-UX 11.i v1

Operating systems no longer supported

The following operating systems are no longer supported:

- HP-UX 11.0, 10.20
- AIX 4.3x
- IRIX 6.3, 6.4, 6.5
- Solaris 2.6, 7
- Windows 95, 98, ME, NT 4.0

Database Integrity Monitoring

Added a capability to repair a corrupted or damaged database before a server procedure starts.

Support for Windows groups with names longer than 31 characters

License Use Management Runtime now supports Windows groups with names longer than 31 characters. Names longer than 31 characters are truncated.

xvii

Chapter 1. Introduction

ALL

License Use Management 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:

License Use Management Application Developer's Toolkit (License Use Management ADK)

Enables vendors to implement license management in an application. To implement license management, vendors embed API calls and API code in their products. The product is then described as *license-enabled*. Vendors can license-enable C-language programs and Java™ applications.

The License Use Management ADK offers the vendor great flexibility:

- Level of control that is used by the license-enabled application
- Type of customer licensing environment for which the application is intended
- Implementation of policies

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

License Use Management Runtime

Enables users of license-enabled software to manage the licensing environment. The License Use Management Runtime software is free of charge and is available for download from the License Use Management Web site:

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

License Use Management Vendor Perspective

License Use Management benefits software vendors by enabling them to do the following:

- 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

License Use Management Customer Perspective

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

- 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 term *license* in this context does not refer to the license agreement that governs use of, and rights to, a product.

License-Enabling a Product

Vendors deliver licenses to customers in the form of a *license password*. The password contains encrypted terms of the usage of the software product, such as the following:

- The number of licenses or concurrent copies of the product the customer can use
- The expiration date of the licenses
- The license type (see "License Types" on page 6)

In the license-enabling process, the vendor can do the following:

- Select among the various license types
- Decide to distribute licenses one-by-one or in packages of multiple licenses from which individual licenses can be extracted (see "License Passwords" on page 4)
- Implement direct controls over the use of licenses, or make it possible for the customer to control use of licenses (see "Use-Control Levels" on page 5)
- Impose, or allow the user to impose, various types of control over administration of licenses (see "License Policies" on page 10)

When the product is installed, License Use Management Runtime checks that a license is available each time the product is run.

Nodelocked and Network Licenses

Vendors can supply licenses in two forms: *nodelocked* licenses or *network* licenses.

A *nodelocked* license is a license for a specified workstation (node). It is stored on the specified workstation, and the license-enabled product will run on only that workstation.

A *network* license is a license, stored on a server, that can be used by any client workstation. Many License Use Management 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 Management Runtime at the license server determines whether a license is available.

License-enabled Java products can have network licenses or nodelocked licenses.

Managing Licenses

License Use Management Runtime includes the *basic license tool*, which manages both nodelocked and network licenses. The basic license tool enables you to do the following:

- 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 Chapter 5, "Administering License Use Management Runtime," on page 101.

License Use Management Runtime Platforms

Table 1 shows which platforms License Use Management Runtime fix pack 12 supports, and where to get the License Use Management Runtime code.

Table 1. License Use Management Runtime Platforms

AIX 6.1 or higher	License Use Management Runtime Version 4.6.8 base code is part of the base operating system, and is installed on every workstation when the operating system is installed. You can install Version 4.6.8.12 from the product package downloaded from the IBM License Use Management Web site: http://www.ibm.com/software/lum
AIX 5.x	The following levels of License Use Management Runtime are part of their associated level of the AIX operating system. They are installed on each workstation when AIX is installed. ``` AIX License Use Management Runtime 5.3 or higher...4.6.8 5.2.0.10 4.6.8 5.1.0.25 4.6.8 5.1.0.16 4.6.8 5.1.0.10 4.6.8 5.2.0.30 4.6.7 5.2.0.10 4.6.6 5.1.0.25 4.6.5 5.1.0.16 4.6.4 5.1.0.10 4.6.2 ``` Optional packages and filesets can be installed from the AIX installation media. To upgrade to License Use Management Runtime Version 4.6.8 or higher, download the code from the License Use Management Web site: http://www.ibm.com/software/lum

Table 1. License Use Management Runtime Platforms (continued)

Windows 2000 Windows Vista and XP Windows Server 2003	License Use Management Runtime Version 4.6.8.12 can either be redistributed with the license-enabled product or be downloaded from the IBM License Use Management Web site:
Sun Solaris 8, 9, and 10 (SPARC)	http://www.ibm.com/software/lum/component.htm
HP-UX 11.i v1	

License Passwords

Vendors provide license passwords (also known as *license keys*) with their products.

A license password is an encrypted character string. It contains information about the license, such as the following:
- The types of licenses (see "License Types" on page 6)
- The date when the licenses become active
- The date when the licenses expire

There are two types of password: simple and compound. A *simple password* represents one or more licenses. A *compound password* contains multiple simple passwords, each of which represents one or more licenses.

Note: The expiration date of simple licenses that are contained in a compound password must fall on or before the expiration date of their compound password.

Compound passwords can be used as follows:
- Vendors use compound passwords to distribute multiple licenses to their customers.
- The customer installs the compound password on the assigned license server. The customer extracts the licenses and distributes them, as required, to different license servers. (These servers are not specified in the compound password.)
- A sales representative extracts the licenses from the compound password and distributes them to customers.

The vendor includes the password, along with other information about the application, in the *enrollment certificate file* (ECF). You then make licenses available to users of the product by *enrolling* the product using the license password. See "Enrolling the Product" on page 103.

Use-Control Levels

Products can have one of the following predefined use control levels:
- Vendor-managed use control
- Customer-managed use control

Vendor-Managed Use Products

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

When you request licenses for a vendor-managed use product, the product vendor asks you to supply the unique identifier (target ID) of each workstation 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. You must also supply the number of licenses that you want to acquire. The vendor uses this information to create the password that you use to install and activate the licenses.

Vendors ship a vendor-managed use product typically with a simple password. They can also use compound passwords for this purpose.

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 workstations 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 workstation. 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 Management Runtime supports customer-managed use (CMU) 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 Management 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 Management 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 licenses 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 restricts the use of a product to the workstation specified in the license. Vendors typically use nodelocked licenses for standalone applications, rather than for client/server applications.

For nodelocked licenses there are two license-enabling models: *non-runtime-based* and *runtime-based*. If a vendor chooses non-runtime-based enabling, the license-enabled product itself, rather than License Use Management Runtime manages use of the nodelocked license. The password for such a product is stored in a *nodelock file*. When you start the application, it checks the nodelock file to ensure that 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 workstation. You then use the basic license tool to view and update information about the nodelocked licenses. See "License-Enabling Models" on page 15 for more information about license-enabling models.

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 workstation. A word processor is a typical example of a product that uses nodelocked licenses. Simple nodelocked licenses with no capacity, are valid only for vendor-managed use products.

Concurrent Nodelocked Licenses

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 that are 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 change 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 workstation. Every time the product is started, one license is used.

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.

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 change this number at any time.

Per-Server Licenses

Per-server licenses are similar to concurrent nodelocked licenses, except that at any time, you can change them into per-seat licenses (see "Scenario 3: Switching from Per-Server to Per-Seat Licenses" on page 112).

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 workstation 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 either of the following apply:
- Each application client connects to only a single application server
- 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 can 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 workstation, are stored on a network license server and shared among multiple network license clients.

Vendors can enable their products to use the following kinds of network licenses:
- Concurrent licenses
- Concurrent-offline 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.

Concurrent-Offline Licenses

The concurrent-offline license allows users to reserve a concurrent license for a certain number of days and to use it on a portable computer disconnected from the network. Only authorized users can exploit this feature. The users must be defined by the license administrator at the customer site. The portable computer can use supported Windows platforms only (see Table 1 on page 3 for a list of supported platforms).

A concurrent-offline license is installed on the network license server. It is similar to a normal concurrent license because it can also be used as a normal concurrent license.

For the duration of the reservation, the license remains unavailable to other users of the product. The license becomes available to other users when one of the following occurs:

- The reservation period expires
- The license is returned to the server from the portable computer

Concurrent-offline licenses can be enrolled and administered only on a server running License Use Management Version 4.6.7 (or later), by means of a Basic License Tool 4.6.7 (or later).

When the concurrent-offline license has been reserved for use, a copy is installed on a portable computer connected to the network. This copy of the concurrent-offline license is called an offline-nodelocked license. When the offline-nodelocked license has been installed on the portable computer, the portable computer can be disconnected from the network.

Reservable Licenses

A reservable license is a network license that the license administrator 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.

Notes:

1. A reservable license is reserved by the license administrator for the use of a user, group, or node. A concurrent-offline license is reserved by an authorized user who wants to use the license on a portable computer disconnected from the network.

2. Use of a reserved license on a portable computer requires the Nodelock License Server to be present on that portable computer. Use of a concurrent-offline license on a portable computer does not require the Nodelock License Server to be present on that portable.

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

The license administrator 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.

Note: Use of a reserved license on a portable computer requires the Nodelock License Server to be present on that portable computer. Use of a concurrent-offline license on a portable computer (the offline-nodelocked license) does not require the Nodelock License Server to be present on that portable.

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 used.

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.

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.

For 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 for a C or C++ application or for a Java application that uses the Java native interface 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 7.)

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

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 license policies: named user, capacity, try-and-buy, multiuse rules, value unit policy, product wait queues, license annotation, custom configuration, bundle, and product-specific program.

Named User Policy
With this policy, reservable licenses can be reserved and unreserved by the administrator in the same manner as for reservable licenses (see "Reservable Licenses" on page 9 for more information). The major difference is that the named user policy always requires you to reserve a license for a specific user. Also, with the named user policy, when a license is requested, only the central registry server is checked for an existing reservation. No communication occurs between the application and the nodelocked license server. Therefore, if you are using this policy, a nodelocked license server does not need to be installed on the client where this policy is in force. However, the workstation where the application is running must always be connected to the network.

If the central registry license server allows the license to be granted, the license is granted, but the transaction is not recorded. This allows the administrator to unreserve the license at any time, without waiting for the reservation period to expire.

Capacity
With this policy vendors can comply with the terms and conditions of products that are priced based on certain characteristics of the system on which the product is running, not only on the number of users who are using the product. This includes, for example, the number of processors on which the application can run or the maximum size of the physical memory.

In such cases the vendor can specify a *capacity type* that defines the unit of measure for the internal counter ("capacity units") associated to the license password. The capacity type is an integer value between 1 and 255; some of these values have a predefined value; others may have any vendor specific meaning.

The predefined capacity types and supported platforms are shown in Table 2:

Table 2. Predefined capacity types and supported platforms

	Configured processors	Online processors	Physical disks	Physical memory (in MB)
AIX	Y	Y	Y	Y
HP-UX	N	Y	Y	Y
Solaris	Y	Y	Y	Y
Windows	N	Y	Y	Y

When you use one of the predefined capacity types, the License Use Management library embedded in the license application can calculate the number of capacity units that are used by the application itself. In the other cases the vendor must calculate them before passing the results to the License Use Management library. In either case, the license is granted only if the capacity type of the password matches the capacity type specified in the request, and the capacity units are not exceeded. This applies to simple nodelock, concurrent nodelock, and concurrent network license types, under either vendor-managed or customer-managed use control.

Notes:

1. Not all the above capacity types are supported on all the platforms (see *Using License Use Management Application Developer's Toolkit*).

2. Hyper-threading technology enhances the performance of Intel® Xeon® processors by providing a mechanism for better utilization of the processor's execution units. For those applications that are licensed based on the number of operating system processors, License Use Management distinguishes between physical and logical processors and counts only the number of physical processors. This type of support is available on both Linux® and Windows platforms (running on Xeon processors).

   Processors that implement hyper-threading technology logically appear as two processors and provide incremental performance over a single processor without hyper-threading technology. However the performance will not be equivalent to two physical processors.

Value Units Policy

With this policy vendors price their products by assigning a value expressed in units to each resource in an environment. All the different metrics (like usage count, system resources or other product-specific indicators) are defined into the same license certificate, as well as the rules and relative weights for their translation into value units. The Value Units policy can only be applied to concurrent and concurrent-nodelocked licenses, or to server-based simple nodelocked licenses. See the table below.

Type of resource	Server Type A	Server Type B	No. of clients	No. of clients	No. of users
	2 processors	1 processor	1 - 10	11 - 50	
No. of value units assigned	50.15	30.20	20.30	15.00	10.00

Value units are multiplied by the number of resources in your environment and defined in the enrollment certificate. This policy enables you to change your environment to suit your needs, without exceeding the value units assigned.

Try-and-Buy

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 that is 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 calls of a product require only a single license. These rules are applicable only to concurrent network, concurrent nodelocked, concurrent-offline, and per-server licenses.

The vendor can enable a product so that after a license has been granted to a particular user, group, node, job ID, or accessor ID, a second call of the product does not require a second license.

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. For example, the vendor might implement a multiuse rule that applies when a request is associated with the same display as a previous request. Vendors can also change the meaning of "same group" in other ways, to implement whatever multiuse rules they design. Any vendor-specific rule overrides the "same group" 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.
- The request for a license is associated with the same accessor ID as a previous request.

  An accessor ID is a generic string that the vendor can use to further identify the caller of the license request. For example, if a user calls a compiler repeatedly, a multiuse rule might specify that the second and subsequent calls do not require additional licenses.

Product Wait Queues

Some products with concurrent licenses may use wait queues.

When a user calls such a product, and there are no licenses 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 Management Runtime network license server that provides concurrent licenses for

the product. User names are added to the wait queue in chronological sequence. When a license becomes available, it is granted to the first user in the queue.

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.

License Annotation File

If the vendor requires more annotation than is available using the 255 characters in the license annotation, the vendor can create an additional license annotation file, and refer to it using the **-j** parameter of the **i4lct** tool (see on page 0 for more information). The maximum size of the file is 32 KB. This file is distributed with the ECF and must be present on the system during the enrollment step.

Custom License Attributes

Vendor-specific license attributes can be defined and included as part of the license information when a license is created. When the license is granted, the attribute data is passed to the enabled application.

Three general purpose attributes can be used in License Use Management Runtime. These are specified with the **-A** parameter (see on page 0 for more information):

1. Custom attribute 1
2. Custom attribute 2
3. Custom attribute 3

These attributes can be exploited by a vendor to enforce any specific policy. License Use Management Runtime only records the attribute and makes it available to the vendor's application upon request.

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 group 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 a concurrent network license or a concurrent or 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.

Bundle

Sometimes it may be useful to treat a set of products as a unique entity. By defining *bundles*, licensing system administrators can manage the licenses referring to products as a unit.

The licensing system administrator can view all the bundles in a licensing system and see the licenses that are in bundles. You cannot delete individual licenses in a bundle, but you can delete the bundle.

Also, it is possible to verify the consistency of a bundle if all the licenses in the bundle have been enrolled.

Products in the bundle are linked by a common serial number, but each product is still represented by its own license password. The custom configuration policy, however, provides a way of bundling in which all the products are represented by a single license.

Product-Specific Program

Vendors who need to perform some product-specific actions at the moment of the license enrollment, can define in the license password the name and path of a program that License Use Management Runtime will run as part of the enrollment process. The product is responsible for the existence of such a program on the workstation where the enrollment takes place. If License Use Management does not find the program, the enrollment fails. This policy is available only for simple nodelock licenses under a non-runtime-based enabling (see "License-Enabling Models" on page 15).

Customer Controlled Policies

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

Hard Stop/Soft Stop

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.

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).

When you enroll a product enabled for hard stop/soft stop, the default is hard stop. To use soft-stop, you must enable a network license server with the correct option. You can use the basic license tool to change the policy to soft stop and back again. For details, see "Scenario 4: Using the Hard Stop/Soft Stop Policy" on page 113.

When the soft stop policy is set, License Use Management Runtime keeps track of the high-water mark. The *high-water mark* is the maximum number of licenses that are 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 zero. Use this number to help you decide the number of additional license keys 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.

While for CMU licenses the vendor can decide to let the user exploit the License Use Management Runtime hard stop/soft stop policy, VMU application vendors can decide to have a soft stop policy that is under the full control of the enabled application itself.

User Access Restriction

You can use the user file to control which users have access to licenses for specific products. The user file is an American National Standard Code for Information Interchange (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).

For details, see "Scenario 5: Restricting User Access" on page 115.

In addition to the user file method, the License Administrator can grant or deny usage of a product to specific users or specific groups through the basic license tool graphical user interface or command line. This applies only to concurrent and concurrent-offline products. It is impossible to change authorization remotely even if remote server administration is allowed.

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.

For information to help you decide between per-server and per-seat, see "Per-Server Licenses" on page 7 and "Per-Seat Licenses" on page 10. For information about how to perform the switch, see "Scenario 3: Switching from Per-Server to Per-Seat Licenses" on page 112.

License-Enabling Models

The product vendor can create license-enabled products that use nodelocked or network licenses.

Nodelocked license-enabled products (non-runtime-based)

If the vendor chose nodelocked non-runtime-based enabling, the product does not make use of License Use Management Runtime on the workstation 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 that 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.

Nodelocked license-enabled products (runtime-based)

If the vendor chose runtime-based enabling for a product with nodelocked licenses, the product makes use of License Use Management Runtime on the workstation 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.

Network license-enabled products (non-runtime-based)

A non-runtime-based network license-enabled product does not make use of License Use Management Runtime on the workstation where the product runs and requires manual configuration on that workstation. The licenses are stored on one or more network license servers. When the user at a client starts a licensed program, the license server determines whether or not a license is available.

Network license-enabled products (runtime based)

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

Table 3 summarizes the license-enabling models, license types, and license policies.

Table 3. License Types and Policies Available for License Use Management Runtime Products

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 Capacity Product-specific Program
Nodelocked license-enabled products (Runtime-based enabling)	Simple nodelocked	Try-and-buy License annotation Capacity Value Units Bundles Custom configuration
	Use-once nodelocked	User access restriction License annotation
	Concurrent nodelocked	Hard stop/soft stop Multiuse rules User access restriction License annotation Capacity Value Units Bundles

Table 3. License Types and Policies Available for License Use Management Runtime Products (continued)

License-Enabling Model	License Types Available	License Policies Available
	Per-server	Hard stop/soft stop Multiuse rules User access restriction License annotation Bundles
Network license-enabled products (runtime and non-runtime-based enabling)	Concurrent	Hard stop/soft stop Multiuse rules User access restriction Product wait queues License annotation License annotation file Capacity Value units Bundles Custom configuration
	Reservable	Hard stop/soft stop (when unreserved) Named User 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
	Concurrent-offline	Hard Stop/Soft Stop (if used as concurrent) User access restriction Multiuse rules (if used as concurrent) License annotation Custom configuration

Note: Custom license attributes are available for any license type.

Scalable Installation

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

For example, in AIX 5.3, all required License Use Management Runtime features are automatically installed on every AIX workstation as part of the AIX 5.3 installation. You can install optional features, such as the graphical user interface, either when you install AIX or later.

On Windows, you can select the appropriate features depending on the role the workstation is to play in your licensing environment. There are features for runtime, communications, and online documentation. Vendors can, optionally, incorporate the communications feature 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 download and install License Use Management Runtime.

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

The tool for configuring License Use Management Runtime license servers and clients (i4cfg) has a command-line interface and a graphical user interface on all platforms, and an alternative, interactive script interface on all UNIX platforms.

Administering Licenses

You use the *basic license tool* is to administer licenses.

The basic license tool has a command-line interface and a graphical user interface.

The tool enables you to do the following:
- Manage nodelocked and network licenses.
- Add licenses to or delete licenses from the network license server or nodelocked licensed server database; update information about existing customer-managed use licenses.
- Display a notebook of information about the licenses installed for each product.
- Extract licenses from a compound password and distribute them among the network license servers available on the network.
- Manage the reservation of reservable licenses for the exclusive use of certain users.
- Manage multiple network and nodelocked license servers. From any properly configured workstation, 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 workstations. The capability to manage licenses on nodelocked license servers is particularly useful for per-server and concurrent nodelocked licenses.
- Generate the following 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, and high-water mark changes)

– 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 that are 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 (when customer-managed) or license keys (when vendor-managed) 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.

• 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. You can change that threshold, and you can also set a separate threshold for each 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.

• Administer high-availability licensing (see "High-Availability Licensing"). That is, create clusters, activate and deactivate members of clusters, and show cluster status.

• Display a list of bundles, display a list of bundle features, and delete bundles.

For help on using the basic license tool, see "Performing Basic Administration" on page 102.

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 workstation.

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.

Changes of the state of each High Availability Licensing member are recorded on every cluster server in the server log file. A record is added specifying the current state of every server in the cluster. .

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 Internet Protocol (IP) and does not support the product wait queue policy. Before you decide to use high-availability licensing for a product, make sure that 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.

For planning information, see "Planning Clusters for High-Availability Licensing" on page 37. For an example, see "Creating a Cluster" on page 120.

Note: Only normal network concurrent licenses can exploit HAL.

Backing Up Databases and Files

On license servers, there is an automatic backup procedure for License Use Management Runtime databases and files. See "Automatic Backup Procedure" on page 199..

Working with Licensed Products

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

Only C, C++, and Java programs can be license-enabled.

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

This section assumes that, if required:
- License Use Management Runtime is installed properly.
- License Use Management Runtime is configured properly.
- 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 *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.

The central registry can contain:
- A record of all per-seat licenses
- The list of application clients to which per-seat licenses have been granted
- A record of soft stop license use
- The high-water mark
- Reserved licenses that have not yet been granted

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

Figure 1 shows what happens when an end user calls an application with nodelocked licenses for which the vendor chose non-runtime-based enabling. The licenses must be simple nodelocked licenses.
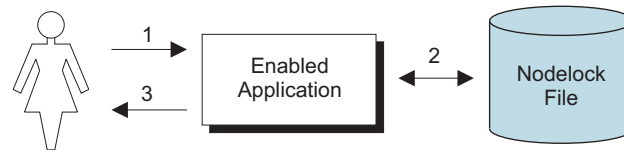
Figure 1. Using a Nodelocked License (Non-Runtime-Based Enabling)

1. The end user calls the application.
2. The application checks the nodelock file to ensure that 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 249.

Working with Nodelocked Licenses (Runtime-Based Enabling)

Figure 2 on page 22 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.

Working with Nodelocked Licenses
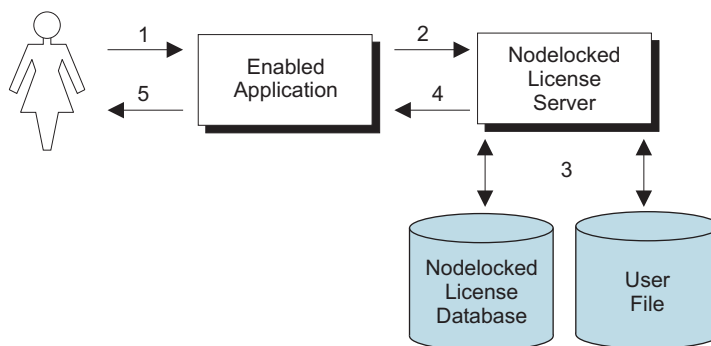
Figure 2. Using a Nodelocked License (Runtime-Based Enabling)

1. The end user calls 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 workstation 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 Use-Once Licenses

Figure 3 shows what happens when an end user calls an application with use-once licenses.
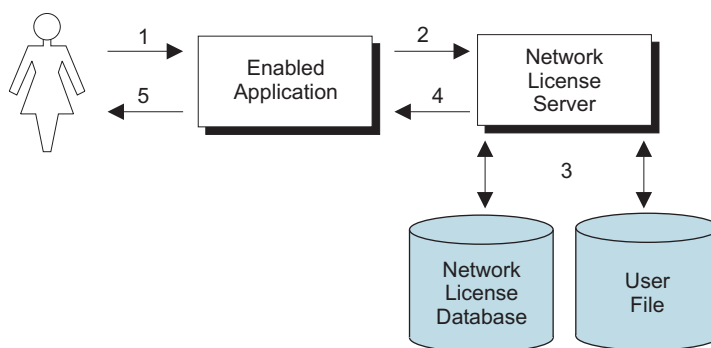
Figure 3. Using a Use-Once License

1. The user calls 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 is 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.

Working with Concurrent Licenses

Figure 4 shows what happens when an end user calls an application with concurrent licenses.
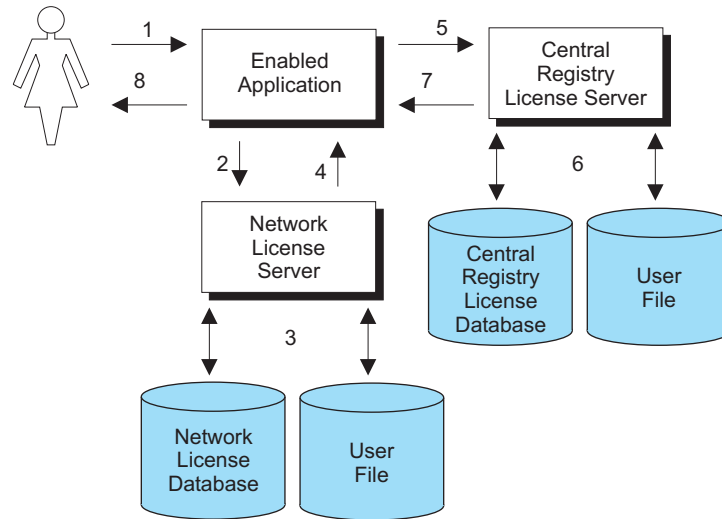
Figure 4. Using a Concurrent License

1. The user calls the application.
2. The application requests a license from the network license server.
3. The network license server checks the following:
   - Its license database for an authorization record that allows a request for that license for the user and group specified in the client request.
   - Its user file for authorization.
4. The network license server returns the status of the request to the application. If a license is found and granted, the application runs.
5. If a network license is 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 is 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.

Working with Concurrent-Offline Licenses

shows what happens when an end user calls an application with offline licenses.
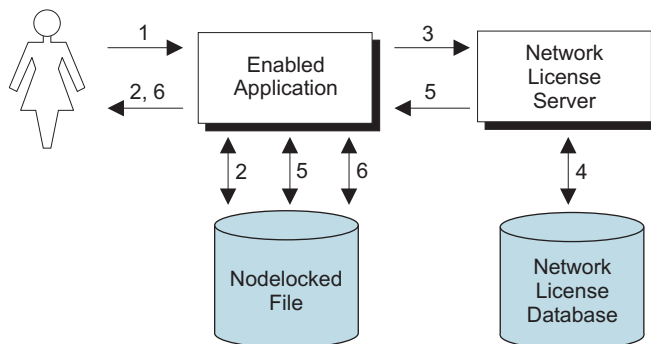
Figure 5. Using a Concurrent-Offline License

1. The user calls the application.
2. The application looks for an offline-nodelocked license in its nodelocked file.

   If a license is found and is valid, the application runs.
3. If the application does not find a license in the nodelocked file and the portable computer can reach a License Use Management network license server, the application requests a concurrent-offline license from it.
4. The network license server checks if the license has an authorization for the user, group, or for the target ID of the machine, and validates the password provided by the user.
5. If the authorization is for a concurrent-offline license, the server creates an offline-nodelocked license and copies it to the nodelocked file of the portable computer. The license is marked as in use on the server.
6. The application checks that the nodelocked file contains a valid license and the application starts. The application can now run without a connection to the License Use Management network license server (that is, the application can run on a portable computer).

Note: At the end of the reservation period, the copy of the license stored in the nodelocked file of the portable computer expires (the license can also be returned by the user before the reservation period expires). The license can be used by another user both as a normal concurrent license and as a new offline-nodelocked license.

Working with Reservable Licenses

Figure 6 on page 25 shows what happens when an end user calls an application with reservable licenses.
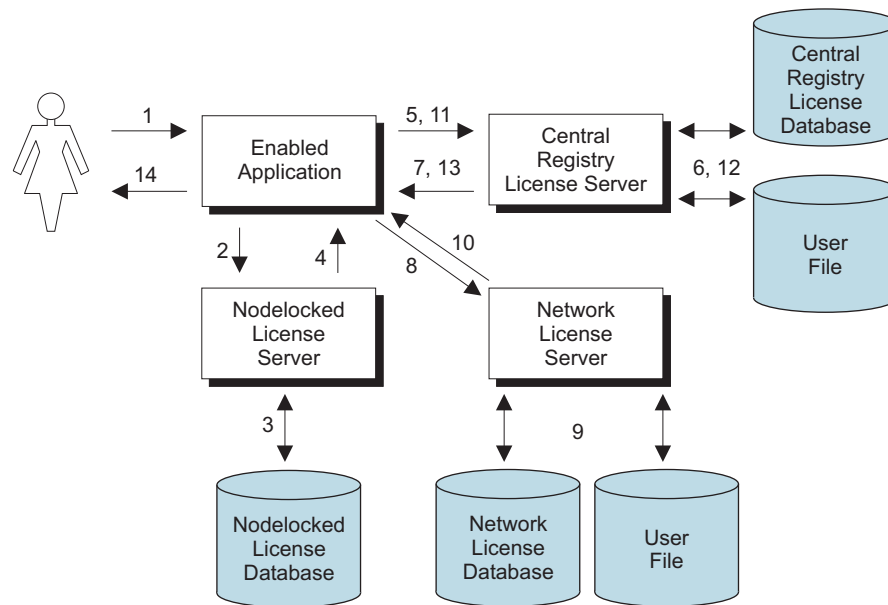
Figure 6. Using a Reservable License

1. The user calls 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 workstation, in response to a previous request.
4. The nodelocked license server returns the status of the request to the application. If a license is 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 is found and granted, it is stored in the nodelocked license server's database, and the application runs.
8. If a reserved license is 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 is found and granted, the application runs.
11. If a reservable license is 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 is 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.

Working with Reservable Licenses in Named User Policies

Figure 7 shows what happens when an end user calls an application with reservable licenses, when using the named user policy (see "Named User Policy" on page 10 for more information).
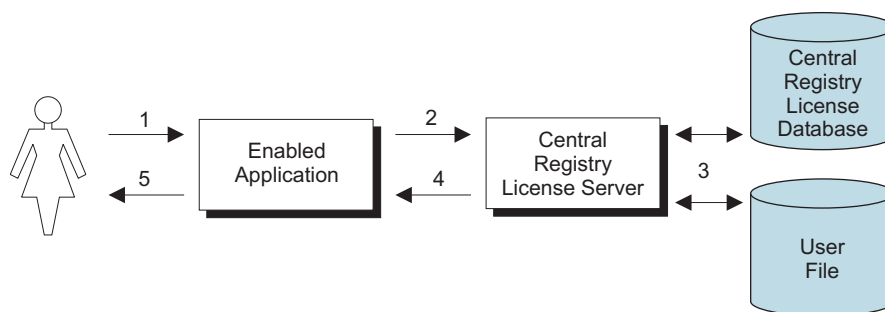
Figure 7. Using a Reservable License in a Named User Policy

1. The user calls the application.
2. 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.
3. The central registry license server checks its database for a reserved license and the user file for authorization.
4. The central registry license server returns the status of the request to the application.
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 Per-Server Licenses

Figure 8 shows what happens when an end user calls an application with per-server licenses when per-seat has not been enabled. In the figure, the application server is license-enabled.
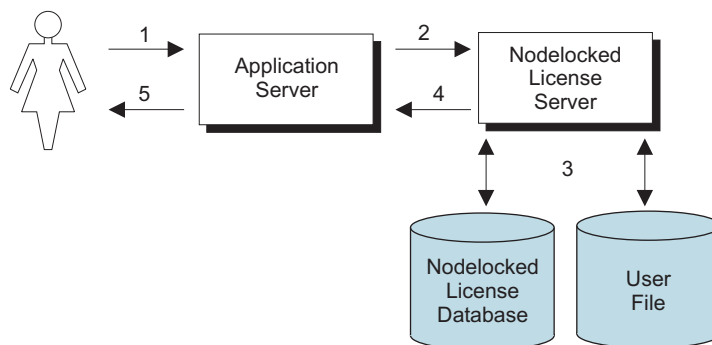
Figure 8. Using a Per-Server License

1. The application client user calls 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 is found, or if a soft stop license is 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 Per-Seat Licenses

Figure 9 shows what happens when an end user calls an application with per-server/per-seat licenses when per-seat has been enabled. In the figure, the application server is license-enabled.
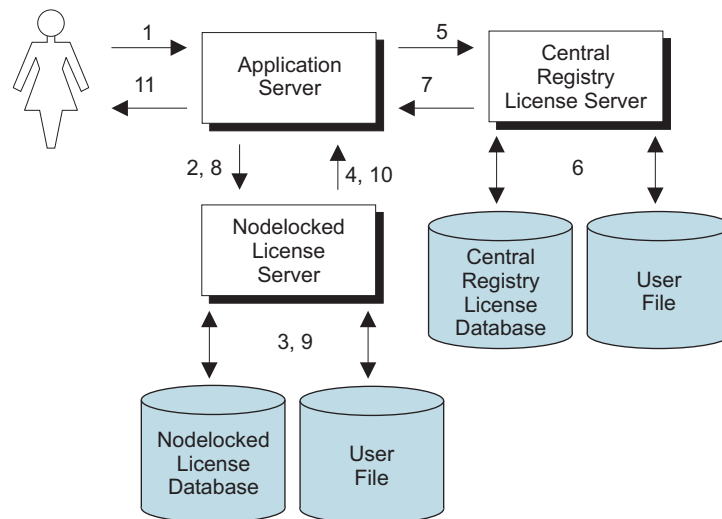
Figure 9. Using a Per-Seat License

1. The application client user calls 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 workstation.

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 is found, the application runs.

5. If no per-seat license is 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 is 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.

Chapter 2. Planning Network Licensing

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

- You are using applications with nodelocked licenses only
- You do not intend to use the basic license tool from one workstation to administer licenses on other workstations

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 Management Runtime in large networks or across subnetworks.

The decisions you need to make include the following:

- How many network license servers you will set up
- Which workstations, 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
- How clients will locate the servers
- Whether all servers will serve all clients, or whether you will set up independent groups of servers and clients
- Which workstation will be the central registry (if required)
- Whether you will make use of License Use Management Runtime high-availability licensing; and, 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 workstation the network and nodelocked licenses on other workstations
- Whether your network will be configured with firewalls.

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.

In designating workstations 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. Workstations 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.

- 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 occur 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. "Selecting a Type of Network Binding" on page 31 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 its tools are included in the License Use Management Runtime components.

NCS includes the following tools:
- Remote Procedure Call (RPC) Runtime Library.

  The backbone of the network computing system. It provides the calls that enable local programs to run procedures on remote hosts. These calls transfer requests and responses between clients (the programs that call the procedures), and servers (the programs that run 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.
- Location Broker.

  The location broker processes, as discussed under "Namespace Binding" on page 32, and tools to administer them.

After configuration, these mechanisms are invisible to the end user of the software product.

Installing on a Workstation where NCS is Already Installed

UNIX

When you install License Use Runtime, the NCS shipped as part of License Use Runtime does not replace any other NCS that is already installed on the workstation. Instead, it is installed in parallel directories.

At any given time, you can have one set of NCS processes active. It is strongly recommended that you select as License Use Runtime servers workstations that are not servers for any license management product. If you configure a workstation as a License Use Runtime server, whenever an NCS process is started, any corresponding process that is already running is stopped.

A License Use Runtime Server can provide licenses only for products that are enabled using the License Use Management Application Developer's Toolkit. It cannot provide licenses that require any other license management product, such as iFOR/LS or NetLS, that may be installed on the system.

On client workstations you can install products that were enabled using License Use Management, iFOR/LS, or NetLS. The client part of iFOR/LS or NetLS can coexist with the client part of License Use Management on the same system. You should configure License Use Runtime clients to work with License Use Runtime servers only.

Selecting a Type of Network Binding

ALL

License Use Management Runtime provides two types of network configuration to enable clients to locate (or *bind to*) the network license servers and the central registry. They are direct binding and namespace binding. On Windows, *only* direct binding is available.

Direct Binding

Direct binding is the simpler binding mechanism, suitable for small networks and for networks that do not change frequently. For environments, with one or two network license servers, direct binding provides a simple, effective licensing environment.

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 71.

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

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

Table 4. Communication Ports Used

Protocol	Ports	Operating Environments and Systems
UDP/IP	1515, 10999, 12999	Windows 2000, Windows XP and Windows Server 2003, Solaris, HP-UX, AIX

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 to 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.

The Basic License Tool uses the 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 workstation configured only as a nodelocked license server. When you configure a nodelocked license server, you can create a direct binding servers list that contains all the license servers (both network and nodelocked) whose licenses you want to administer remotely.

Namespace Binding

As the licensing environment increases, keeping the direct binding environment up-to-date becomes more complex, and namespace binding becomes the better way to manage the license use management environment. Namespace binding is a powerful method of administering large client/server networks and networks that change frequently.

With namespace binding, one or more network license servers must run a subsystem called the *global location broker*. All the network license servers register themselves with the global location broker. The global location broker maintains a database of all the network license servers and the license-enabled products for which they have licenses. When a client requests a license, the global location broker locates a server for the client.

The client workstation does not need to have a list of all the network license servers. It needs only the address of a server on which the global location broker runs.

The global location broker dynamically updates network location information for each license server. If you configure new license servers, or move existing license servers to new locations on the network, licensed applications will always be able to find them.

You may want to set up your namespace binding environment so that some of the servers serve only some of the clients. Such a grouping of clients and servers is called an NCS cell, or just a cell.

When a network license client requests a license, only a license server in the same cell as the client can satisfy the request.

If multiple servers in the cell have licenses for the product, the servers are checked for an available license in random sequence. This automatically balances the workload of the servers.

The cell is analogous to the direct binding servers list (see "Direct Binding" on page 31). You should configure each server as part of a direct binding servers list or a namespace binding cell, but not both.

Network license servers configured using namespace binding cells can support clients that locate the server through either namespace binding or direct binding.

In addition to enabling clients to locate license servers, the namespace binding mechanism makes it possible to use the Basic License Tool to do remote administration of licenses on all the servers in the cell.

Performance Considerations in Choosing a Binding Mechanism

Assuming that licenses are equally spread over all the license servers and that all servers belong to the same licensing environment, then do the following:

- If namespace binding is used the license workload is automatically balanced between the different servers.
- If direct binding is used then the workload balance is not automatic. The license system administrator must simulate it arranging the clients' Direct Binding list so that each license server occupies different positions on the same number of clients.

The same concept applies to the list of servers running a global location broker listed in the `glb_site.txt` file when using namespace binding, and clients or servers are located in more than two subnetworks.

In all cases in which there is discretion in deciding the order of the servers, consider the following:

- What has to be balanced is not the number of client workstations, but the number of application clients; if more than one enabled application runs on a workstation this must be taken into account.
- The frequency of license requests and license checks is relevant. That is, applications that frequently request and check licenses impact server performance more than applications that request and check licenses in longer time intervals. Fast applications that are frequently run belong to this category.
- In a normal environment, clients reach servers on the same subnetwork more quickly.

On the client side it is possible to distinguish two scenarios:

- Some clients are configured in namespace binding only and others in direct binding only.

  For the clients configured in namespace binding, workload balancing is automatic. For the clients configured in direct binding, workload balancing over multiple servers is not automatic, you must simulate it. This can be done by varying the order of the servers in the clients' direct binding list so that the servers occupy different positions on the same number of clients. This scenario makes sense if some clients need to contact only a subset of the license servers in the licensing system. These clients may be configured in direct binding to contact only the servers they actually need to reach.

- Namespace binding and direct binding coexist in the same client.

  In this case the same client is configured both in namespace binding and in direct binding. If all license servers listed in the direct binding list are located in the same NCS cell, it makes no difference if the client is configured in direct

binding mode. The only difference is that the servers in the direct binding list are contacted before the global location broker is contacted, and this may impact the workload balancing.

Planning Direct Binding

ALL

Before you begin configuring workstations to use direct binding, be sure you have identified all the servers in your direct binding servers list and refer to the configuration scenarios in"Setting Up Your Servers and Clients" on page 71. As you configure direct binding at each server and at each client, be sure you enter exactly the same list of servers.

It is important that the direct binding servers list include all the servers, and that it excludes workstations that will not function as servers. If there are any workstations that are not servers 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 in which 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 all the licenses requested by a particular network license client are to 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. This will 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 obtain licenses.

Planning Namespace Binding

UNIX

In setting up namespace binding, you must decide:
- How you will group the servers and clients
- Which servers will run the location brokers and other NCS tools, within each group.

Note: Namespace binding is not supported in systems protected by firewall.

Planning Cells

In namespace binding, all nodes belonging to a cell are identified by a single universal unique identifier (UUID). The UUID is a 36-byte string that identifies the host on which it was created and the time at which it was created.

A node cannot be in more than one cell.

You can, optionally, place servers in the default cell (that has a default UUID). If a network license client is configured for namespace binding, and is not configured as part of another cell, it joins the default cell.

You can create cells to isolate individual departments or other groups of users. Be careful that different NCS users at your location do not inadvertently create two or

more default cells. Because the cells would have the same UUID, they would not be isolated from one another, and results would be unpredictable.

Because cells cannot overlap, it is important to understand who should have access to which licenses before you configure your servers. In a production environment, you may want to configure all your license servers to run in the default cell. This simplifies the task of managing servers and allows a central administrator to control all the license servers. However, if some licenses are to be restricted to a certain group of users, you can install those licenses on servers running in another cell, and make the clients that use the licenses part of that cell.

It is good practice to create cells dedicated to a test environment, because the unstable nature of the environment could negatively affect regular production users.

When you have decided which servers and clients belong to each cell, see the configuration scenarios in "Setting Up Your Servers and Clients" on page 71 for information on how to configure your environment. As you configure namespace binding at each server and at each client, place it in the selected cell.

If you are setting up License Use Management Runtime on a workstation that is not on a network, but you plan to use license-enabled products with network licenses, you need to have a network license server running on this workstation. In this case, it is best to configure NCS to be in its own cell.

Selecting the Global Location Brokers

If the network is small to medium and has high-speed connections throughout, one global location broker is probably sufficient. Choose one of the network license servers to run the global location broker. If the network is large, it might be best to set up one server that runs the global location broker on each LAN.

When you are deciding which workstations should run the global location broker, keep in mind that the process runs continuously in the background, waiting for a request for the function it provides. The function it provides is called infrequently. It is usually in wait state and has little effect on system performance.

In a namespace binding environment, each network license server and the central registry license server, including systems that run the global location broker, run a process called the *local location broker*. The local location broker handles communication with the global location broker. When you configure a network license server, you specify whether it is to run just the local location broker or additionally, the global location broker.

Running the Location Brokers

For information about configuring and starting the location broker processes, see "Setting Up Your Servers and Clients" on page 71. License Use Management Runtime provides tools to administer the location brokers. For information about using these tools, see "Using NCS Tools" on page 36, and Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Running the Global Location Broker Database Cleaner

The global location broker database cleaner is a process that should always be active; it automatically and periodically cleans up the global location broker databases.

When a license server starts up, it registers itself to the global location broker. When the server stops, it deletes itself from the global location broker database. Should the server accidentally go down without being able to deregister itself, invalid entries remain in the global location broker database. The global location broker database cleaner deletes them.

Using NCS Tools

NCS provides the tools given in the following table, which you can optionally use to administer your namespace binding environment.

Table 5. NCS Tools

Tool	Description
Local Broker Administration (lb_admin)	Administers the registration of the servers in global location broker or local location broker databases. It can be used to look up information, add new entries, and delete existing entries in a specified database.
GLBD Replicas Administration (drm_admin)	Monitors and modifies the list of the replicated versions of the global location broker databases. It can be used to modify, or merge databases to force convergence among replicas, to stop servers, and to delete replicas. **Note:** The glbd replica is not available on Linux operating systems.
GLBDs List (lb_find)	Lists the servers running the global location broker in the network.
Universal Unique Identifier generator (uuid_gen)	Generates the UUID for an NCS cell.

For a detailed explanation of how to use these tools, see Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Reaching a Global Location Broker in a Different Subnetwork

Normally, products on network license clients contact a global location broker by broadcasting on the local network. If your system does not support broadcasting, or if the global location broker is running on a license server in a separate subnetwork, you need to set up an alternative mechanism to enable the workstation to locate a global location broker. The mechanism is the file glb_site.txt, which you create on the workstation that needs to reach a global location broker.

The glb_site.txt file lists the network addresses of servers where a global location broker may be running. A workstation that has a glb_site.txt file tries these addresses in order. Once it locates a server that is running the global location broker, it can locate network license servers. If it does not locate a server that is running the global location broker, the workstation does not broadcast.

For information about how to create the glb_site.txt file, see "Configuring to Reach a Global Location Broker in a Different Subnetwork" on page 93.

Planning the Central Registry

ALL

The central registry license server subsystem 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. For namespace binding, only one central registry subsystem must be running in a cell. For direct binding, only one central registry subsystem must be running in a direct binding servers list. This ensures that the data is accurate and complete.

Notes:

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 workstation 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 Clusters for High-Availability Licensing

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.

For 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 workstation.

Different license servers (both in terms of License Use Management versions and operating systems) can coexist in the same cluster.

Restrictions on Cluster Size and Composition

For security reasons, it is necessary to impose strict rules on the size and composition of clusters.

Attention: 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 as long as it exists.

When you create a cluster, 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

If you want to replace a server workstation 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.

Note: 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 6.

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 6. Number of Servers in a Cluster

Initial Number	Minimum Activated for Cluster to Work	Maximum Added after Cluster Creation	Actual Activated	Minimum Up and Running for Cluster to Work
3	2	1	2	2
			3	2
			4	3
4	4	2	4	3
			5	3
			6	4
5	4	1	4	3
			5	3
			6	4
6	6	2	6	4
			7	4
			8	5
7	6	1	6	4
			7	4
			8	5
8	8	2	8	5
			9	5
			10	6

Table 6. Number of Servers in a Cluster (continued)

Initial Number	Minimum Activated for Cluster to Work	Maximum Added after Cluster Creation	Actual Activated	Minimum Up and Running for Cluster to Work
9	8	1	8	5
			9	5
			10	6
10	10	2	10	6
			11	6
			12	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, but the first two servers declared in the cluster cannot be replaced (due to security exposure). This means you can add a new server with upgraded hardware, and deactivate the original third declared server, only once during the life of the cluster. This scenario has the effect of replacing a server with an upgraded workstation. 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.

  The relationships between these cluster attributes is shown in Table 6 on page 38.

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

Table 7 shows how the servers are deployed, depending on how many are activated, and how many are up and running.

Table 7. 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*
	2	2	0*
3	3	2	1
	3	3	0*
4	4	3	1

* 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, but the first six servers declared in the cluster cannot be replaced (due to security exposure).

  The relationships between these cluster attributes is shown in Table 6 on page 38.
- 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 8 shows how the servers are deployed, depending on how many are activated, and how many are up and running.

Table 8. 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

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

Cluster Membership Considerations

If you use direct binding, and you want to take full advantage of the high availability of licenses, make sure the following have the cluster servers in their direct server binding lists:

1. The network clients that will use licenses bound to the cluster
2. All the servers of the cluster

If you use namespace binding, all the servers in a cluster and all their network clients must be in the same cell in order to exploit fully the high availability of licenses.

If you use HAL with remote administration disabled, there are some important points to take into consideration:

- All cluster members must adhere to the same remote administration policy (that is, either all disabled, or all enabled).
- From a remote administration point of view, the cluster behaves as a single server.
- If remote administration is disabled, it is not possible to enroll or delete licenses from a machine that is not a cluster member.
- A cluster can be created only if the request comes from the local node. Cluster members can be deactivated only from another cluster member.

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.

Notes:

1. Either disable or enable remote administration on all servers. If you enable or disable it only on some servers, you may have problems enrolling and removing licenses bound to the cluster.
2. Due to incompatibility between HAL cluster members 4.6.4 and HAL cluster members of previous releases of License Use Management Runtime, you cannot create a cluster using a mixed server environment. All HAL cluster members have to be 4.6.4 or higher.

Firewall Support

If a server runs behind a firewall with respect to the client side, the client-server communications are prevented. This method increases the level of security on those special company assets which are represented by the LUM License Electronic Certificates. In this way nobody outside the Local Area Network of the company that purchased the certificates can have a license granted from the company server.

License server communications take place on two ports. The main server port is defined (**ipPort=** in the configuration file), however, server responses to client requests for authentication take place on a secondary, randomly chosen, port. This is why client/server communications cannot take place through a firewall. If the port number is defined, communications can take place through a firewall, once the firewall administrator has applied proper rules to UDP packets flowing to/from the license server on that port number.

Firewall support has been introduced, based on the customization of the server authentication port. This is represented by a new **ipAuthPort=** tag in the configuration file. By default the port number is randomly chosen, but if a specific value is set then the authentication server port will be fixed to that value, for example:

```
ipAuthPort=1516
```

Using this method, it is possible to install the LUM license server inside the company LAN, protected by the firewall, and to enable the incoming connections of LUM clients from, for example, a DeMilitarized Zone (DMZ), by opening the firewall on the two ports used by the server. Packets directed the opposite way, from the LAN to the DMZ, must not be filtered by the firewall.

When configured to support clients through a firewall License administration is permitted only when the Basic License Tool is run on the same machine as the license server. When multiple servers are connected to set up a High Availability Licensing cluster environment, all the server members of the cluster must be placed in the same side of the firewall.

The main server port number can be changed through the panels of the LUM Configuration tool or its command line interface. However, the authentication port number can be changed only by editing the i4ls.ini configuration file.

Only the network license server can use firewall support, the central registry license server is excluded from using a firewall.

Namespace binding is not supported in a firewall system.

Verifying Network Connections

License Use Management Runtime provides the i4tv tool to verify that network license servers, and the central registry, are running properly. For information on how to use the tool, see "i4tv - Test Verification Tool" on page 173.

Network Examples

ALL

This section shows some of the possible network configurations you can have in your environment. For simplicity, the examples show an environment with at most five network license clients, three network license servers, and two nodelocked license servers.

Figure 10 shows a configuration where all the required NCS subsystems run on the same server.
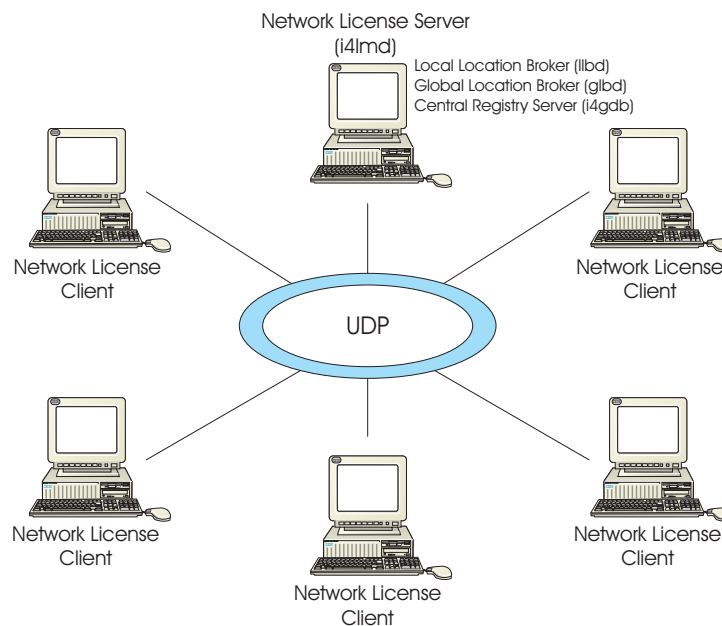
Figure 10. NCS Cell with All the Subsystems on the Same Server

Figure 11 on page 43 shows a network with two network license servers (A and D) and two nodelocked license servers (B and C). One network license server, and both nodelocked license servers, run only the local location broker, which is mandatory on all servers. One network license server also runs the central registry and the global location broker. From any of the license servers, it is possible to administer licenses on all the license servers.
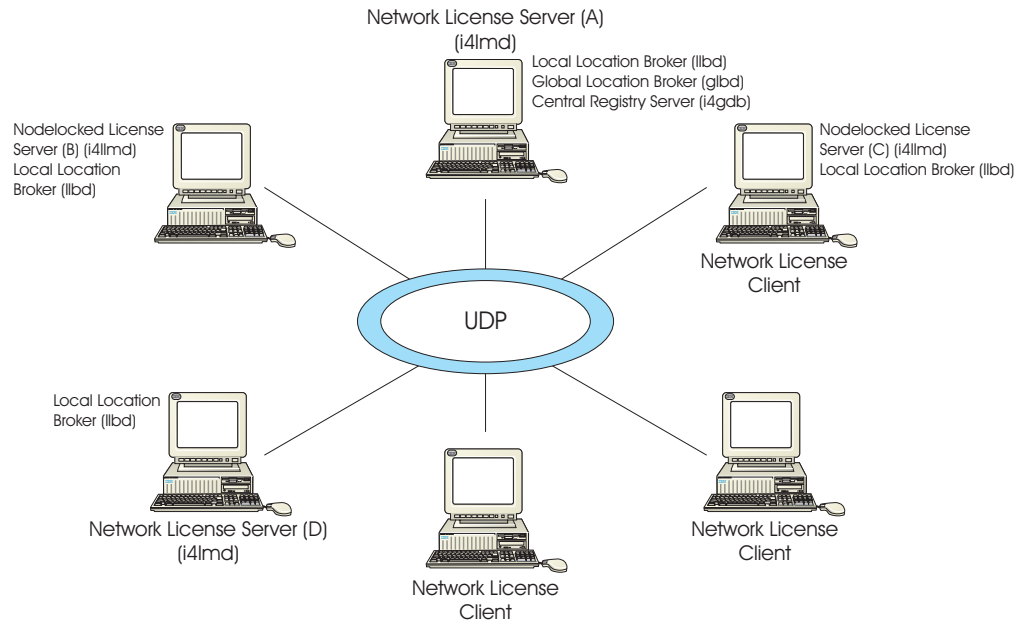
Figure 11. NCS Cell with Network License Servers and Nodelocked License Servers

Figure 12 shows a network with three network license servers. This example shows that more than one license server in the network can run the global location broker, but with only one central registry. Server C runs a global location broker that is a replica of the first one that was started on server B.
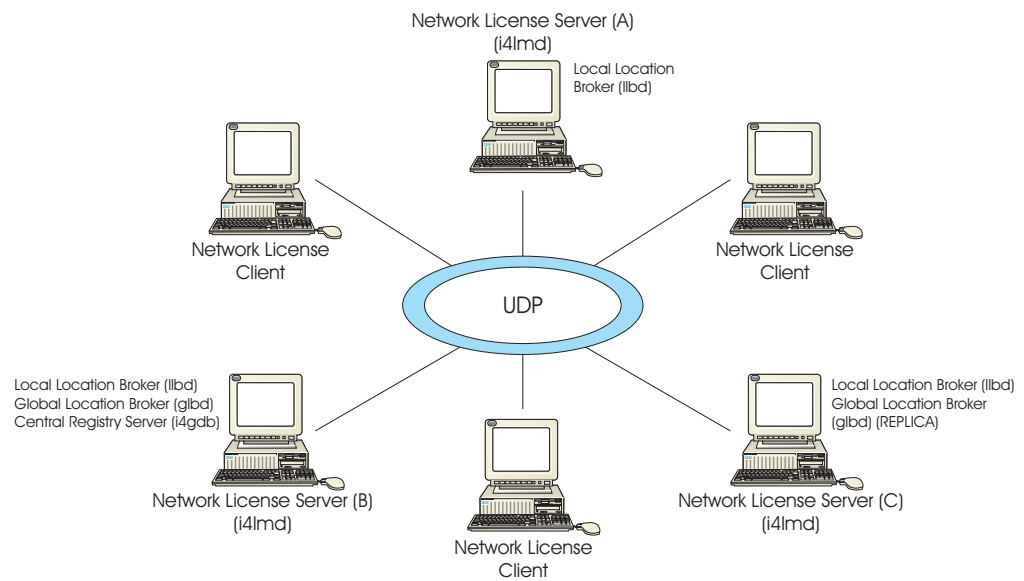
Figure 12. NCS Cell with Three Network License Servers and Three Clients

Figure 13 on page 44 shows an 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 the license servers.
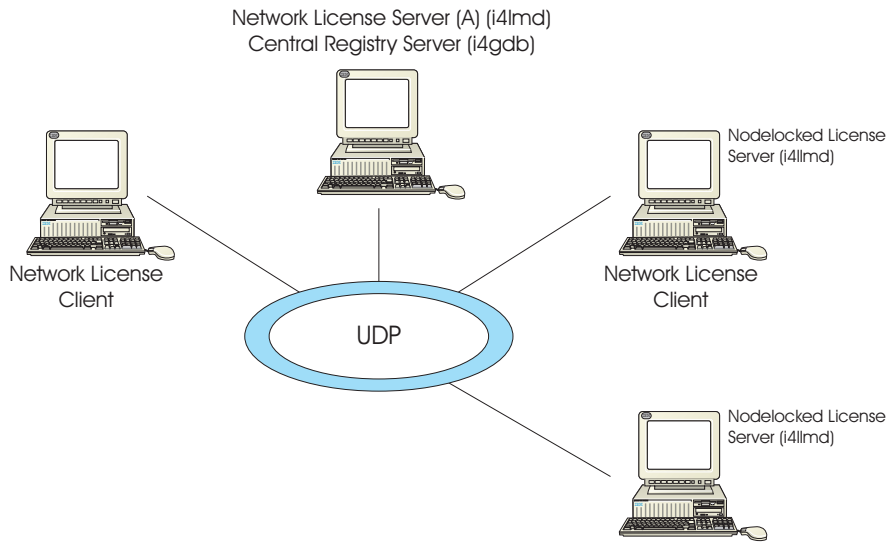
Figure 13. Direct Binding with Network License Servers and Nodelocked License Servers

Figure 14 shows an example of a network configuration that uses a firewall. The example shows a network consisting of a network license server and three network license clients on the protected side of a firewall, and two network license clients on the unprotected side of a firewall. All network license clients run on the same network license server inside the firewall, or on different servers outside the firewall.
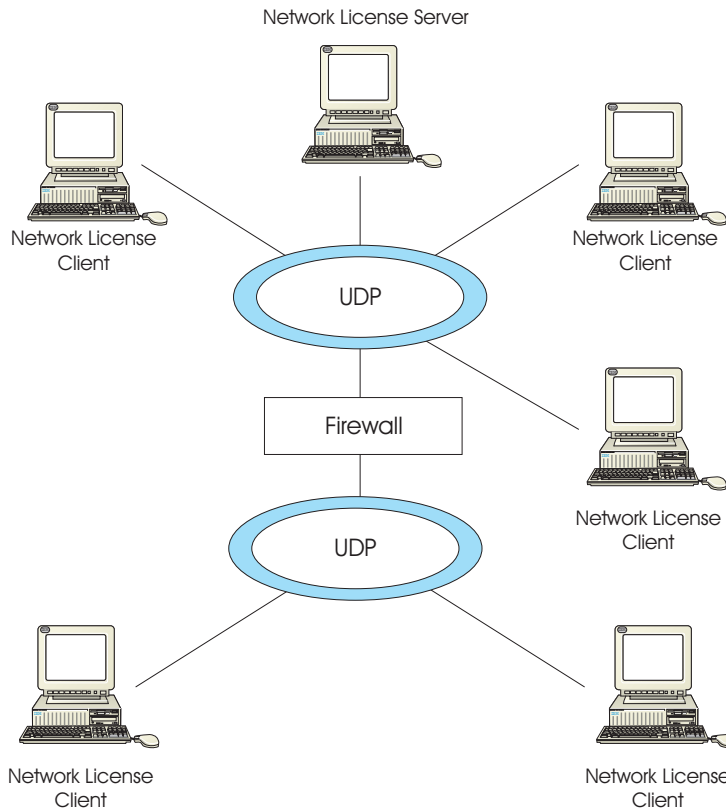
Figure 14. Direct Binding with Network License Servers, Firewall, and Network License Clients

Chapter 3. Installing License Use Management Runtime

This chapter is intended for administrators that are using license-enabled products and for software programmers who are developing license-enabled products. The following sections show you how to install License Use Management Runtime on UNIX, AIX, and Windows.

Attention: If any other license management product is installed on the machines in your environment, you must be aware of certain requirements before you install. Be sure you read Chapter 2, "Planning Network Licensing," on page 29 carefully. Considerations for coexistence with other license management products are in the sections "Selecting Your Servers" on page 29, "Installing on a Workstation where NCS is Already Installed" on page 31, "Direct Binding" on page 31, and "Planning Cells" on page 34.

Installing License Use Management Runtime on AIX

This section shows you how to install License Use Management Runtime and the License Use Management Runtime Java GUI on AIX. The License Use Management Runtime base code and command line interface are included in the basic operating system components installed on every machine. This section shows you how to install the latest version of License Use Management Runtime on AIX.

Note: If the License Use Management Runtime code is incorporated in the license-enabled product, you can install License Use Management Runtime packages as part of the installation of the licensed product, rather than by using the procedures in this section.

Before Installing License Use Management Runtime Packages

This section identifies the components of License Use Management Runtime and gives their disk space requirements. It also gives the hardware and software requirements for License Use Management Runtime. This section explains how to determine the level of License Use Management Runtime installed on a system. It explains how to determine whether you need to install License Use Management Runtime.

License Use Management Runtime Packages, Components, Disk Space Requirements

Before installing License Use Management Runtime, allocate sufficient disk space in the /usr file system. Table 9 identifies the components of License Use Management Runtime and gives their disk space requirements.

Table 9. License Use Management Runtime Packages, their Components, and their Disk Space Requirements on AIX

Package	License Use Management Runtime Component	Disk Space
ifor_ls.base.cli	Base code and command-line interface	15 MB
ifor_ls.java.gui	Base Java GUI	7 MB
Note: The default directory for these packages is /usr/sys/inst.images		

License Use Management Runtime also requires space in the /var file system for databases, log files, and trace output. If you are using several licensed products and there is extensive activity on your system, License Use Management Runtime might need 10 MB or more in /var.

For an overview of the packages, see "Installing License Use Management Runtime Components" on page 47.

Hardware and Software Requirements

License Use Management Runtime requires a system that can run AIX.

Before you install License Use Management Runtime, Network Computing System (NCS) 1.5.1 must be installed on your system. You can only install Fix pack 12 if License Use Management Runtime 4.6.8 is installed on your machine.

You can also download NCS 1.5.1 from the License Use Management Runtime Web site:

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

For AIX 5.1, 5.2, 5.3, "IBM XL C/C++ Enterprise Edition V8.0 for AIX, Runtime Environment and Utilities" package needs to be installed before License Use Management Runtime 4.6.8 Fix Pack 12 can be installed. This package can be downloaded from:

http://www-1.ibm.com/support/docview.wss?rs=0&ql=XL+C%2fC%2b %2b+V8.0+runtime&uid=swg24010670&loc=en_US&cs=utf-8ç=us(=en

Your Web browser is required to view .htm documentation.

GUI Requirements: Java Runtime Environment, as follows:
- AIX 5.1 requires Java Runtime Environment 1.3.1, or later.
- AIX 5.2 requires Java Runtime Environment 1.3.1, or later.
- AIX 5.3 requires Java Runtime Environment 1.4.0, or later.

The License Use Management Runtime Java GUI requires a minimum of 192 MB of RAM.

Determining the Level of License Use Management Runtime Installed

If you need to determine the level of License Use Management Runtime Version 4 already installed on your workstation, check the file:
/var/ifor/VERSION

Alternatively, use the smitty List Installed Software panel. Next to each License Use Management Runtime fileset:
- 5.1.0.0 indicates License Use Management Runtime Version 4.6.2
- 5.1.0.10 indicates License Use Management Runtime Version 4.6.2 with AIX APAR IY21215 (14th September 2001)
- 5.1.0.15 indicates License Use Management Runtime Version 4.6.4 installed with AIX 5.1.0.15
- 5.1.0.16 indicates License Use Management Runtime Version 4.6.4 with AIX 5.1.0.16 (19 December 2001)

- 5.1.0.18 indicates License Use Management Runtime Version 4.6.4 available from the Web
- 5.1.0.25 indicates License Use Management Runtime Version 4.6.5 installed with AIX 5.1.0.25
- 5.1.0.26 indicates License Use Management Runtime Version 4.6.5 available from the Web
- 5.2.0.0 indicates License Use Management Runtime Version 4.6.6 installed with AIX 5.2.0.0
- 5.2.0.1 indicates License Use Management Runtime Version 4.6.6 available from the Web
- 5.2.0.10 indicates License Use Management Runtime Version 4.6.6 installed with AIX 5.2.0.10
- 5.2.0.11 indicates License Use Management Runtime Version 4.6.7 available from the Web
- 5.2.0.30 indicates License Use Management Runtime Version 4.6.7 installed with AIX 5.2.0.30
- 5.2.0.40 indicates License Use Management Runtime Version 4.6.8 installed with AIX 5.2.0.40
- 5.3.0.0 indicates License Use Management Runtime Version 4.6.8 installed with AIX 5.3.0.0
- 5.3.7.0 indicates License Use Management Runtime Version 4.6.8.10 installed with AIX 5.3.7.0 or higher (AIX 5.3 TL7 or higher)
- 6.1.0.0 indicates License Use Management Runtime Version 4.6.8.10 installed with AIX 6.1.0.0 or higher (AIX 6.1 TL0 or higher)
- 6.1.0.21 indicates License Use Management Runtime Version 4.6.8.12 downloaded from the License Use Management support web site.

Installing License Use Management Runtime Components

This section outlines the content of the License Use Management Runtime packages, summarizes the installation procedure, and provides instructions on what to install from which package.

You can install License Use Management Runtime from the AIX installation media. Before you install License Use Management Runtime packages, make a backup copy of the originals.

Note: You can only upgrade from License Use Management Runtime 4.6.8 to License Use Management Runtime 4.6.8.12. If you are migrating to Licence Use Management Runtime Version 4.6.8.12, make sure that you save your log databases in a report. Due to incompatibility between the structure of the new and the old log files, you can only read these files if you reinstall the previous version. During Licence Use Management Runtime Version 4.6.8.12 installation, the log files (logdb*.dat, logdb*.idx, crlog*.dat, crlog*.idx, llmlg*.dat, llmlg*.idx) are moved to the /var/ifor/oldlogs directory.

Log in with root authority, and use the smit command to install License Use Management Runtime from the following packages:
- ifor_ls.base.cli. This is the runtime code and the command line interface for the Basic License Tool and the Configuration Tool.
- ifor_ls.java.gui. The Java graphical user interface for the basic license tool and the configuration tool.

For information about License Use Management Runtime packages in languages other than US English, see .

The sequence of `smit` panels to follow is:
1. Software Installation and Maintenance
2. Install and Update Software
3. Install Software

If you do not have a graphical terminal, use the `smitty` command.

Setting Up the .profile File

After installation, and before starting License Use Management Runtime, log in with root authority and check that there is a .profile file. If there is not, create it and add the following path to the .profile:

`PATH=/usr/opt/ifor/ls/os/aix/bin:$PATH`

The default directory for the .profile file of the user **root** is /. License Use Management Runtime looks for this file and automatically updates it, if it is found. However the user **root** can choose to change the location of its .profile file. In this case License Use Management Runtime cannot perform the automatic update and it must be done manually by the user **root**. A typical path for the root .profile file is `/home/root`.

You may also need to add a path for the man pages. For details, see "MAN Pages for Command Syntax" on page xiii.

Exit and log in again for the changes to take effect.

Uninstalling License Use Management Runtime Packages

This section describes how to remove the License Use Management Runtime packages from your workstations:

The GUI filesets can be removed without removing the base filesets. Removal of the base filesets requires the removal of the GUI filesets.

Uninstall the packages and filesets in the following sequence:
1. The License Use Management Runtime base, ifor_ls.base.
2. NCS.

The uninstallation procedure does not delete the License Use Management Runtime databases, which are in the `/var/ifor` directory.

Use the `smit` command to uninstall License Use Management Runtime. If you do not have a graphical terminal, use the `smitty` command.

Installing License Use Management Runtime on HP-UX

UNIX

Licence Use Management Runtime Version 4.6.8.12 contains one package, named License Use Management - License Use Management Runtime Base Code.

Before installing License Use Management Runtime

To determine whether License Use Management Runtime Version 4.6.8 is already installed on your workstation, refer to the file:

`/var/lum/VERSION`

Before you install:
- Allocate at least 20 MB of disk space in the /opt file system.
- Delete any directory (not a symbolic link to a directory) that is named /var/lum.

Note: You can only migrate to Fixpack 12 from License Use Management 4.6.8. If you are migrating to Licence Use Management Runtime Version 4.6.8.12, make sure that you save your log databases in a report. Due to incompatibility between the structure of the new and the old log files, you can only read these files if you reinstall the previous version. During Licence Use Management Runtime Version 4.6.8.12 installation, the log files (logdb*.dat, logdb*.idx, crlog*.dat, crlog*.idx, llmlg*.dat, llmlg*.idx) are moved to the /opt/lum/ls/conf/oldlogs directory.

Hardware and Software Requirements

License Use Management Runtime requires no hardware other than a system that can run HP-UX 11.1.

The following patch is a prerequisite on applications that have been enabled with License Use Management Application Developer's Toolkit when running License Use Management Runtime:
- PHNE_29387

TCP/IP must be installed.

A Web browser is required to view the .HTM documentation.

Obtaining the License Use Management Runtime Code

You can get the License Use Management Runtime code:
- By downloading it from the Web. Download the TAR file from http://www.ibm.com/software/lum. Type the command `tar -xvf` to extract the file so that you have the package ready for installation.
- With an enabled application, if the vendor chose to redistribute the code.

Installing the License Use Management Runtime Code

Log in with root authority. Use SAM to install License Use Management Runtime as follows:

1. Click the **Software Management** icon.
2. Select **Install Software to Local Host** icon.
3. In the **Specify Source** window, set **Source Depot Type** to **Local Directory**. Set **Source Depot Path** to the directory where you have downloaded the code. Click **OK**.
4. Select **License Use Management Runtime Base Code** from the list of programs.
5. From **Actions**, select **Mark for Install**.
6. From **Actions**, select **Install (Analysis)**.
7. When the analysis is complete, click **OK**, then **Yes** to confirm installation, and then **Done**.

If you do not have a graphical terminal, use the `swinstall` command.

Note: If you migrate from an older Version of License Use Management Runtime 4.6.x to License Use Management Runtime Version 4.6.8.12 make sure that you installed Version 4.6.8.12 of the Java GUI package. Different versions of License Use Management Runtime and the Java GUI cannot coexist on the same machine.

For more information about installation in general, refer to the `swinstall` MAN pages.

After Installing License Use Management Runtime

After installation, before starting License Use Management Runtime, log in with root authority and add the following path to the `.profile` in the PATH environmental variable:

`/opt/lum/ls/os/hpux/bin`

Then exit and log in again to effect the changes.

Uninstalling License Use Management Runtime

To remove License Use Management Runtime from your workstation, log in with root authority. Use SAM to uninstall, as follows:

1. Click the **Software Management** icon.
2. Select **Remove Software** and then **Remove Local Host Software**.
3. Select **License Use Management Runtime Base Code** from the list of programs.
4. From **Actions**, select **Mark for Remove**.
5. From **Actions**, select **Remove (Analysis)**.
6. When the analysis is complete, click **OK**, then **Yes** to confirm uninstallation, and then **Done**.

If you do not have a graphical terminal, use the `swremove` command.

Installing the Java GUI on HP-UX

The Java GUI contains one package, named License Use Management Runtime Java GUI.

Before Installing the Java GUI

Before you install the Java GUI, allocate at least 7 MB of disk space in the `/opt` file system.

Hardware and Software Requirements

Java Runtime Environment 1.3.1 or later.

HP-UX 11.1.

Install License Use Management Runtime *before* installing the Java GUI.

Obtaining the Java GUI Code

You can get the Java GUI code:

- By downloading it from the Web. Download the TAR file from http://www.ibm.com/software/lum. Use the `tar -xvf` command to extract the file so that the Java GUI package is ready for installation.
- With an enabled application, if the vendor chose to redistribute the Java GUI code.

Installing the Java GUI Package

Log in with root authority. Use SAM to install the Java GUI, as follows:

1. Click the **Software Management** icon.
2. Click the **Install Software to Local Host** icon.
3. In the Specify Source window, set **Source Depot Type** to **Local Directory**. Set **Source Depot Path** to the directory where you have downloaded the code. Click **OK**.
4. Select **License Use Management Runtime Java GUI** from the list of programs.
5. From **Actions**, select **Mark for Install**.
6. From **Actions**, select **Install (Analysis)**.
7. When the analysis is complete, click **OK**, then **Yes** to confirm installation, and then **Done**.

If you do not have a graphical terminal, use the `swinstall` command.

Note: If you migrate from an older Version of the Java GUI 4.6.x to Java GUI 4.6.8 make sure that you install Version 4.6.8.12 of License Use Management Runtime. Different versions of License Use Management Runtime and the Java GUI cannot coexist on the same machine.

For more information about installation in general, refer to the `swinstall` MAN pages.

After Installing the Java GUI

After installation, before starting the Java GUI, log in with root authority and add the following path to the .profile in the SHLIB_PATH environmental variable:

`/opt/lum/ls/os/hpux/dll`

Then exit and log in again to effect the changes.

Uninstalling the Java GUI

To remove the Java GUI from your workstation:

1. Log in with root authority. Use SAM to uninstall the Java GUI, as follows:
2. Click the **Software Management** icon.
3. Select **Remove Software** and then **Remove Local Host Software**.

4. Select **License Use Management Runtime Java GUI** from the list of programs.

5. From **Actions**, select **Mark for Remove**.

6. From **Actions**, select **Remove (Analysis)**.

7. When the analysis is complete, click **OK**, then **Yes** to confirm uninstallation, and then **Done**.

If you do not have a graphical terminal, use the `swremove` command.

Installing License Use Management Runtime on Solaris

License Use Management Runtime Version 4.6.8 contains one package, named License Use Management - License Use Management Runtime Base Code.

Before Installing License Use Management Runtime

To determine whether License Use Management Runtime is already installed on your workstation, refer to the file:

`/var/lum/VERSION`

Before installing, allocate at least 23 MB of disk space in the `/opt` file system.

Notes:

1. If you are migrating from any previous version of License Use Management Runtime to Licence Use Management Runtime Version 4.6.8.12, make sure that you save your log databases in a report. Due to the incompatibility between the structure of the new and the old log files, you can only read these files if you reinstall the previous version. During Licence Use Management Runtime Version 4.6.8.12. installation, the log files (logdb*.dat, logdb*.idx, crlog*.dat, crlog*.idx, llmlg*.dat, llmlg*.idx) are moved to the `/opt/lum/ls/conf/oldlogs` directory.

2. If you migrate from an older version of the Java GUI 4.6.x to Java GUI 4.6.8 make sure that you install Version 4.6.8.12 of License Use Management Runtime. Different versions of License Use Management Runtime and the Java GUI cannot coexist on the same machine.

Note: Solaris 8, 9, and 10 are supported.

Hardware and Software Requirements

Solaris 8, 9 or 10 must be installed.

TCP/IP must be installed.

A Web browser is required to view the .HTML documentation.

Obtaining the License Use Management Runtime Code

You can get the License Use Management Runtime code:

- By downloading it from the Web. Download the TAR file from http://www.ibm.com/software/lum. Use the `tar -xvf` command to extract the file so that the package is ready for installation.
- With an enabled application, if the vendor chose to redistribute the code.

Installing the License Use Management Runtime Package

Follow these steps:

1. Log in with root authority. Use `admintool` to install the License Use Management Runtime package as follows:

2. From **Browse**, select **Software**.

If you have an earlier version of License Use Management Runtime, and you are now installing Version 4.6.8.12, select **Select Properties --> Package Administration**, and then change the value of **Existing Packages** to **Overwrite**.

3. From **Edit**, select **Add**.

4. In **Software Location**, select **Hard Disk**, and fill in the directory where you have downloaded the code.

5. Select **License Use Management Runtime Base Code**, and click **Add**.

If you do not have a graphical terminal, use the `pkgadd` command:

```
pkgadd -d
```

If you have an earlier version of License Use Management Runtime, and you are now installing Version 4.6.8.12, the `pkgadd` command should also contain the parameter:

```
-a lumadmin
```

Note: If you have installed the patch number 110934-20 for Solaris 10, and you are installing the License Use Management Runtime 4.6.8 , the `pkgadd` command with the parameter:

```
-a lumadmin
```

must be always used.

For more information about installation in general, refer to the *Solaris System Administration Answer Book*.

After Installing License Use Management Runtime

After installation, before starting, log in with root authority and add the following path to the .profile in the PATH environmental variable:

```
/opt/lum/ls/os/solaris/bin
```

Exit and login again to effect the changes.

Uninstalling License Use Management Runtime

To remove License Use Management Runtime from your workstation:

1. Log in with root authority. Use `admintool` to uninstall License Use Management Runtime as follows:

2. From **Browse**, select **Software**.

3. Select **License Use Management Runtime Base Code**.

4. From **Edit**, select **Delete**.

5. Select **Delete** to confirm, and **Y** when asked if you want to remove this package.

If you do not have a graphical terminal, use the `pkgrm` command.

Installing the Java GUI on Solaris

The Java GUI contains one package, named License Use Management Runtime Java GUI.

Before Installing the Java GUI

Before installing the Java GUI, allocate at least 40 MB of disk space in the /opt file system.

If you are installing on Solaris 8, download the required operating system patches from the following web site:

http://java.sun.com/j2se/1.3/install-solaris-patches.html

Hardware and Software Requirements

Solaris 8, 9 or 10.

License Use Management Runtime already installed.

Obtaining the Java GUI Code

You can get the Java GUI code:
- By downloading it from the Web. Download the TAR file from http://www.ibm.com/software/lum. Use the `tar -xvf` command to extract the file so that the Java GUI package is ready for installation.
- With an enabled application, if the vendor chose to redistribute the Java GUI code.

Installing the Java GUI Package

Follow these steps:

1. Log in with root authority.
Use `admintool` to install as follows:

2. From **Browse**, select **Software**.

3. From **Edit**, select **Add**.

4. In **Software Location**, select **Hard Disk**, and fill in the directory where you have downloaded the code.

5. Select **License Use Management Runtime Java GUI**, and click **Add**.

If you do not have a graphical terminal, use the `pkgadd` command:

```
pkgadd -d
```

Note: If you migrate from an older Version of the Java GUI 4.6.*x* to Java GUI 4.6.8 make sure that you install Version 4.6.8.12 of License Use Management Runtime. Different versions of License Use Management Runtime and the Java GUI cannot coexist on the same machine.

For more information about installation in general, refer to the *Solaris System Administration Answer Book*.

After Installing the Java GUI

After installation, before starting , log in with root authority and add the following path to the .profile in the LD_LIBRARY_PATH environmental variable:

```
/opt/lum/ls/os/solaris/dll
```

Then exit and login again to allow changes to take effect.

Uninstalling the Java GUI

To remove the Java GUI from your workstation:

1. Log in with root authority.
Use `admintool` to uninstall as follows:

2. From **Browse**, select **Software**.

3. Select **License Use Management Runtime Java GUI**.

4. From **Edit**, select **Delete**.

5. Select **Delete** to confirm, and **Y** when asked if you want to remove this package.

If you do not have a graphical terminal, use the `pkgrm` command.

Installing License Use Management Runtime on Windows

This section is intended for:

- The software vendor who needs to install License Use Management Runtime on Windows 2000, Windows XP, Windows Server 2003 or Windows Vista for use in license-enabling a product
- The customer of a license-enabled product if the vendor did not incorporate the License Use Management Runtime features into the product

Before You Install

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

Notes:

1. To install License Use Management 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 Management Runtime from another user ID, the installation program issues the message:

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

2. You can only upgrade from License Use Management Runtime 4.6.8 to License Use Management Runtime 4.6.8.12. If you have any previous version of License Use Management Runtime, you need to migrate to License Use Management Runtime 4.6.8 first. If you are migrating to Licence Use Management Runtime Version 4.6.8.12, make sure that you save your log databases in a report. Due to incompatibility between the structure of the new and the old log files, you can only read these files if you reinstall the previous version. During Licence Use Management Runtime Version 4.6.8.12 installation, the log files (logdb*.dat, logdb*.idx, crlog*.dat, crlog*.idx, llmlg*.dat, llmlg*.idx) are moved to the `c:\ifor\ls\conf\oldlogs` directory.

3. To use License Use Management tools on Windows Vista, you must increase the rights to the Administrator level, switching to the elevated mode. This behavior is introduces in the Windows Vista operating system only as a brand new concept of User Account Control (UAC). For more information see the Windows Vista website: http://www.microsoft.com/windows/products/windowsvista/default.mspx

To determine whether License Use Management Runtime is already installed on your workstation, refer to the following files:

Communications feature
```
c:\ifor\ls\conf\comver
```

Runtime feature
```
c:\ifor\ls\conf\runver
```

Selecting License Use Management Runtime Features

License Use Management Runtime consists of these features:

- Runtime
- Communications
- Documentation

Before you install on any workstation, you must decide which features to install. Your selection of features depends on the role the workstation will play in your licensing environment, as shown in Table 10. If your workstation will play more than one role in the licensing environment, be sure you install the features required for all the roles. For information about configuring your workstations to act in these roles, see "Setting Up Your Servers and Clients" on page 71.

Table 10. Required Features by Workstation Role

Role of the workstation	License Administration	Required Features
Nodelocked license server	Local administration only	Runtime
Nodelocked license server Network license server Network license client 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	No administration	Communications

Install the Documentation feature on any workstation to give you access to the online *Using License Use Management Runtime* and *Message Reference* books, in .HTML format for viewing with a Web browser.

Hardware Requirements

The hardware requirements for the various Windows platforms are shown in Table 11.

Table 11. Hardware Requirements

Minimum		Suggested	
Processor	**RAM**	**Processor**	**RAM**
233 MHz Pentium®® II	128 MB	1.5 GHz Pentium IV or higher	512 MB

Before installing, make sure that there is enough disk space for the features, as shown in Table 12 on page 57.

Table 12. Disk Space Requirements

Feature	Requirement
Java Gui	23 MB
Runtime	9 MB
Communications	6 MB
Documentation	3 MB

Software Requirements
- A Web browser in which you can view html documentation
- One of the following operating systems:
  - Windows 2000 with Service Pack 2
  - Windows XP with Service Pack 1
  - Windows Server 2003
  - Windows Vista
- For all supported Windows platforms:
  - A device driver that complies with network driver interface specification (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 Management Runtime is not yet installed on the workstation, you can download it from the License Use Management Web site:

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

    or the vendor of a license-enabled product can supply it to you.
- The TCP/IP transport protocol.

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

Obtaining the License Use Management Runtime Code
1. Download the self-extracting EXE or the compressed equivalent TAR file from:

   http://www.ibm.com/software/lum
2. Use the command `tar -xvf` to extract the TAR file.
3. The i4lct license creation tool is no longer included in the runtime package for Windows platforms. If you require the i4lct license creation tool, contact LUM Support.

Deciding How to Install
Before you install, you need to decide:
- Whether to install interactively, using the installation GUI (see"Installing License Use Management Runtime Interactively" on page 58)
- Whether to install noninteractively, using the setup command (see "Unattended License Use Management Runtime Installation" on page 67)

Installation Considerations
Windows Installer is used for installing License Use Management Runtime.

When you install a new release of License Use Management Runtime over an earlier release, Windows Installer first silently deletes the earlier release. It then

installs all, and only, those features that were previously installed. You cannot change the selection of features to be installed.

If License Use Management Runtime program files are damaged and need to be replaced, run the setup program again, and select the Repair option.

Installing License Use Management Runtime Interactively

To install License Use Management Runtime interactively, follow these steps:

1. Log on as Administrator.
2. Shut down any currently running programs, including virus scanning programs.
3. Start the installation program:
   - Switch to the directory to which you have downloaded the code, and run setup.

     The Welcome window is displayed (see Figure 15).
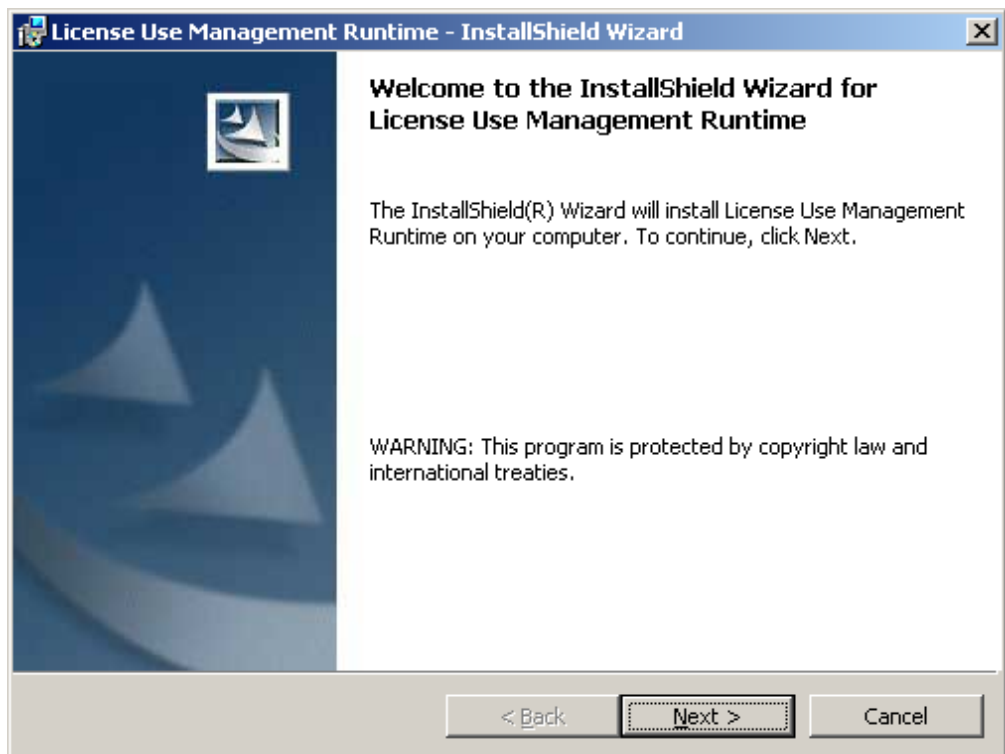
Figure 15. Welcome Window

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

   The License Agreement Window is displayed (see Figure 16 on page 59.
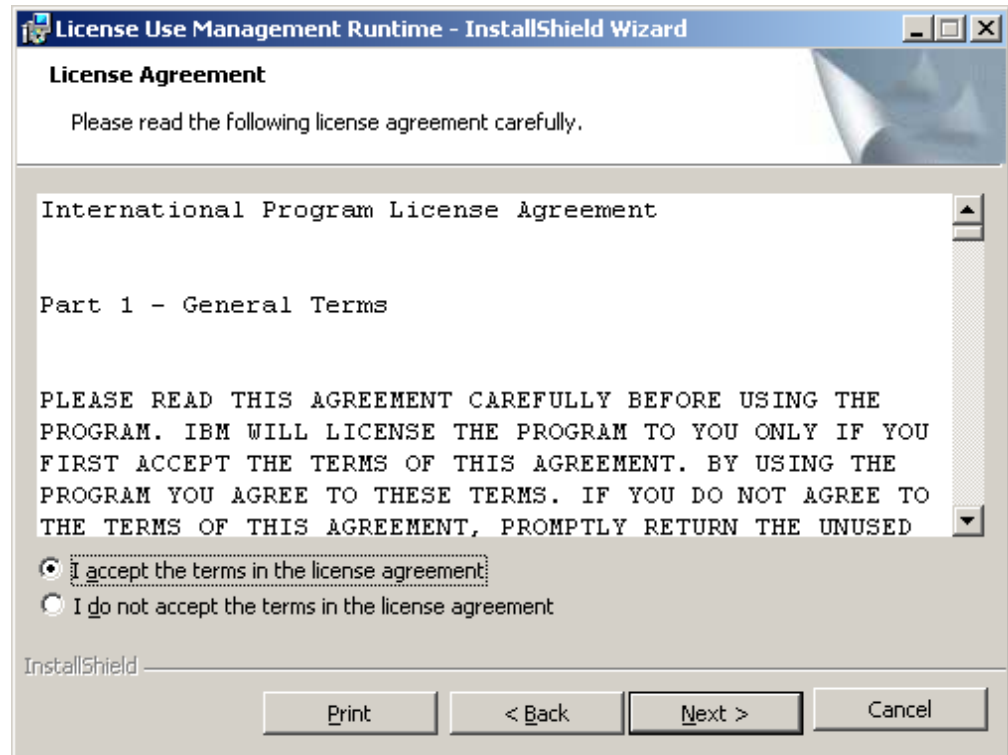
Figure 16. The License Agreement Window

5. In the License Agreement window, read the license agreement carefully and if
   you agree with its content choose **I accept the terms in the license agreement**
   and select **Next >**.

   The READ.ME file for License Use Management Runtime is displayed (see
   Figure 17 on page 60).
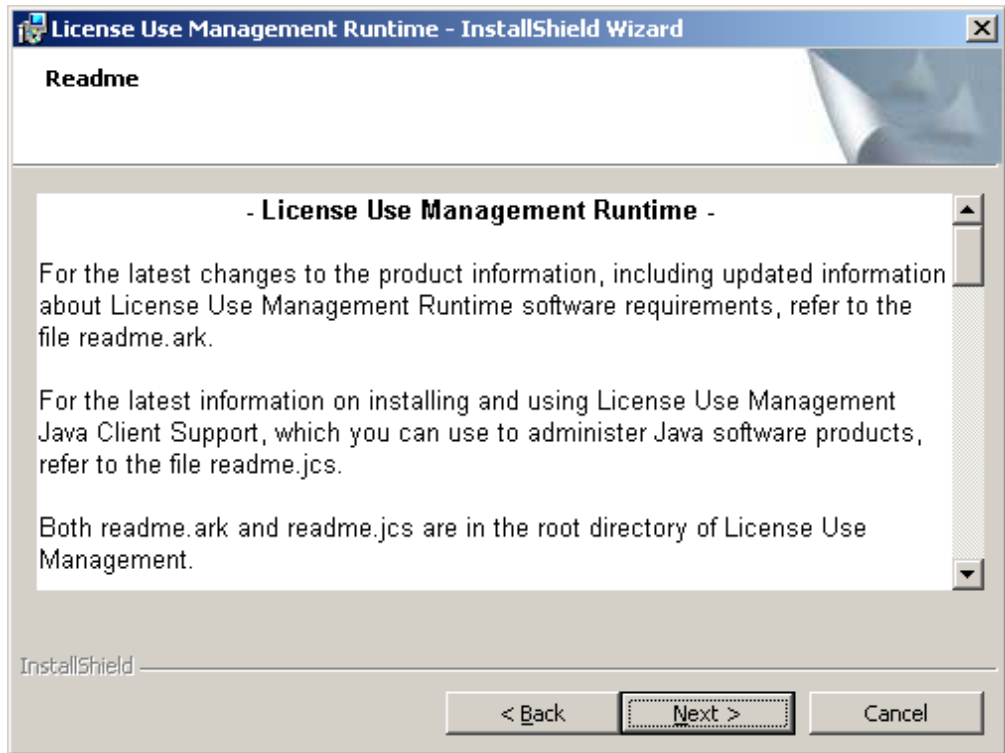
Figure 17. READ.ME Window. The text displayed on your monitor may differ from that shown here.

This file contains an overview of the installation procedure.

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

The Customer Information window is displayed (see Figure 18 on page 61).
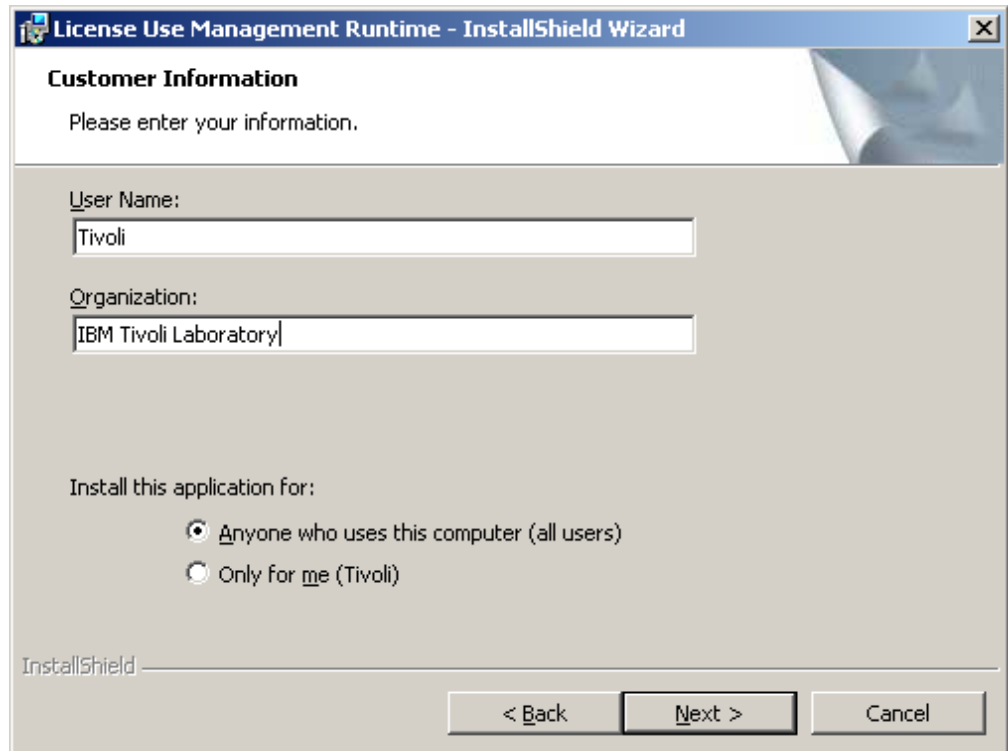
Figure 18. Customer Information Window

7. Enter your customer details, then click **Next >**. The Setup Type window is
   displayed (see Figure 19).
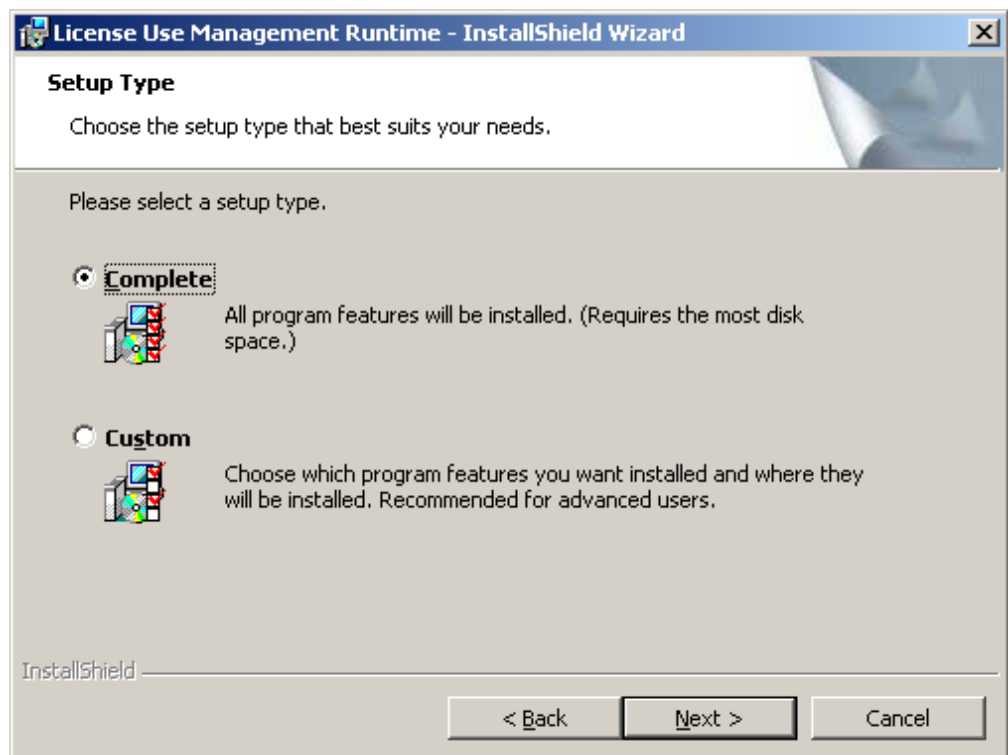
Figure 19. Setup Type Window

8. Choose **Complete** or **Custom** setup. If you choose Custom setup, you will be able to specify the disk drive on which to install License Use Management Runtime and the features to install.

9. Click **Next >**.

    The Custom Setup window is displayed (see Figure 20).
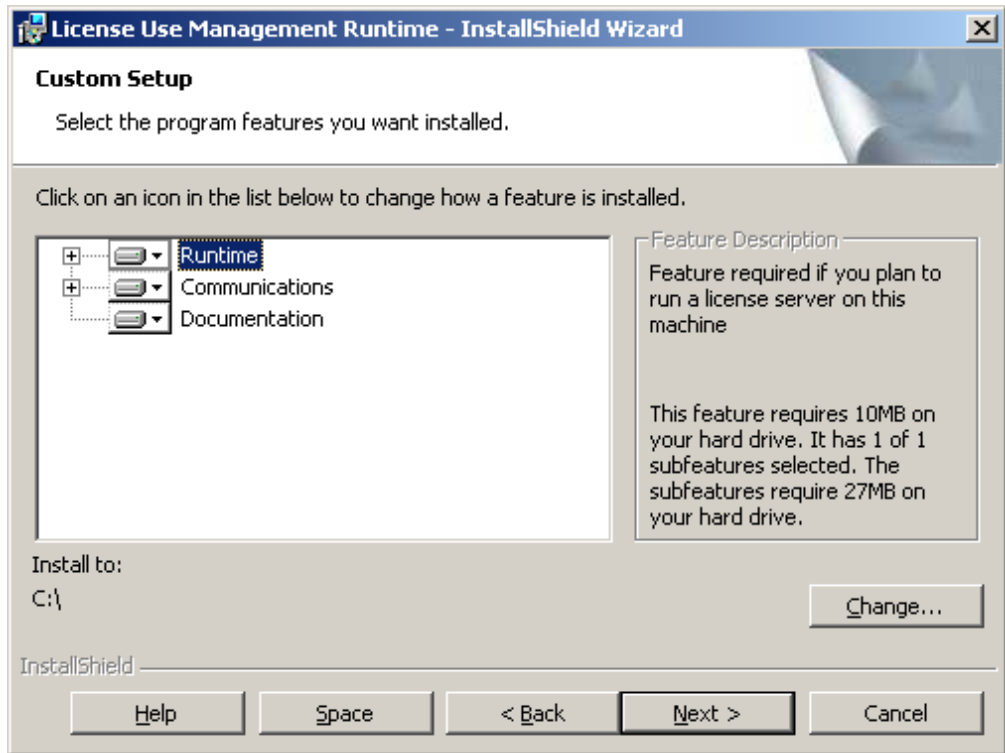
Figure 20. Custom Configuration Window

By default, all the features will be installed on the drive on which the operating system is installed. If you want to install all the features on that drive, go to step 15 on page 64.

If you want to install the features on a different drive, go to step 10.

If you want to install only one or two features, go to step 13 on page 63.

10. To change the drive on which to install the features, click **Change....**

    The Change Current® Destination Drive window is displayed (see Figure 21 on page 63).
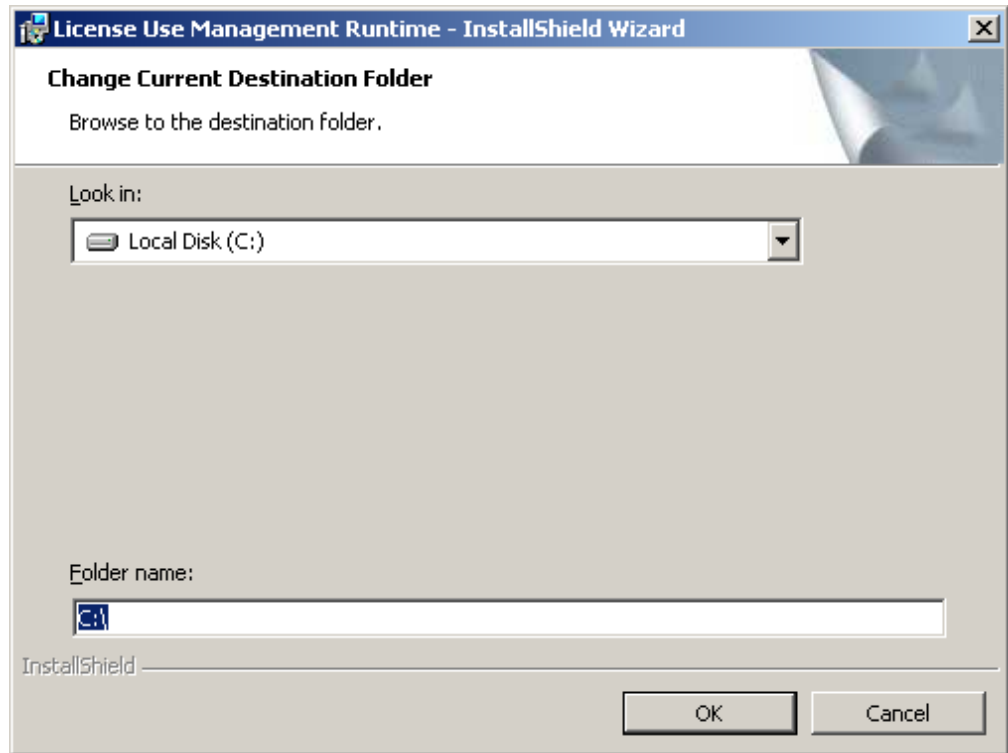
Figure 21. Change Current Destination Drive Window

11. Select the drive from the **Look in** drop-down list.

    Notes:

    a. Specify the drive only. The product is always installed in the `ifor` directory. Do not specify a folder name in the **Drive letter** field.

    b. If the Application Developer's Toolkit is installed on the workstation, License Use Management Runtime is always installed on the same drive.

12. Click **OK**.

    The Custom Setup window is displayed again, showing the changed drive information.

    If you want to install all the features, continue at step 15 on page 64

13. To not install a feature, click the icon:

    beside its name.

    The feature selection menu is displayed (see Figure 22).
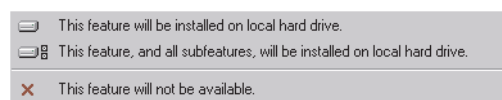
Figure 22. Feature Selection Menu

14. Select the choice **This feature will not be available.** The icon changes to:

to indicate that the feature will not be installed.

15. Click **Next >**. The Ready to Install the Program window is displayed (see Figure 23).
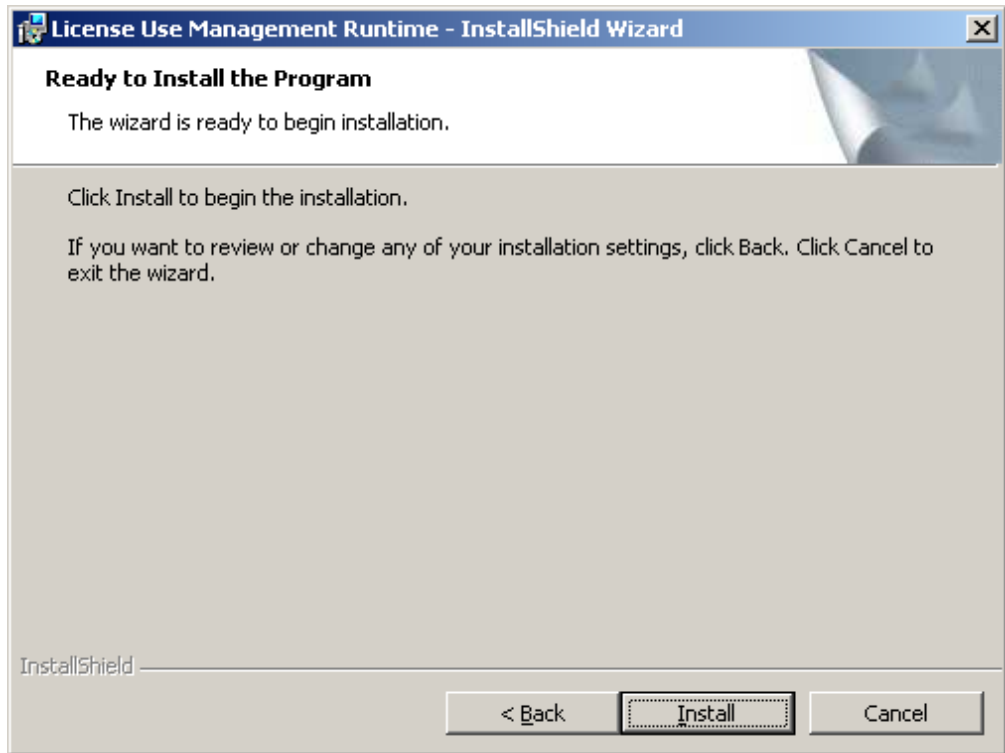
Figure 23. Ready to Install Program Window

16. Click **Next >**.

    The Installing License Use Management Runtime window is displayed (see Figure 24 on page 65), and License Use Management Runtime starts to copy files.
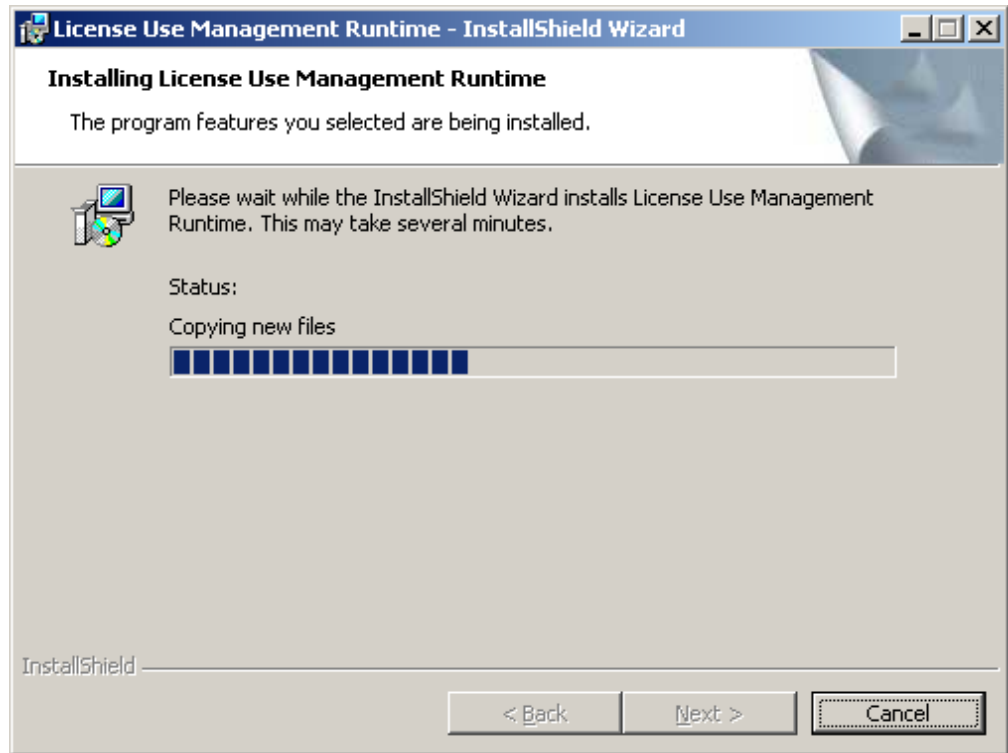
Figure 24. Installing License Use Management Runtime Window

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:

language

   Is the installation language; for example, EN_US for U.S. English

Windows has not updated the PATH environment variable, because the sum of the length of the current environment variable plus the length of the environment variable would exceed 1024 bytes. See "Verifying and Updating the PATH Environment Variable" on page 69 for information about how to manually add the License Use Management Runtime path information.

When installation is complete, the InstallShield Wizard Completed window is displayed.

17. Click **Finish**.

   If you installed License Use Management Runtime on a Windows XP, Windows 2000, Windows Server 2003 or Windows Vista workstation, you can start using it immediately.

18. If you are planning to use the machine in a Windows Terminal Server environment, run the **reglum.cmd** script (located in the `Windows/Application Compatibility Scripts/Install` directory) to register the DLLs.

License Use Management Runtime is installed on your workstation, and a folder called License Use Runtime is added to the Programs menu.

After installation you must configure License Use Management Runtime as explained in Chapter 4, "Configuring License Use Management Runtime," on page 71.

Installing after an Uninstallation

If you uninstall 4.*x.y*, your existing license databases are not deleted. They remain in the directory `c:\ifor\ls\conf`. If you then install License Use Management Runtime Version 4.6.8, the installation process renames the databases to have an extension of .old. If you want to use the databases with License Use Management Runtime Version 4.6.8, follow these steps:

1. Enter the following sequence of commands:

   ```
   i4cfg -stop
   cd c:\ifor\ls\conf
   dir
   ```

2. Look for the following files in the `ifor\ls\conf` directory list, and if present, perform the appropriate actions, in the order shown:

Step	If these files are present:	Enter the following commands:	Then perform the procedure:
1	crpdb.dat.old	`copy crpdb.dat.old crpdb.dat`	"crpdb file migration"
	crpdb.idx.old	`copy crpdb.idx.old crpdb.idx`	
2	licdb.dat.old	`copy licdb.dat.old licdb.dat`	"licdb file migration" on page 67
	licdb.idx.old	`copy licdb.idx.old licdb.idx`	
3	llmdb.dat.old	`copy llmdb.dat.old llmdb.dat`	"llmdb file migration" on page 67
	llmdb.idx.old	`copy llmdb.idx.old llmdb.idx`	
4	adddb.dat.old	`copy adddb.dat.old adddb.dat`	"adddb file migration" on page 67
	adddb.idx.old	`copy adddb.idx.old adddb.idx`	

crpdb file migration

To migrate the crpdb.* files, perform these steps:

1. Use an ASCII editor to create and save the file:

   ```
   c:\ifor\ls\conf\DBVersion
   ```

   that contains only the following text:

   ```
   451
   ```

2. Run the following migration option:

   ```
   i4mig -r -c -lmdLic
   ```

3. Reopen the file **DBVersion**.
   - If it contains the string **467**, the migration has been successful, skip to step 6.
   - If it contains the string **455**, go to step 4.
   - If it contains the string **460**, go to step 5.

4. Run the following migration option:

   ```
   i4mig -r -c -lmdLic455
   ```

5. Run the following migration option:

   ```
   i4mig -r -c -lmdLic460
   ```

6. Delete the **DBVersion** file.

licdb file migration

To migrate the licdb.* files, perform these steps:

1. Use an ASCII editor to create and save the file:

   `c:\ifor\ls\conf\DBVersion`

   that contains only the following text:

   `451`

2. Run the following migration option:

   a. `i4mig -r -lmdLic`

3. Reopen the file **DBVersion**.
   - If it contains the string **467**, the migration has been successful, skip to step 6.
   - If it contains the string **455**, go to step 4.
   - If it contains the string **460**, go to step 5.

4. Run the following migration option:

   `i4mig -r -lmdLic455`

5. Run the following migration option:

   `i4mig -r -lmdLic460`

6. Delete the **DBVersion** file.

llmdb file migration

To migrate the llmdb.* files, perform these steps:

1. Use an ASCII editor to create and save the file:

   `c:\ifor\ls\conf\DBVersion`

   that contains only the following text:

   `451`

2. Run the following migration option:

   `i4mig -r -n -lmdLic`

3. Reopen the file **DBVersion**.
   - If it contains the string **467**, the migration has been successful, skip to step 6.
   - If it contains the string **455**, go to step 4.
   - If it contains the string **460**, go to step 5.

4. Run the following migration option:

   `i4mig -r -n -lmdLic455`

5. Run the following migration option:

   `i4mig -r -n -lmdLic460`

6. Delete the **DBVersion** file.

adddb file migration

To migrate the adddb.* files, enter the following command:

`i4migadd`

Unattended License Use Management Runtime Installation

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

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.

1. To install License Use Management Runtime noninteractively, follow these steps: Make sure that the environment variable I4_INSTALL_DRIVE is set. To do this, look in the windows registry at the entry: **HKEY_LOCAL_MACHINE** > **SYSTEM** > **CurrentControlSet** > **Control** > **Session Manager** > **Environment**.

2. To set the I4_INSTALL_DRIVE variable, follow these steps:

   a. For Windows Server 2003 and XP, open **Start** > **Settings** > **Control Panel** > **System** > **Advanced** > **Environment Variables**

   b. For Windows Vista, open **Control Panel** > **System** > **Advanced system settings** > **Environment Variables**.

      The Environment Variables dialog is displayed.

   c. In the System Variables box, click **New**. In the New System Variable dialog, add the following information:

      - In the **Variable Name** field, type:

        I4_INSTALL_DRIVE

      - In the **Variable Value** field, type the installation drive letter; for example, **c:**.

   d. Click **OK**.

3. Check, in the file `ifor.bat`, whether the value of the ADDLOCAL parameter is suitable. If it is not, change it. This parameter identifies the features to be installed.

   If you are installing License Use Management Runtime over an older release of the product, all currently installed features are automatically upgraded.

4. In a new command prompt window, run `ifor.bat`.

   If you install or reinstall only some License Use Management Runtime features on a workstation on which more than one feature is already installed, the setup program silently installs or reinstalls those other preinstalled features.

5. When installation is complete, follow the instructions in Chapter 4, "Configuring License Use Management Runtime," on page 71.

Using the Installation Command

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

```
setup /s/v"ADDLOCAL=Runtime,GUI_Runtime,Communications,GUI_Communication,
Documentation LUM_SILENT=1 /qn /l log_file"
```

where:

s Specifies silent installation

v Indicates that what follows is a command line parameter

ADDLOCAL
   Specifies the features to be installed

LUM_SILENT=1
   Specifies that installation is to run silently

qn Specifies that the user interfaces not to be displayed during installation

l log_file
   Specifies the log file in which installation information is to be logged

When the installation completes successfully, the message `Installation operation completed successfully` is written to the log file.

Verifying and Updating the PATH Environment Variable

Check whether the PATH environment variable contains the following information:

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

where:

language

        Specifies the installation language; for example, EN_US for U.S. English

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

1. For Windows Server and XP, click **Start** > **Settings** > **Control Panel** > **System** > **Advanced** > **Environment Variables**.
2. For Windows Vista, click **Control Panel** > **System** > **Advanced System Settings** > **Environment Variables**.
3. If necessary, delete unused path information. The License Use Management Runtime path information requires 38 bytes. Remember to add any necessary delimiting semicolon to the end of the original PATH statement.
4. Highlight the system variable PATH. Click **Edit**. In the Edit System Variable dialog, add the following information to the end of the PATH variable:

   `;`*c*`:\ifor\WIN\BIN;` *c*`:\ifor\WIN\BIN\`*language*
5. Click **OK**.

The maximum length of the PATH environment variable is 1024 bytes. Therefore, if the sum of the length of the current path plus the length of the License Use Management Runtime PATH information would exceed 1024 bytes, Windows will not update the current PATH environment variable. You will then not be able to run License Use Management Runtime, until you manually update the PATH statement.

If License Use Management Runtime Fails to Start:

In certain unusual circumstances, the License Use Management Runtime environment variables are set during installation, but are not propagated to the system. The product therefore fails to start. If you experience this situation, select **Start** > **Settings** > **Control panel** > **System** > **Advanced** > **Environment Variables** > **OK** for Windows Server 2003 and XP or select **Control Panel** > **System** > **Advanced System Settings** > **Environment Variables** > **OK** for Windows Vista. License Use Management Runtime should then start correctly.

Uninstalling License Use Management Runtime

You can remove features of License Use Management Runtime, or the whole product, from your computer.

Remove License Use Management Runtime features from your workstation, as follows:

1. Click **Start** --> **Settings** --> **Control Panel** --> **Add/Remove Programs**. The Add/Remove Programs window is displayed.
2. Select **License Use Management Runtime**.
3. Click **Change**.

4. The License Use Management Runtime installer's Welcome window is displayed.

5. Click **Next**. The Program Maintenance window is displayed.

6. Click **Modify**.

7. Click **Next**. The Custom Setup window is displayed.

8. Select the features you want to delete. The feature selection menu is displayed.

9. Click **This feature will not be available** for each feature to be deleted.

10. Click **Next**. The Ready to Modify the Program window is displayed.

11. Click **Install**. When uninstallation is complete, the InstallShield Wizard Completed window is displayed.

12. Click **Finish**.

The selected features have been removed from your computer.

Remove the whole License Use Management Runtime product from your workstation, as follows:

1. Click **Start** --> **Settings** --> **Control Panel** --> **Add/Remove Programs**. The Add/Remove Programs window is displayed.

2. Select **License Use Management Runtime**.

3. Click **Remove**. A window is displayed asking you if you are sure that you want to remove License Use Management Runtime. Click **Yes** to confirm. The Windows InstallShield Wizard deletes License Use Management Runtime from your computer.

Note: 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.

Chapter 4. Configuring License Use Management Runtime

ALL

The scenarios in this chapter describe how to set up your License Use Management Runtime environment.

License Use Management Runtime is not required for non-runtime nodelocked workstations (see "License Use Management at a Glance" on page 1).

Setting Up Your Servers and Clients

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

The configuration tool has a GUI, a script interface on UNIX and AIX, and a command line interface. The information you provide is saved in a configuration file. When you start License Use Management 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 Management Runtime Configuration File," on page 239 for reference information on the configuration file.

Configuring to Handle Nodelocked Licenses

To handle products that have nodelocked licenses only, you need to configure the workstation as a nodelocked license server ("Scenario 1: Configuring a Standalone Nodelocked License Server" on page 77) and have the nodelocked license server up and running. Every workstation is automatically configured as a nodelocked license server when License Use Management Runtime is installed. You do not have to do any configuration unless you want to change the default configuration.

If a workstation configured as a nodelocked license server is in a network, instances of the basic license tool on other workstations 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 80), you can run the basic license tool on the nodelocked license server workstation 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 84) and configure each client as a network license client ("Scenario 5: Configuring a Network License Client" on page 90). If you use products with customer-managed or reservable licenses, you must also configure one server as the central registry server ("Scenario 4: Configuring the Central Registry License Server" on page 87).

Determining the Required Configuration

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

If you know the types of licenses that are used by the license-enabled products in your environment, Table 13 will help you to determine how to configure:

- The license servers for the application
- The workstations that will request licenses for the application

See the application documentation for information that might affect your configuration. The *enrollment certificate file* that you receive from the vendor shows the type of license.

Table 13. Configuration Required to Support All Types of Licenses

License Type	License Requester	License Server
Simple Nodelocked (Non-Runtime-Based Enabling) (note 1)	License-Enabled Application	None
Simple Nodelocked (Runtime-Based Enabling) (note 1)	License-Enabled Application	Nodelocked License Server on the Same workstation as the Application
Use-Once Nodelocked	License-Enabled Application	Nodelocked License Server on the Same workstation as the Application
Concurrent Nodelocked	License-Enabled Application	Nodelocked License Server on the Same workstation as the Application
Per-Server	License-Enabled Application	Nodelocked License Server on the Same workstation as the Application
Use-Once	Network License Client	Network License Server (note 2), and Central Registry License Server (note 3)
Concurrent	Network License Client	Network License Server (note 2) and Central Registry License Server (note 3)
Concurrent-Offline	Network License Client	Network License Server (note 5) and Central Registry License Server (note 3)
Per-Seat	Network License Client and Nodelocked License Server	Central Registry License Server (note 3)
Reservable	Network License Client and Nodelocked License Server (note 4)	Network License Server (note 2) and Central Registry License Server (note 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 workstation (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 licenses, 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, unless the license is requested in named user mode.
5. No server is required once the reservation has been made.

Before You Configure

Before you begin the configuration process, for every workstation you are going to configure, you need to decide which roles it will play and how you plan to set up direct binding or namespace binding. You might also decide to override some of the configuration defaults. The actions you can take during configuration are summarized in Table 14. Check the table for all the roles your workstation will play, and make all the indicated decisions before you start configuration.

Table 14. 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" on page 74)	Yes		Yes	Yes
Customize selection of information logged or accept default? ("Customizing Log Information" on page 74)	Yes		Yes	Yes
Start license servers at system startup (default is no)? ("Automatically Starting License Servers" on page 75)	Yes		Yes	Yes
Disable remote administration of network license servers (default is no)? ("Disabling Remote Administration" on page 75)			Yes	
Disable remote administration of nodelocked license servers (default is yes)? ("Disabling Remote Administration" on page 75)	Yes			
Set up direct binding ("Configuring Direct Binding" on page 75)				
Prepare a list of nodelocked license servers for remote administration	Yes (note 1)	Yes	Yes	Yes
Have your direct binding servers list ready	Yes (note 1)	Yes	Yes	Yes
Know which workstation is the central registry	Yes (note 1)	Yes	Yes	Yes
Change default direct binding ports?	Yes (note 1)	Yes	Yes	Yes
Set up namespace binding (UNIX and AIX only) ("Configuring Namespace Binding" on page 75)				
Join an existing alternate cell?	Yes (note 1, 2)	Yes	Yes	Yes
Know the name of a server already there	Yes (note 1, 2)	Yes	Yes	Yes
Run a replica GLB?	Yes (note 1, 2, 3)		Yes	Yes
Join the default cell?	Yes (note 1, 2)	Yes	Yes	Yes
Run a replica GLB?	Yes (note 1, 2)		Yes	Yes
Start a new cell?	Yes (note 1, 2)		Yes	Yes

Table 14. Configuration Options (continued)

Configuration Options	Nodelocked License Server	Network License Client	Network License Server	Central Registry License Server

Notes

1. Applicable if you plan to run the basic license tool and administer licenses on other workstations from this nodelocked license server.

2. Applicable if you plan to run the basic license tool and administer licenses on other workstations from this nodelocked license server, or if you want instances of the basic license tool on other workstations to be able to administer licenses on this workstation.

3. The GLB replica is not available on Linux operating systems.

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 and load messages
Logs the successful start or stop of the license server, and when the threshold on the workload is exceeded or regained. 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 workstation. Otherwise, you must remember to start the services manually after you configure and before you use 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 can be administered by running the basic license tool on that machine only.

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. If you configure in this way, clients will be able to locate the server only through direct binding.

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 workstation.

You will enter the hostnames or network addresses of all the servers in the list, except for the nodelocked license server on the local workstation, 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.

Configuring Namespace Binding

AIX

When you configure a nodelocked license server, network license server, network license client, or central registry license server to be part of a namespace binding environment, clients will be able to locate the server. The basic license tool will be able to locate remote servers, through either namespace binding or direct binding.

You must know which cell this workstation is to be part of (see "Planning Cells" on page 34 and "Selecting the Global Location Brokers" on page 35). If the workstation is to join an existing cell, other than the default cell, you must be able to identify a server that is already in the cell.

If there are other users of NCS at your location who might create a default cell, it is safer to configure only alternate cells. Since the default cell always has the same Universal Unique Identifier (UUID), results would be unpredictable.

In the case of a server joining an existing cell, you must decide whether the server is to run a replica of the global location broker.

If your workstation is on a subnetwork different from that of the server that starts the global location broker, or if your system does not support broadcasting, further

configuration steps are needed after you do the basic configuration (see
"Configuring to Reach a Global Location Broker in a Different Subnetwork" on
page 93).

Using the Configuration Tools

To configure License Use Management Runtime, you can use one of the following
tools:
- Configuration Tool script (UNIX and AIX only)
- Configuration Tool graphical user interface
- Configuration Tool command line interface

Using the Configuration Tool Script

License Use Management Runtime provides a configuration script you can use if
you want to perform a guided configuration of your workstation without using the
graphical user interface.

Login with root authority and issue the following command:

```
i4cfg -script
```

You are then offered the choice of four configuration scenarios:
- If the workstation you are configuring is the central registry license server, select
  4. During the question-and-answer session that follows, the workstation will
  automatically be configured as a network license client, and you can optionally
  configure as a nodelocked license server or a network license server, or both.
- If the workstation you are configuring is not the central registry license server,
  but is a network license server, select **3**. During the question-and-answer session
  that follows, the workstation will automatically be configured as a network
  license client, and you can optionally configure as a nodelocked license server.
- If the workstation you are configuring is a nodelocked license server and is
  neither the central registry license server nor a network license server, select **2**.
  During the question-and-answer session that follows, you can optionally
  configure as a network license client.
- If the workstation is a network license client only, select **1**.

If you run the script more than once, the latest information entered takes effect.

Using the Configuration Tool GUI
If the License Use Management Runtime graphical user interface is installed on
your workstation, you can use it to configure your workstation.

Login with root authority and issue the command:

```
i4cfg
```

Login with administrator authority and start the configuration tool by one of the
following means:
- Click **Start** --> **Programs** --> **License Use Runtime** --> **Configuration Tool**.
- In the **License Use Runtime** folder, click the **Configuration Tool** icon.
- Enter the i4cfg command with no parameters.

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

Each of the configuration scenarios in this chapter shows how to configure a workstation to play only one role in the licensing environment. If the workstation 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.

For details of the i4cfg command, see "i4cfg - Configuration Tool" on page 167.

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 Management Runtime servers. Use "Scenario 2: Configuring a Nodelocked License Server in a Network" on page 80 if you want to:

- Run the basic license tool and administer licenses on other license servers from this workstation
- Administer nodelocked licenses on this workstation from instances of the basic license tool on other workstations

A nodelocked license server can use licenses of the types shown in Table 13 on page 72. By default, every workstation 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 workstation as a standalone nodelocked license server after installation, perform the following steps:

1. Start the **Configuration Tool.** See "Using the Configuration Tool GUI" on page 76.
2. On the **Configure As** page, select **Nodelocked License Server**. The notebook is shown in Figure 25 on page 78.
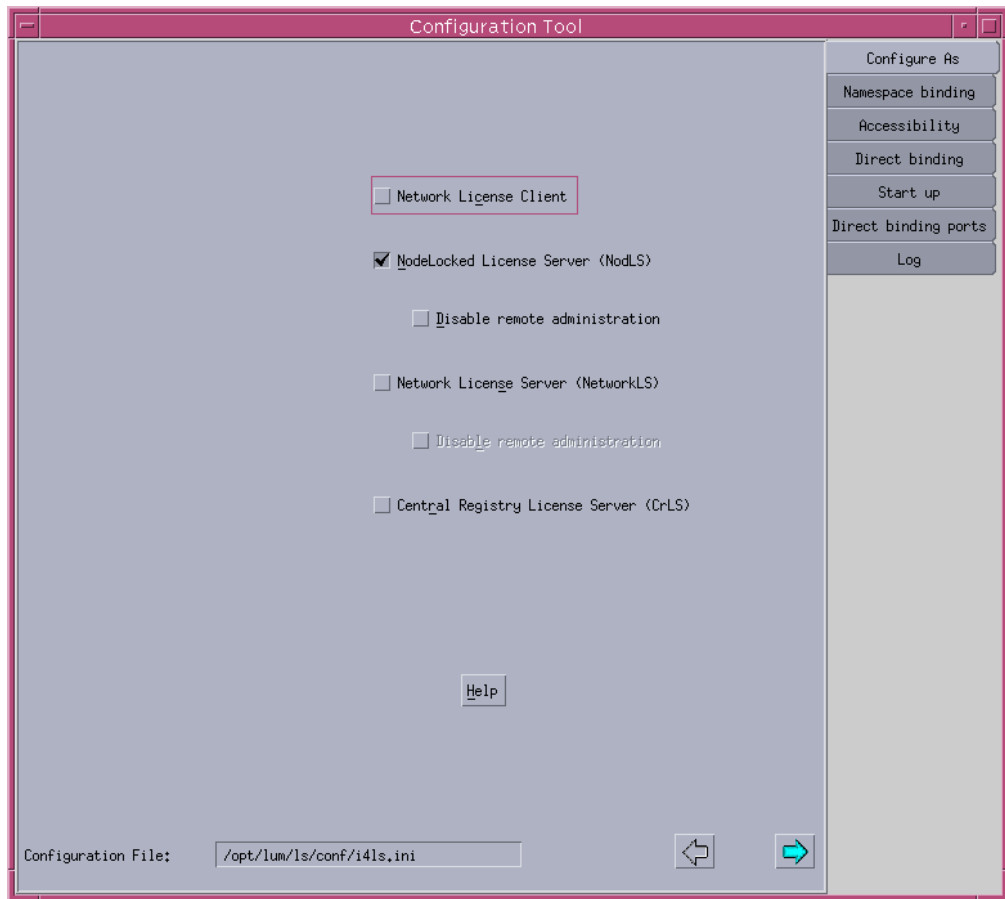
Figure 25. Configuration Tool Notebook - Standalone Nodelocked License Server

For the type of configuration you are doing, ignore the **Direct binding** and **Direct binding ports** pages 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 workstation.

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 26 on page 79.
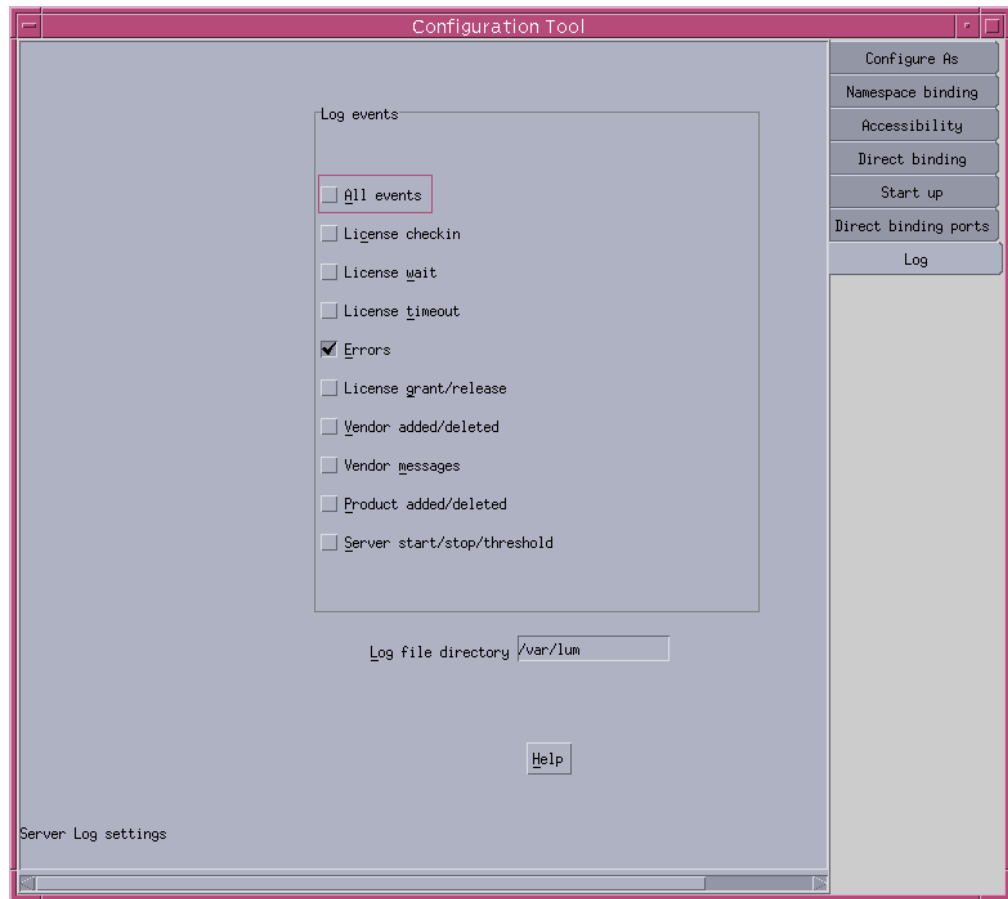
Figure 26. Configuration Tool Notebook - Log Page

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 llmlg*nn_*, where *nn* assumes values from 00 to 99. When a file is full, according to a maximum number of lines that are fixed in the configuration file, a new log file is started.

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.

6. To run the configured server, see "Starting and Listing Your Servers" on page 94.

Configuration Script Equivalent

To configure the standalone nodelocked license server using the configuration script, enter the command:

```
i4cfg -script
```

In response to the first question, select **2**; then respond to the questions as they are asked.

Command-Line Equivalent

To configure the standalone nodelocked license server:

```
i4cfg -a n -S a -e a
```

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 or a namespace binding cell. Configuring in this way, you can:

- Run the basic license tool and administer licenses on other license servers from this workstation
- Allow instances of the basic license tool on other workstations to administer nodelocked licenses on this workstation

A nodelocked license server can use licenses of the types shown in Table 13 on page 72.

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

1. Start the **Configuration Tool.** See "Using the Configuration Tool GUI" on page 76.
2. On the **Configure As** page, select **Nodelocked License Server**. On Windows, this adds the **User** tab to the notebook.
3. Leave the **Disable remote administration** box unchecked, to enable instances of the basic license tool on other workstations to administer nodelocked licenses on this workstation.
4. If you want to change the defaults, complete the **Start up** and **Log** pages, as explained in "Scenario 1: Configuring a Standalone Nodelocked License Server" on page 77.

Windows

5. On Windows, select the **User** tab. The User section is displayed, as shown in Figure 27 on page 81.
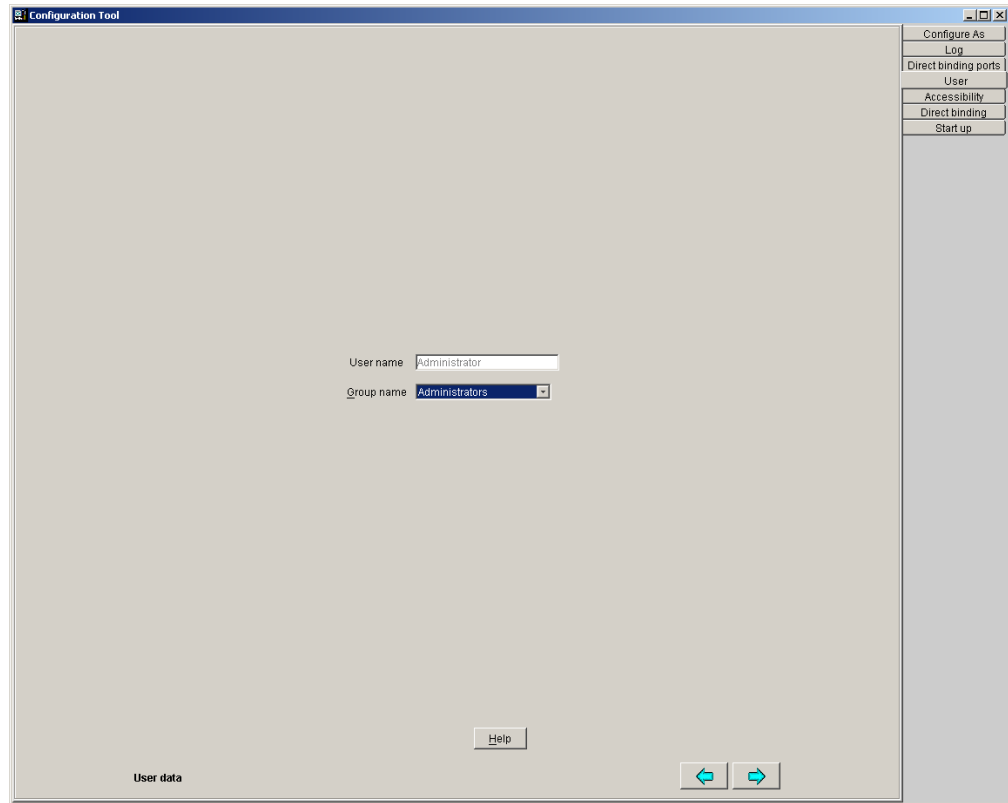
Figure 27. Configuration Tool Notebook - User Section

> The user name is the logon user ID in effect when you do the configuration and it cannot be changed. Select a group name of the user if you want to change the default. The group name can be chosen from the list of Windows group names.
>
> License Use Management Runtime treats the user name and group name as the name of the user and group who requests and uses licenses on this workstation. The user name and group name are displayed as part of various reports about license usage. They are also used to reserve reservable or concurrent-offline licenses for a specific user or for a specific group.

`ALL`

6. Use the remaining pages of the notebook to establish remote connections between this server and other License Use Management Runtime license servers. From this workstation, you will be able to use the basic license tool to administer licenses on those servers.

7. If you have decided that this server is to use direct binding to connect to other servers, select the **Direct binding** tab. In this case, or if you are working in a Windows environment, skip the next step (**Namespace binding** page).

   The Direct binding section is displayed, as shown in Figure 28 on page 82.
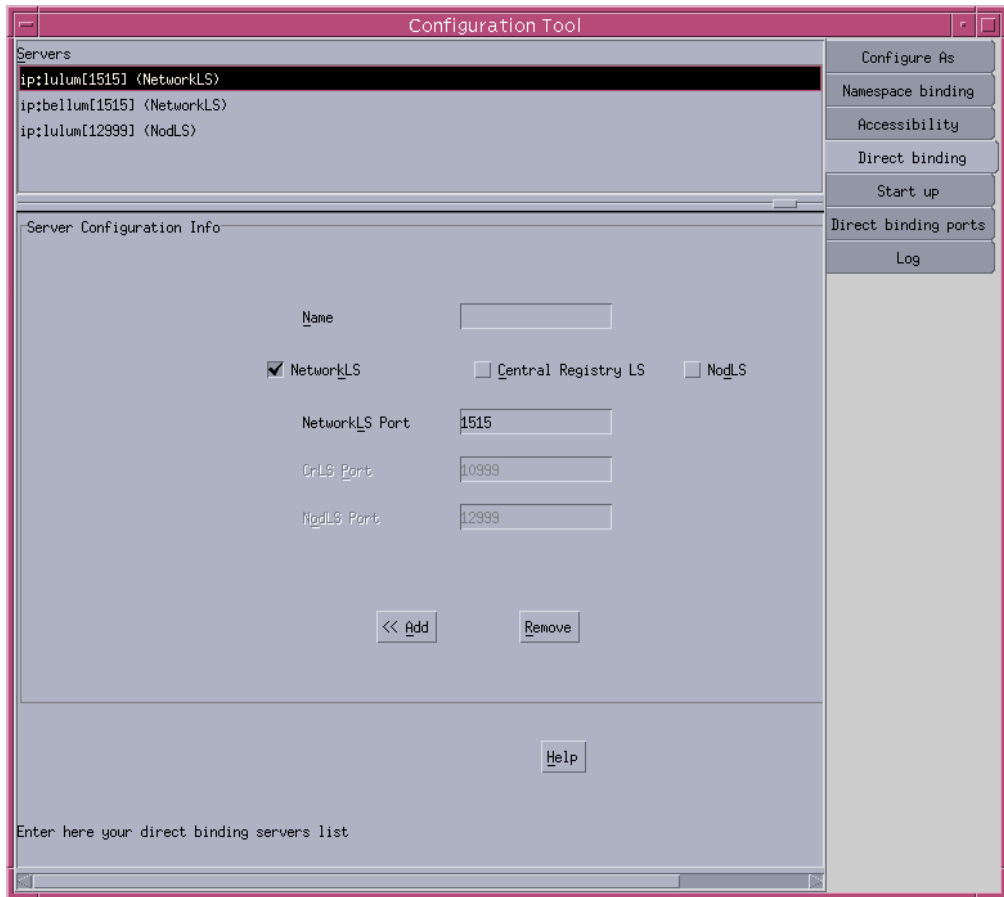
Figure 28. Configuration Tool Notebook - Direct Binding Section

On this page you specify all the network license servers and nodelocked license servers whose licenses you will administer remotely from this workstation. 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 workstation, include all the servers whose licenses you want to administer. In this example, the administrator performs the following steps for each server:

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

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

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

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

8. 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.

UNIX

9. On UNIX or AIX, if you have decided that this server is to communicate with other servers through namespace binding rather than direct binding, select the **Namespace binding** tab. The Namespace binding page is displayed, as shown in Figure 29 on page 83. Make sure that the **Namespace binding support**

check box is selected.
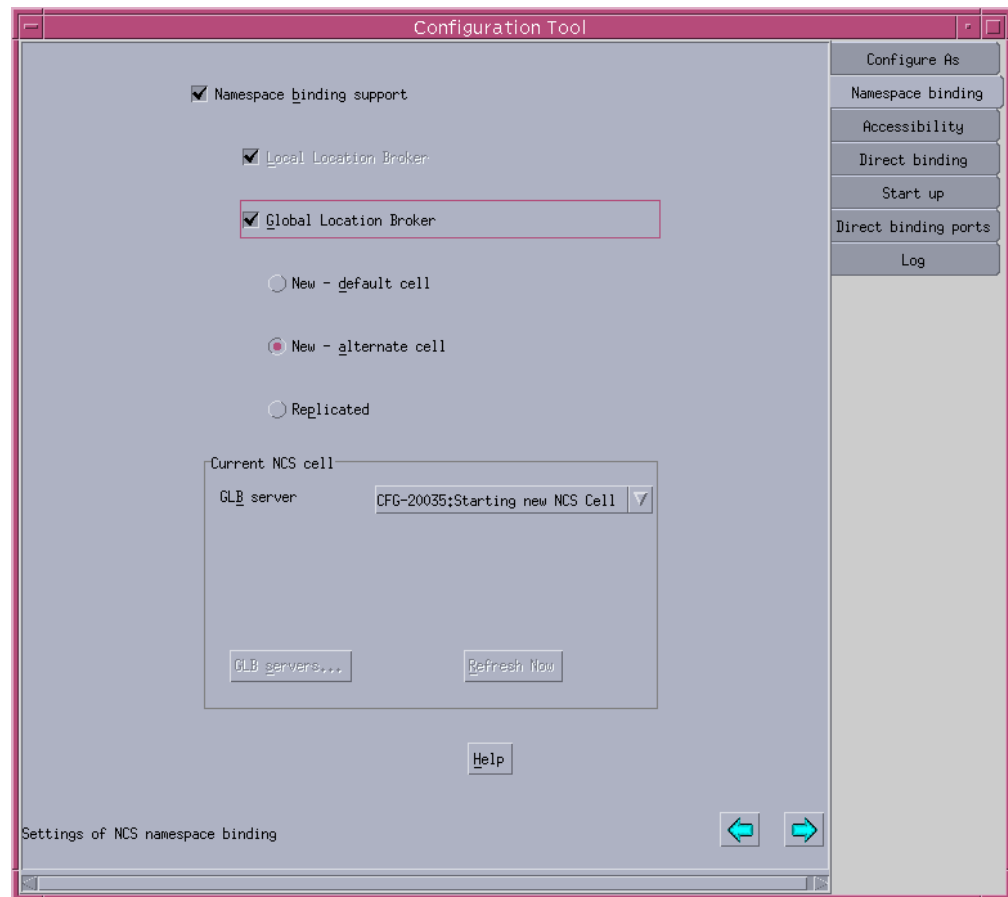
Figure 29. Configuration Tool Notebook - Namespace Binding Section

If this is the first server (including nodelocked license servers, network license servers and the central registry license server) to be configured in the cell, then select the **Global Location Broker** check box to start the global location broker on the server. Select either **New - default cell** or **New - alternate cell** to start the global location broker in a new cell.

If there are other users of NCS at your location who might create a default cell, it is safer to configure only alternate cells. Because there can be only one default cell, results would be unpredictable.

If, alternatively, other servers have already been configured in the cell, follow these steps:

a. If you want the server being configured to have a copy of the global location broker, select the **Global Location Broker** check box and the **Replicated** radio button. If you do not want to run a copy of the global location broker, do not check **Global Location Broker**.

   Note: The **Replicated** radio button is not displayed on the Linux GUI. The glbd replica is not available on Linux operating systems.

b. In the **GLB Server** field, choose the address of a server in the cell that has the global location broker.

c. Check that there is no glb_site.txt file, or, if the file exists, that it includes a server that is in the cell being joined. Otherwise, use the i4cfg -G null command to delete the existing site list.

Chapter 4. Configuring License Use Management Runtime **83**

d. If the selected GLB server is on a workstation that has multiple network adapters, make sure the GLB server has been started on the adapter to which the workstation being configured is connected.

If your workstation is on a subnetwork different from the one of the server that starts the global location broker, or if your system does not support broadcasting, further configuration steps are needed (see "Configuring to Reach a Global Location Broker in a Different Subnetwork" on page 93).

10. 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.

11. To run the configured server, see "Starting and Listing Your Servers" on page 94.

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 Management Runtime servers.

Configuration Script Equivalent

To configure the nodelocked license server in a network using the configuration script, enter the command:

```
i4cfg -script
```

In response to the first question, select 2; then respond to the questions as they are asked.

Command-Line Equivalent

To configure the nodelocked license server in a network:

- With direct binding:

```
i4cfg -a n -S a,n -e a -b "'network ip:lab15099 ip:rocky ip:coyote ip:sprint'
 'nodelocked ip:speedy' 'registry ip:lab15099'" -n n
```

- With namespace binding that is joining an existing cell that has UUID 456b91c50000.0d.00.00.87.84.00.00.00:

```
i4cfg -a n -S a,n -e a -l /var/ifor -b null -n l
-c 456b91c50000.0d.00.00.87.84.00.00.00
```

Note that to achieve the same result as the direct binding example all the other servers must join the same cell.

Scenario 3: Configuring a Network License Server

This scenario shows how the administrator configures License Use Management Runtime as a network license server, making it part of a direct binding server list or a namespace binding cell.

When you configure a workstation as a network license server, you can use licenses of the types shown in Table 13 on page 72. 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 Management Runtime:

1. Start the **Configuration Tool.** See "Using the Configuration Tool GUI" on page 76.

2. On the **Configure As** page, select **Network License Server**. Note that **Network License Client** is then automatically checked. Leave **Disable Remote Administration** unchecked. The **Configure As** page is shown in Figure 30.
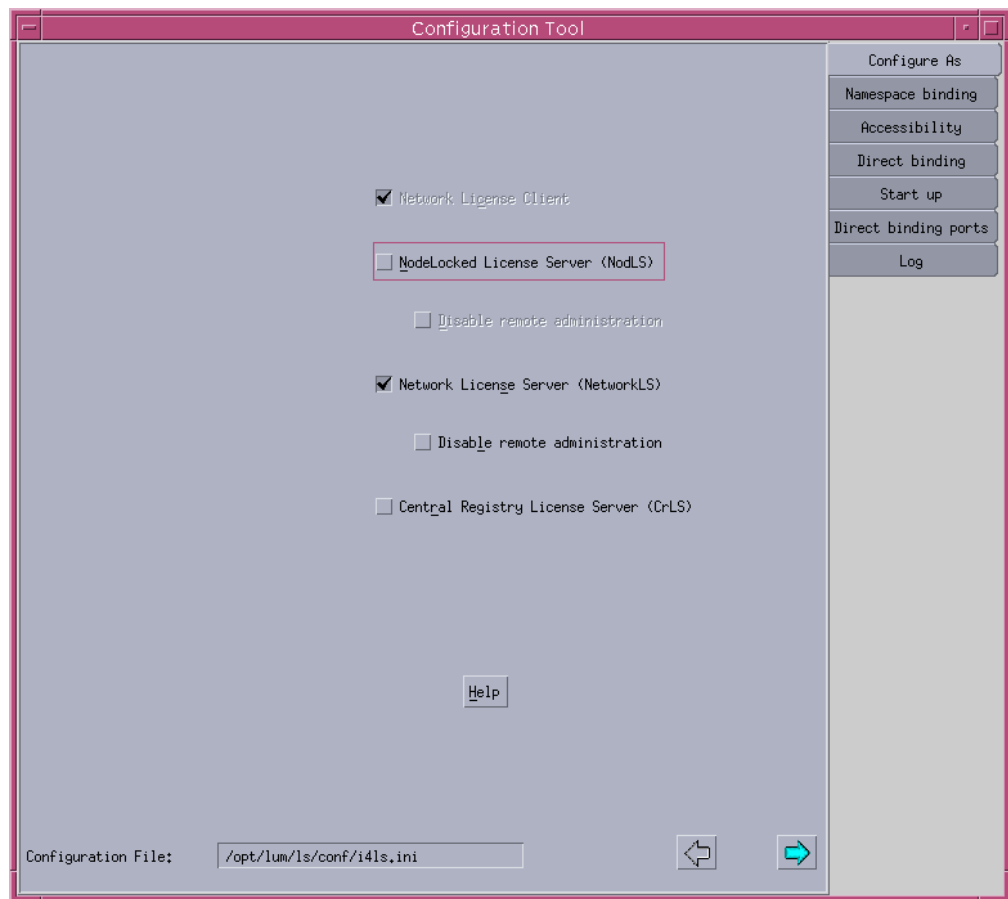
Figure 30. Configuration Tool Notebook - Network License Server

3. On the **Start up** page, select **Start services at system startup** to start the network license server when you power on the workstation.

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 26 on page 79. The logged events are stored in the files `logdb`*nn*`_`, where *nn* assumes values from 00 to 99. When a file is full, according to a maximum number of lines that are fixed in the configuration file, a new one is started.

5. If you have decided that this server is to be part of a direct binding servers list, select the **Direct binding** tab. If you configure in this way, clients will be able to locate the server only through direct binding. In this case, skip the next step (**Namespace binding** page). The **Direct binding** section is displayed, as shown in Figure 28 on page 82.

   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 workstation, 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 workstation.

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

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

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

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

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

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

6. On Windows, 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.

   License Use Management Runtime treats the user name as the name of the user who requests and uses licenses on this workstation. 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 a specific group.

AIX

7. On UNIX or AIX, if you have decided that this server is to be part of a namespace binding cell rather than a direct binding servers list, select the **Namespace binding** tab. The **Namespace binding** page is displayed, as shown in Figure 29 on page 83. Make sure that the **Namespace binding support** check box is selected If you configure in this way, clients will be able to locate the server through either namespace binding or direct binding.

   If this is the first server (including nodelocked license servers, and the central registry license server) to be configured in the cell, then select the **Global Location Broker** check box to start the global location broker on the server. Select either **New - default cell** or **New - alternate cell** to start the global location broker in a new cell.

   If there are other users of NCS at your location who might create a default cell, it is safer to configure only alternate cells. Since the two default cells would have the same UUID, results would be unpredictable.

   If, alternatively, other network license servers or the central registry license server, or both, have already been configured in the cell, follow these steps:

   a. If you want the server being configured to have a copy of the global location broker, select the **Global Location Broker** check box and the **Replicated** radio button. If you do not want to run a copy of the global location broker, do not check **Global Location Broker**.

      Note: The **Replicated** radio button is not displayed on the Linux GUI. The glbd replica is not available on Linux operating systems.

   b. In the **GLB Server** field, choose the address of a server in the cell that has the global location broker.

   c. Check that there is no `glb_site.txt` file, or, if the file exists, that it includes a server that is in the cell being joined. Otherwise, use the `i4cfg -G null` command to delete the existing site list.

d. If the selected GLB server is on a workstation that has multiple network adapters, be sure the GLB server has been started on the adapter to which the workstation being configured is connected.

If your workstation is on a subnetwork different from the one of the server that starts the global location broker, or if your system does not support broadcasting, further configuration steps are needed (see "Configuring to Reach a Global Location Broker in a Different Subnetwork" on page 93).

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 servers, see "Starting and Listing Your Servers" on page 94.

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

Configuration Script Equivalent

To configure the network license server using the configuration script, enter the command:

```
i4cfg -script
```

In response to the first question, select 3; then respond to the questions as they are asked.

Command-Line Equivalent

To configure the network license server:

- With direct binding:

```
i4cfg -a s -S a -e cegvmp -b "'network ip:lab15099 ip:rocky ip:coyote ip:sprint'
'nodelocked ip:speedy' 'registry ip:lab15099'" -n n
```

- With namespace binding that is starting a new alternate cell:

```
i4cfg -a s -S a -e cegvmp -l /var/ifor -b null -n g -r first
```

To achieve the same result as the direct binding example, all servers must join this new cell.

Scenario 4: Configuring the Central Registry License Server

This scenario shows how the administrator configures the central registry license server, making it part of a direct binding servers list or a namespace binding cell.

Be sure you configure only one central registry license server. Carefully plan where to configure it; after you start it, you cannot move it. (See "Planning the Central Registry" on page 37.)

Configuring a workstation as the central registry license server makes it possible to use licenses of the types shown in Table 13 on page 72. 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 Management Runtime:

1. Start the Configuration Tool. See "Using the Configuration Tool GUI" on page 76.

2. On the Configure As page, select **Central Registry License Server**. **Network License Client** is then automatically checked. The **Configure As** page is shown in Figure 31.
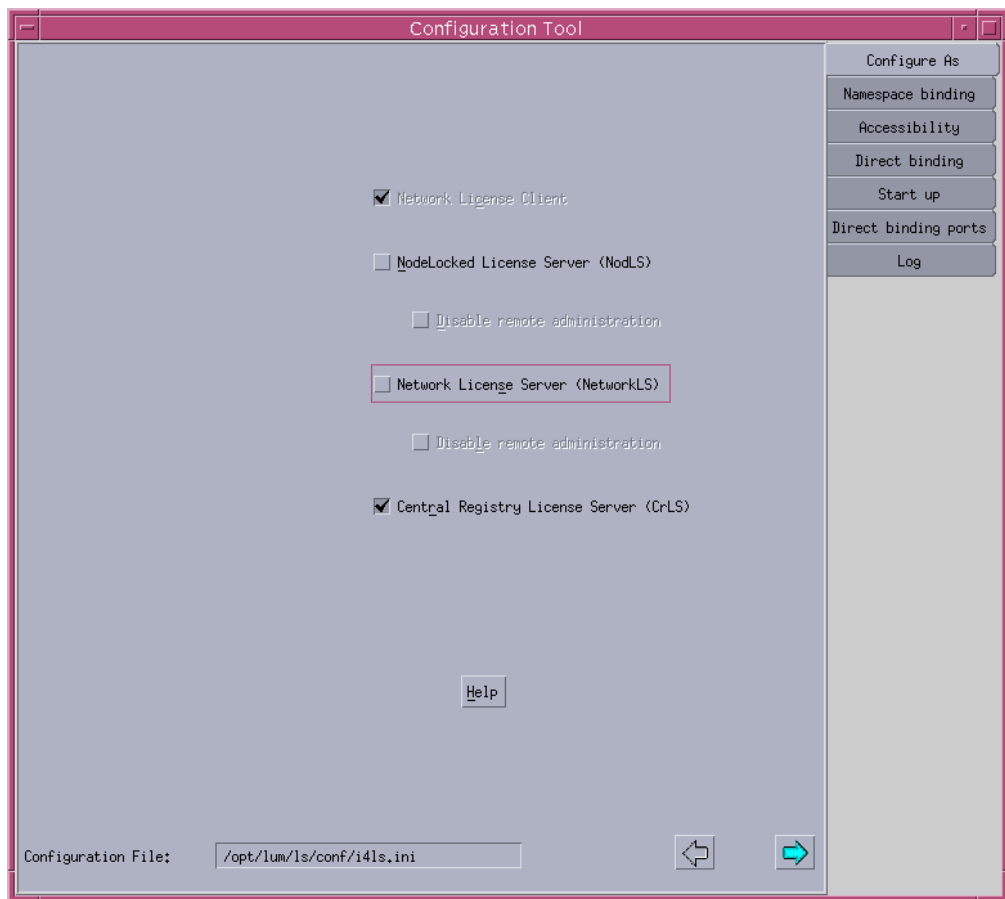
Figure 31. Configuration Tool Notebook - Central Registry License Server

3. On the **Start up** page, select **Start services at system startup** to start the central registry license server when you power on the workstation.

4. If you have decided that the central registry license server is to be part of a direct binding servers list, select the **Direct binding** tab. If you configure in this way, clients will be able to locate the server only through direct binding. In this case, skip the next step (**Namespace binding** page). The **Direct binding** section is displayed, as shown in Figure 28 on page 82.

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 either the a network license server or the nodelocked license server run on the same workstation, 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 workstation.

In this example, the administrator performs the following steps for each server with which this server will communicate: On Windows, make sure that the following are set:

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

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

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

5. On Windows, 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.

License Use Management Runtime treats the user name as the name of the user who requests and uses licenses on this workstation. 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 a specific group.

6. On UNIX or AIX, if you have decided that the central registry license server is to be part of a namespace binding NCS cell rather than a direct binding servers list, select the **Namespace binding** tab. The **Namespace binding** page is displayed, as shown in Figure 29 on page 83. Make sure that the **Namespace binding support** check box is selected. If you configure in this way, clients will be able to locate the server through either namespace binding or direct binding.

If this is the first server (including nodelocked license servers, network license servers, and the central registry license server) to be configured in the cell, then select the **Global Location Broker** check box to start the global location broker on the server. Select either **New - default cell** or **New - alternate cell** to start the global location broker in a new cell.

If there are other users of NCS at your location who might create a default cell, it is safer to configure only alternate cells. Since the two default cells would have the same UUID, results would be unpredictable.

If, alternatively, network license servers have already been configured in the cell, follow these steps:

    a. If you want the server being configured to have a copy of the global location broker, select the **Global Location Broker** check box and the **Replicated** radio button. If you do not want to run a copy of the global location broker, do not check **Global Location Broker**.

       Note: The **Replicated** radio button is not displayed on the Linux GUI. The glbd replica is not available on Linux operating systems.

    b. In the **GLB Server** field, choose the address of a server in the cell that has the global location broker.

    c. Check that there is no `glb_site.txt` file, or, if the file exists, that it includes a server that is in the cell being joined. Otherwise, use the `i4cfg -G null` command to delete the existing site list.

    d. If the selected GLB server is on a workstation that has multiple network adapters, be sure the GLB server has been started on the adapter to which the workstation being configured is connected.

If your workstation is on a subnetwork different from the one of the server that starts the global location broker, or if your system does not support broadcasting, further configuration steps are needed (see "Configuring to Reach a Global Location Broker in a Different Subnetwork" on page 93).

7. 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.

8. To run the configured server, see "Starting and Listing Your Servers" on page 94.

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

Configuration Script Equivalent

To configure the central registry license server using the configuration script, enter the command:

```
i4cfg -script
```

In response to the first question, select **4**; then respond to the questions as they are asked.

Command-Line Equivalent

To configure the central registry license server:

- With direct binding:

```
i4cfg -a r -S a -e cegvmp -b "'network ip:lab15099 ip:rocky ip:coyote ip:sprint'
 'nodelocked ip:speedy' 'registry ip:lab15099'" -n n
```

- With namespace binding, joining an existing alternate cell and replicating the global location broker at the server lab68086:

```
i4cfg -a r -S a -e cegvmp -l /var/lum -b null -n g
 -r from:ip:lab68086 -c 789b91c50000.0d.00.00.87.84.00.00.00
```

Note that to achieve the same result as the direct binding example, all the other servers must join the same cell.

Scenario 5: Configuring a Network License Client

This scenario shows how the administrator or the end user configures License Use Management Runtime as a client of the network license servers configured in "Scenario 3: Configuring a Network License Server" on page 84, and of the central registry license server configured in "Scenario 4: Configuring the Central Registry License Server" on page 87. Configuring a workstation as a network license client makes it possible to use licenses of the types shown in Table 13 on page 72.

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

1. Start the configuration tool. See "Using the Configuration Tool GUI" on page 76. The Configuration Tool notebook is displayed, as shown in Figure 32 on page 91.
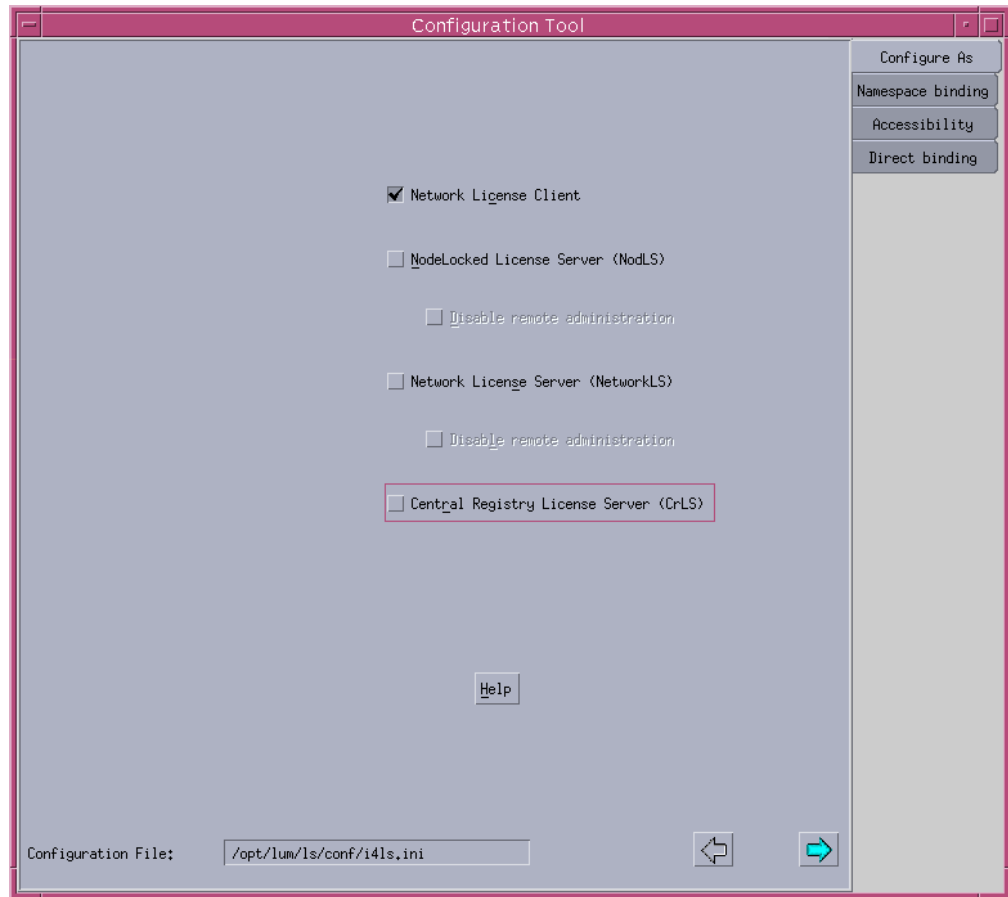
Figure 32. Configuration Tool Notebook - Network License Client

2. If the network license client is to locate network license servers using direct binding, select the **Direct binding** tab. In this case, skip the next step (**Namespace binding** page).

   The **Direct binding** section is displayed, as shown in Figure 28 on page 82.

   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 34.

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

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

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

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

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

3. On Windows, select the **User** tab. The User section is displayed, as shown in Figure 27 on page 81.

   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.

   License Use Management Runtime treats the user name as the name of the user who requests and uses licenses on this workstation. 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 a specific group.

4. On UNIX or AIX if the network license client is to locate the server using namespace binding rather than direct binding, select the **Namespace binding** tab. The **Namespace binding** page is displayed, as shown in Figure 33. Make sure that the **Namespace binding support** check box is selected.
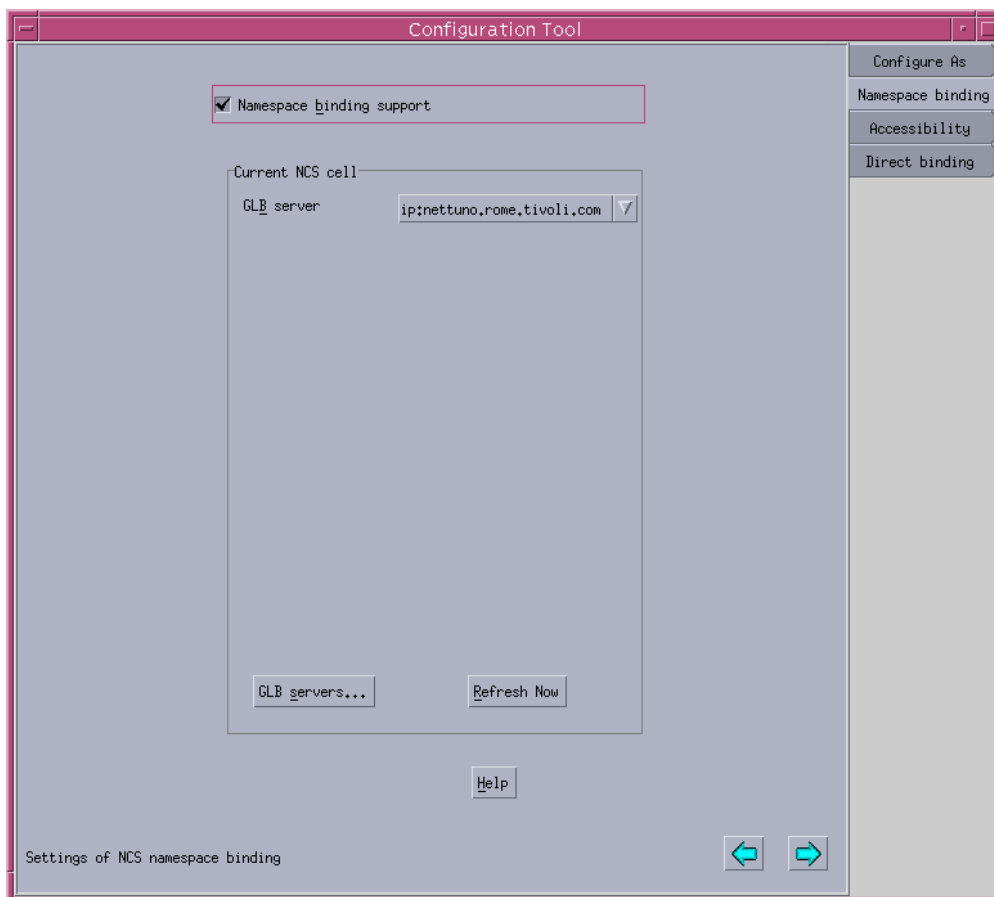
Figure 33. Configuration Tool Notebook - Namespace Binding Section

In the **Namespace binding** section, select the down arrow on the right of the **GLB server** field to see the list of servers where the global location broker runs in each existing cell, and select a server from the list. The client workstation joins the same cell as the selected server. If the selected server is on a workstation that has multiple network adapters, be sure the server has been started on the adapter to which your client is connected.

If your workstation is on a subnetwork different from the one of the server that starts the global location broker, or if your system does not support broadcasting, further configuration steps are needed (see "Configuring to Reach a Global Location Broker in a Different Subnetwork" on page 93).

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.

6. To run the configured server, see "Starting and Listing Your Servers" on page 94.

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

Configuration Script Equivalent

To configure the network license client using the configuration script, enter the command:

```
i4cfg -script
```

In response to the first question, select **1**; then respond to the questions as they are asked.

Command-Line Equivalent

To configure the network license client:

- With direct binding to connect to servers "thelma" and "louise":

```
i4cfg -a c -b "'network ip:thelma ip:louise'
'registry ip:thelma'" -n n
```

- With namespace binding that is joining an existing cell that has UUID 456b91c50000.0d.00.00.87.84.00.00.00:

```
i4cfg -a c -b null -n c -c 456b91c50000.0d.00.00.87.84.00.00.00
```

To achieve the same result as the direct binding example, **louise** and **thelma** must join the same cell.

Configuring to Reach a Global Location Broker in a Different Subnetwork

If your system does not support broadcasting or if the global location broker is running on a workstation in a different subnetwork, perform the following additional configuration steps on your network license servers, network license clients, and central registry license server to enable them to reach the global location broker:

1. Create a file called `glb_site.txt` in the directory:

   `/etc/lum/ncs` (on UNIX)

   `/etc/ncs` (on AIX)

   In the file, make one line for the address of each server that runs the global location broker that you want to enable this workstation to reach. Each address has the following form:

   ```
   ip:host
   ```

   where `host` is the TCP/IP hostname or the ip address. In the latter case, use a leading # to indicate that the host is an address and is in the standard numeric form (for example, #192.9.8.7 or #515c.111g).

   Blank lines and lines beginning with # are ignored.

   This is a sample of a glb_site.txt file:

   ```
   ip:charlie
   ip:#192.9.8.7
   ```

2. If the workstation belongs to an alternate cell, copy the file: glb_obj.txt from the server running the global location broker into the workstation being configured. Put the same value in the NCSCell tag of the configuration file (see Appendix A, "License Use Management Runtime Configuration File," on page 239).

Starting and Listing Your Servers

ALL

When you finish your configuration, issue the command:

```
i4cfg -start
```

to start the server you have configured on your workstation.

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

```
i4cfg -list
```

License Use Management Runtime Processes Defined as Windows Services

Windows

In Windows XP, Windows 2000Windows Server 2003, and Windows Vista, you can start and list your processes either by treating them as Windows services or by using the service manager tool (see"Managing License Use Management Runtime with Service Manager Tool").

During installation, your License Use Management Runtime processes are defined as services on Windows XP, Windows 2000, and Windows Server 2003. You can view them in the **Start** > **Settings** > **Control Panel** > **Administrative Tools** > **Services** window for Windows Server 2003 and XP or **Control Panel** > **Administrative Tools** > **Services** for Windows Vista. 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 workstation. 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 Management Runtime or one or more of its features, 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 Management Runtime processes are logged in the Windows XP, Windows 2000, and Windows Server 2003 log. To view them, go to the **Start** --> **Programs** --> **Administrative Tools (Common)** --> **Event Viewer** window, and select **Log** --> **Application**.

Managing License Use Management Runtime with Service Manager Tool

When you finish your configuration, double-click the Service Manager Tool icon in the License Use Runtime folder to start the processes. The **Service Manager Tool** window is displayed, as shown in Figure 34 on page 95.
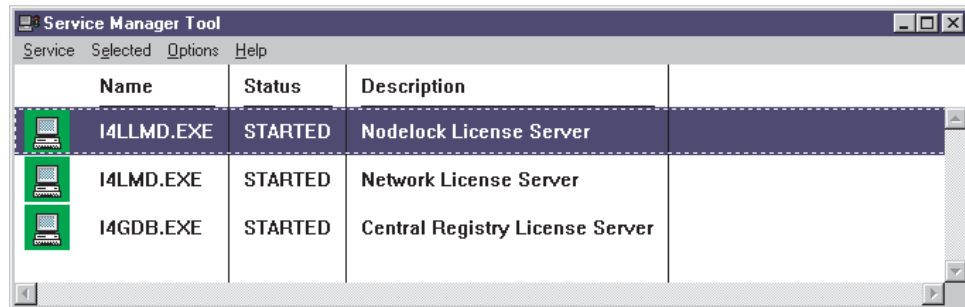
Figure 34. Service Manager Tool Window

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

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 6, "License Use Management Runtime Command Line Interface," on page 139.

License Servers on a System with Multiple Network Interfaces

This section explains special considerations for a license server on a workstation that has more than one network interface (for example, token ring and Ethernet). These considerations apply to:

- Network license servers
- The central registry license server
- Nodelocked license servers whose licenses are administered by instances of the basic license tool on remote workstations

A license server can listen and offer its services on any network interface.

In a direct binding environment, regardless of the TCP/IP LAN configuration, there are no connection problems in contacting the server. Be sure that the workstation that is contacting the server uses the hostname of the license server workstation on the network interface to which the contacting workstation's LAN is attached.

In a namespace binding environment, special care in configuring the TCP/IP LAN environment is required to prevent connection problems.

When you start the license server, it registers information such as the family (TCP/IP), address, and port in the namespace (glbd) database, the data that enables clients and other servers to locate the license server itself. If the license server registers with its token ring address, only token-ring connections will work; if it registers with the Ethernet address, only Ethernet connections will work. The license server retrieves its address by issuing a system call that returns the first value found, independent of the type of network interface (such as token-ring or Ethernet). This address retrieval is carried out by querying, in order, domain name server, if any; network information services (NIS), if any; and the /etc/hosts file.

This section illustrates some potential customer scenarios with network license clients using namespace binding and shows how to make them work.

In these examples, the network license server is on a system called *LSS*. Some computers (called License Clients A) are connected to LSS through token ring LAN A, and other computers (called License Clients B) are connected to LSS through Ethernet LAN B.

These examples are equally applicable if the workstation labeled "License Server System" is a nodelocked license server and the workstations labeled "License Management Clients" are other license servers from which licenses on the nodelocked license server are administered.

Example 1: Network License Clients on Two LANs

In this example, the customer wants LAN A computers and LAN B computers to communicate correctly via TCP/IP and wants the network license server to serve both License Clients A and License Clients B. See Figure 35.
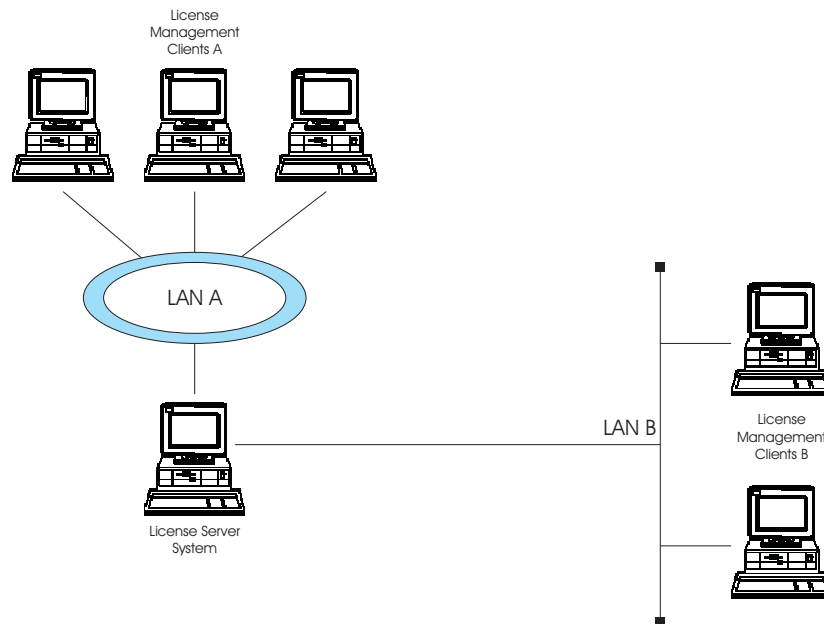
Figure 35. Network License Clients on Two LANs

To set up TCP/IP communication between LAN A and LAN B, the LSS workstation must be configured as a TCP/IP gateway between the two LANs. To do this, either:

- Set LSS as the default gateway in all the computers on both LANs, or
- Add a new entry in the routing tables of all the computers on both LANs, in particular:
  - In the LAN B computers: IP packets for LAN A go to LSS
  - In the LAN A computers: IP packets for LAN B go to LSS

In either case, LSS must have `ipforwarding=1` to enable the exchange of IP packets between the two LANs.

Example 2: Network License Clients on One LAN

In this example, the hardware configuration is the same as in Example 1, but the customer does not want LAN A computers and LAN B computers to communicate at all. The customer wants the license server to serve only Clients A. There are two possible solutions:

Solution 1

To set up TCP/IP communication between LAN A and LAN B, the LSS workstation must be configured as a TCP/IP gateway between the two LANs. (Note that it does not actually function as a gateway.) To do this, either:

- Set LSS as the default gateway in all the computers on LAN A, or
- Add a new entry in the routing tables of all the computers on LAN A; in particular, in the LAN A computers: IP packets for LAN B go to LSS

In either case, LSS must have ipforwarding=0 to disable the exchange of IP packets between the two LANs but enable LSS to read packets for LAN B.

Solution 2

Without configuring LSS as a TCP/IP gateway or changing the routing tables, it is possible to force the choice of adapter on which the license server must start. In the configuration file (`i4ls.ini`), set the parameter **UseHostTable=yes** to have the license server reverse the search order (first hosts file, then domain name server, and NIS). If you put the token-ring entry before the Ethernet entry in the hosts file, the license server will register using the address of the token-ring adapter.

Example 3: Internet Gateway

This scenario is just like Example 1 except that the customer intranet is connected to the Internet by an Internet gateway, **IG**, which is specified as a default gateway for Clients B. As in Example 1, the customer wants LAN A computers and LAN B computers to communicate correctly via TCP/IP and wants the network license server to serve both License Clients A and License Clients B. See Figure 36 on page 98.
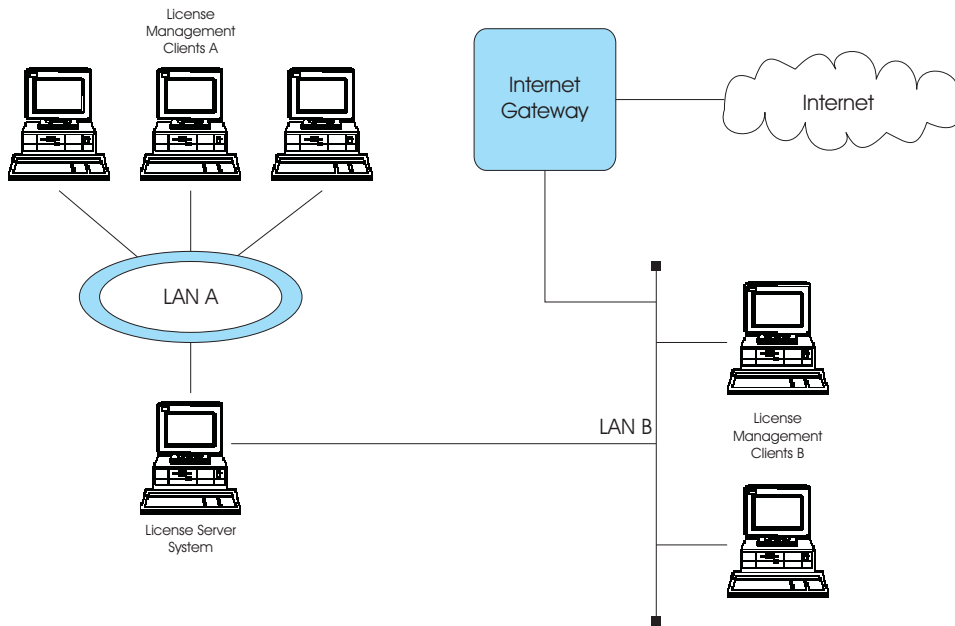
Figure 36. Internet Gateway Connection

In this case, instead of changing the TCP/IP configuration of all computers on both LANs as was required in Example 1, it is enough simply to add a new entry in the IG routing table, so that all packets for LAN B will go to LSS.

Example 4: Clients and Servers Separated by a Firewall

In this example, the customer wants LAN A computers and LAN B computers to communicate correctly via TCP/IP and wants the network license server to serve both License Clients A and License Clients B. See Figure 37.
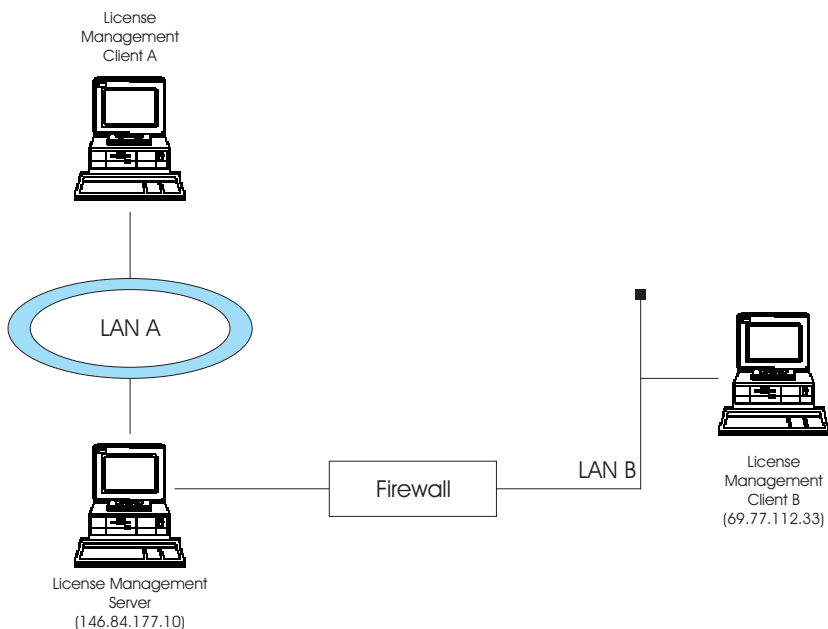

Figure 37. Network License Clients Separated by a Firewall

To set up communication between the license management server and client B, the firewall must be configured. To do this:

- Fix the secondary authentication server port. Edit the `i4ls.ini` file on the license management server and set the **ipAuthPort** value to 9900.
- Define new rules on the firewall to let inbound UDP packets for the destination 146.84.177.10, and ports 1515 and 9900 pass through.
- Define rules to let outbound UDP packets from the source 146.84.177.10 and ports 1515 and 9900 pass through.
- Let ICMP packets to and from the license management server pass through

Chapter 5. Administering License Use Management Runtime

This chapter contains scenarios that illustrate the daily tasks needed to manage license-enabled products.

The tasks can be done using either the basic license tool GUI (*BLT GUI*) or the command-line interface (*CLI*). The scenarios use the GUI. The equivalent CLI commands are shown at the end of each scenario.

Attention: If you want to run the scenarios, you must substitute your own values for the sample or fictitious names: **server name**, **product name**, **vendor name**, **enrollment certificate name**, **product version**, and **user name**.

The scenarios assume that the server **lab15099** is running the nodelocked license server, a network license server, and the central registry license server. The basic license tool is run from **lab15099**.

The scenarios use three sample 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`.

The enrollment certificate files can be placed in any directory, in these scenarios the administrator has placed them in the `home` directory.

Using the Basic License Tool GUI

This section gives some general guidance for using the Basic License Tool GUI.

Starting the Basic License Tool GUI

To start the basic license tool GUI,

- On UNIX and AIX: login with root authority and issue the command:

  `i4blt`

- On Windows: double-click the **Basic License Tool** icon in the **License Use Management Runtime** folder or start the BLT GUI by issuing the command:

  `i4blt`

The main window appears (Figure 38 on page 102).

Figure 38. Basic License Tool GUI - Main Window

The products shown 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

Before viewing information about concurrent, reservable, per-server, and concurrent nodelocked licenses in use for a product, it is a good idea to clean up stale licenses. In the main window:

1. Highlight the product.
2. Select **Clean up stale licenses** from the Selected menu.

If you turned the automatic cleanup feature on (see Appendix A, "License Use Management Runtime Configuration File," on page 239) all stale licenses are cleaned up automatically at each BLT refresh.

Selecting Servers, Vendors, and Licenses

To reduce the number of servers, vendors, or licenses shown in the window:

- Use the **Include...** option of the View menu. You can specify which servers and vendors are shown as of the next refresh.
- Use the **Settings...** option of the Options menu. You can specify the types of licenses (nodelocked, network, or both) to be shown as of the next refresh.

Note: If a selected server shuts down, it is no longer selected, even if it is restarted.

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 of the View menu, or deselect or restart the failing server and try again.

Performing Basic Administration

The scenarios in this section demonstrate how to:

- Enroll a licensed product ("Enrolling the Product" on page 103).
- Distribute licenses from a compound password ("Distributing the Licenses" on page 105).
- Get a report on the use of licensed products ("Generating Reports" on page 106).

- Check the current users of licensed products ("Checking the Number of Product Users" on page 107).
- Reserve reservable licenses for specific users and monitor the use of reservable licenses ("Scenario 2: Managing Reservable Licenses" on page 108).

Scenario 1: 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 main window, select **Enroll** from the Products menu, then **Single product...**. The Enroll Product window is displayed (Figure 40 on page 104).
2. Select the **Import** button. The Import window is displayed. See Figure 39 for an example of the Import window.
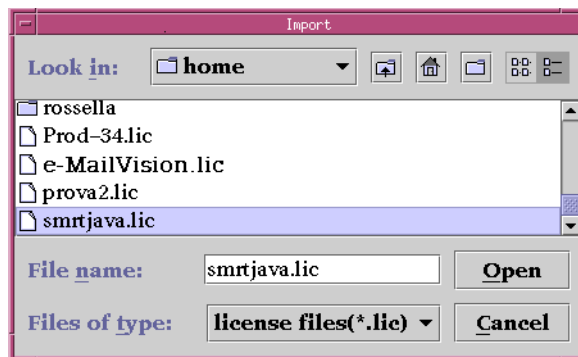

Figure 39. Import Window for Custom Configuration

   a. In the **Look In** drop-down list, select the directory containing the enrollment certificate.
   b. From the list of files, select the enrollment certificate. For the SMARTJava product, the file is **smrtjava.lic**.
3. Click **Open**. The Enroll Product window is redisplayed, filled in with the information from the enrollment certificate file. See Figure 40 on page 104.

Figure 40. 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 **lab15099**.

   Notes:
   a. 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 workstation.
   b. 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.
   c. If the product is customer-managed and has nodelocked licenses, select a nodelocked license server.
   d. 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.

5. Click **OK**.

   Because the product is customer-managed, the Enroll Licenses window is displayed.

6. 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 41 on page 105.
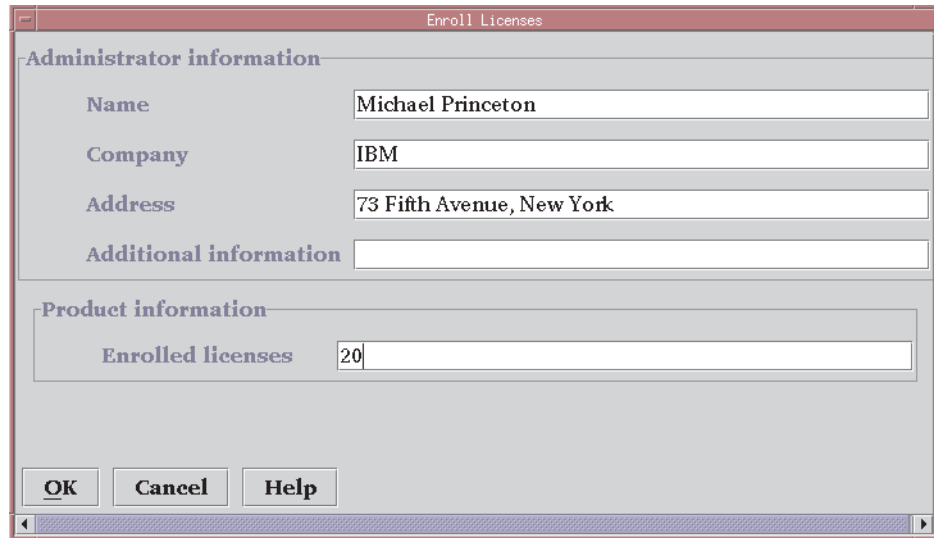
Figure 41. Enroll Licenses Window

7. Click **OK**. A message is displayed indicating that the product has been
    enrolled.

    The main window is displayed again, as shown in Figure 42. There is a new
    line corresponding to SMARTJava, and the icon for the product (a wrapped
    box) indicates that the product has a compound password.
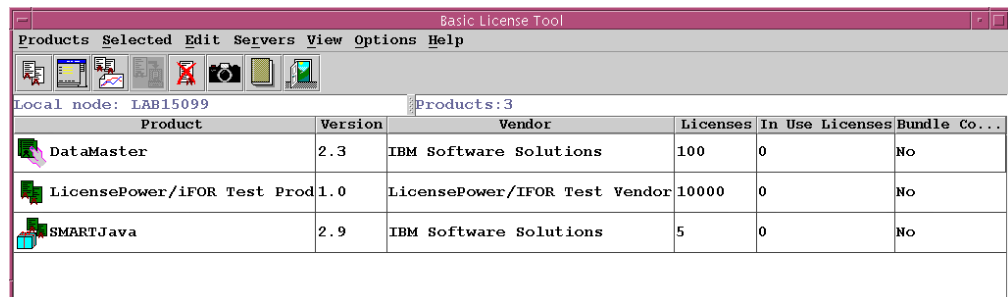
Figure 42. Main Window with SMARTJava Enrolled

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.

Network licenses delivered with a compound password must be distributed.
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 **lab15099**:

1. In the main window, select the line for the product.
2. Select the **Distribute licenses...** option from the Selected menu. The Distribute
    Licenses window is displayed, as shown in Figure 43 on page 106.

Figure 43. Distribute Licenses Window

3. In the Distribute Licenses window, select the network license server. Right-click the selected server, and select the **Set number of licenses....** option from the pop-up menu. The Set Number of Licenses window is displayed.
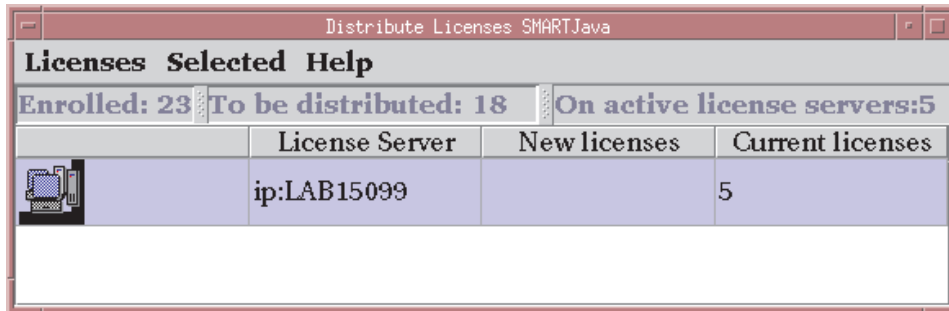
4. Enter **5** in the Number of licenses field, as shown in Figure 44, and click **OK**.
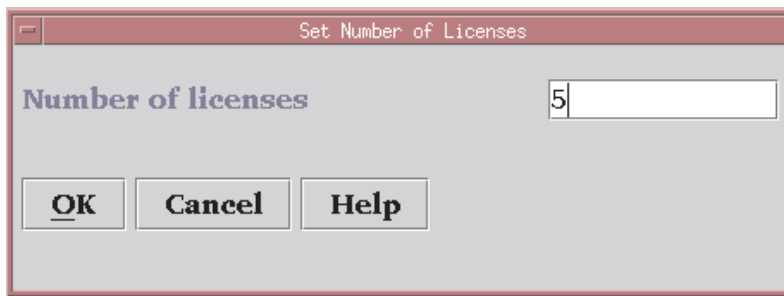
Figure 44. Set Number of Licenses Window

The Distribute Licenses window is redisplayed. The window now shows 15 as the number of licenses available to be distributed. Select the **Distribute** option from the Licenses menu to confirm data and distribute the licenses.

5. In the main window, the number of available licenses changes to 5 and the icon for SMARTJava changes to show a compound password with distributed concurrent licenses.

End users can now use the SMARTJava product.

Generating Reports

This section shows how to get a report of SMARTJava usage for one month: from July 2 to August 2, 2004.

1. Select SMARTJava in the main window (Figure 38 on page 102).

2. Select the **Reports...** option from the Selected menu.

The Reports window is displayed.

Note: You can not generate reports if a selected server shuts down. For more information see "Selecting Servers, Vendors, and Licenses" on page 102

3. Fill in the Reports window as shown in Figure 45 on page 107.

Figure 45. Reports Window

    a. Select the report type **License requests by product**.

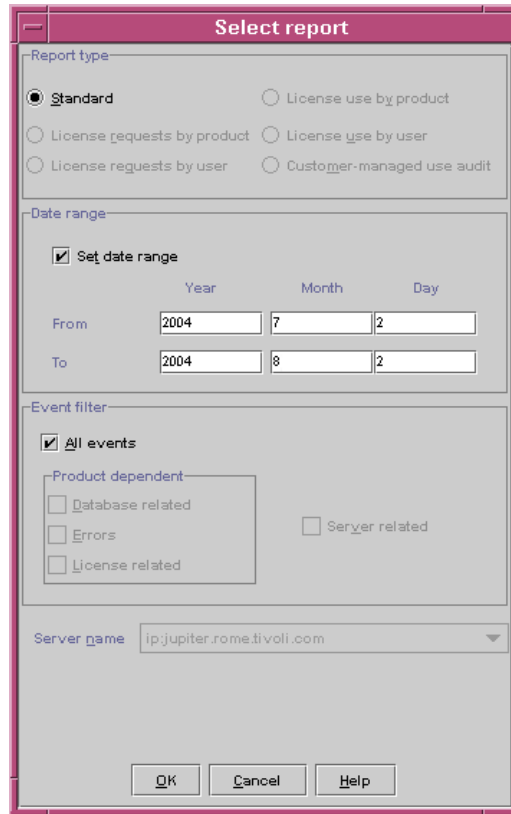   b. Check the **Set date range** check box and set the date range for the report in the **From** and **To** fields (in this example, from July 2 to August 2, 2004).

    c. Click OK.

4. View the Report window, which contains the following report:

```
========================================================
L i c e n s e   R e q u e s t s   B y   P r o d u c t
========================================================


                                 Licenses   Licenses   Percent
Vendor/Product            Vrsn   Requested  Granted    Rejections
------------------------- ----   ---------  --------   -----------
IBM Software Group        2.3      120        120          0
SMARTJava


        =========================================
        *** End of License Requests By Product ***
        =========================================
```

The report shows that 120 requests were made to use SMARTJava Version 2.3, and all of them were granted.

5. Click **Cancel** in the Reports window.

Checking the Number of Product Users

To check the number of concurrent users of SMARTJava:

1. Select the line corresponding to the SMARTJava product on the main window.

2. Select the option **Open as Details** from the Selected menu. The Details notebook is displayed.

3. Select the **Concurrent Users tab**. The Concurrent Users page is displayed, as shown in Figure 46.

Figure 46. Concurrent Users Page

It shows the number of licenses in use, with the name of each user. In this example, the user root is using two SMARTJava licenses.

4. Close the notebook.

Command-Line Equivalent

The GUI scenario shows how to enroll SMARTJava licenses on the central registry license server, distribute five of the licenses to network license server *lab15099*, get a report on one month's usage, and check the number of concurrent users.

The equivalent CLI commands are as follows.

- To enroll the product on the central registry license server:
  - ```
    i4blt -a -n lab15099 -f smrtjava.lic -T 20
      -R "'Michael Princeton' IBM '73 Fifth Avenue, New York'"
      -I "'First installed by Ian'"
    ```

If you use the command line interface to enroll a product, check the top of the enrollment certificate file. The vendor may have generated the command to be used.

- To distribute five licenses to network license server:
  - ```
    i4blt -E -n lab15099 -v "'IBM Software Group'" -p "SMARTJava 2.3"
      -A 5 -w lab15099 -I "'Cooper using root'"
    ```
- To generate a report of requests for SMARTJava from July 2 to August 2, 2004:
  ```
  i4blt -r2 -p "SMARTJava" -b 07/02/2004 -g 08/02/2004
  ```
- To display information about concurrent users of SMARTJava:
  ```
  i4blt -s -lc -p "SMARTJava"
  ```

For more information about the command line interface, see Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Scenario 2: Managing Reservable Licenses

In this scenario, the administrator manages licenses of the DataMaster product. This scenario shows you how to:

- Reserve some reservable licenses for the exclusive use of a specified user
- Monitor usage of reserved licenses by the users for which they were reserved
- Monitor use of unreserved reservable licenses by other users

In this scenario, 100 reservable licenses for DataMaster have already been enrolled. The enrollment process is the same as for concurrent licenses of a

customer-managed use product, as shown in "Enrolling the Product" on page 103, except that the Enroll Licenses window is not used. After enrollment, the product appears in the main window. Note that the icon (a hand holding some licenses) indicates that the licenses are reservable.

To reserve some of the licenses for a specific user and monitor use of the reservable licenses:

1. Double-click the product in the main window and go to the **Reservable tab** of the Details notebook, as shown in Figure 47.
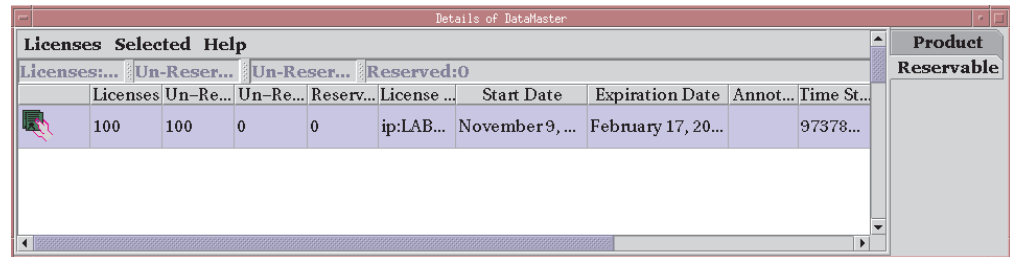

Figure 47. Details Notebook with Reservable Tab. Custom configuration serial number not shown.

Note that there are 100 total licenses, with none in use and none reserved.

2. Right click the product, and select **Reserve...** from the pop-up menu. The Reserved Licenses - Details window is displayed.

3. Fill in the number of licenses you want to reserve; the user, group, and node ID (alone or in any combination) for which you want to reserve licenses; and the date and time that the reservation is to expire, as shown in Figure 48. The latest allowed expiration date of a reservation is 12/31/2037.

   Note: If the reservation is made for a product that is exploiting the named user policy, the **Name** field must always be completed. A blank **Name** field means any user, and is not allowed for the named user policy. **Group name** field has length limited to 31 characters. Any name longer than 31 characters will be truncated. **Node ID** has length limited to 31 characters. Any name longer than 31 characters will be truncated. Uniqueness of **Node ID** is preserved.
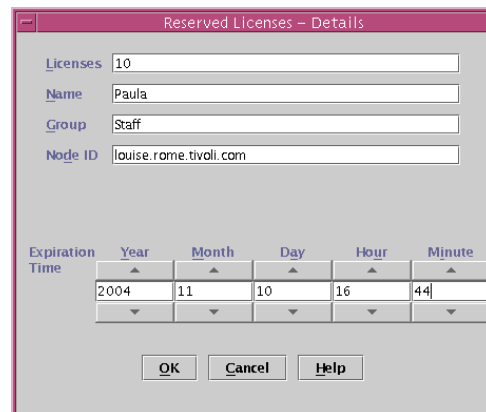

Figure 48. Reserving Reservable Licenses

In the figure, the administrator is reserving ten licenses for user Paula, a member of the group staff, for use on the node `louise.rome.tivoli.com`. The reservation is to expire on November 10, 2004 at 16:44. Note that normally you reserve one license for a specific user and more than one license for a group or node.

4. Click **OK**.

5. The **Reserved** page is added to the Details notebook, as shown in Figure 49.

Figure 49. Details Notebook with Reserved Licenses

The example shows 10 licenses reserved for the user Paula in the group staff.

6. Now assume the user Paula requests a license, the **Reserved Users** page is added to the notebook details, as shown in Figure 50. To see the change, you need to refresh the window (using the F5 keyboard key, or the menu option **Licenses** then **Refresh now**).

Figure 50. Reserved Users Page with Reservable Licenses in Use

In this example, the **Reserved Users** page shows that the user Paula is using one reserved license.

7. Now assume that the user root requests a license to DataMaster. No licenses have been reserved for root, but there are 90 unreserved licenses. Such licenses are available to all users until the administrator reserves them. One of those licenses is granted to root. It is managed exactly like a concurrent license.

The **Un-Reserved Users** page is added to the Details notebook for DataMaster, as shown in Figure 51 on page 111.
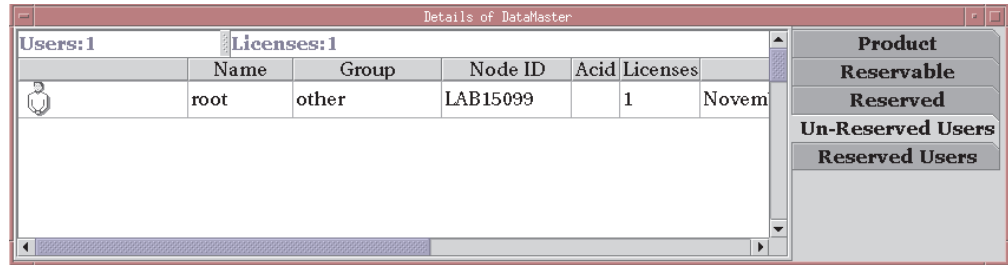
Figure 51. Un-Reserved Users Page

8. The Reservable page now shows 10 reserved licenses, 1 unreserved license in use, and 90 unreserved licenses, as shown in Figure 52.

Note: Steps 6, 7, and 8 do not apply in the case of a reserved license for a product using the named user policy. In this case, when the user Paula requests a license, the successful license grant event is logged and can be viewed in the reports. Other users cannot use the product, even in a concurrent way, because they do not have reservations.
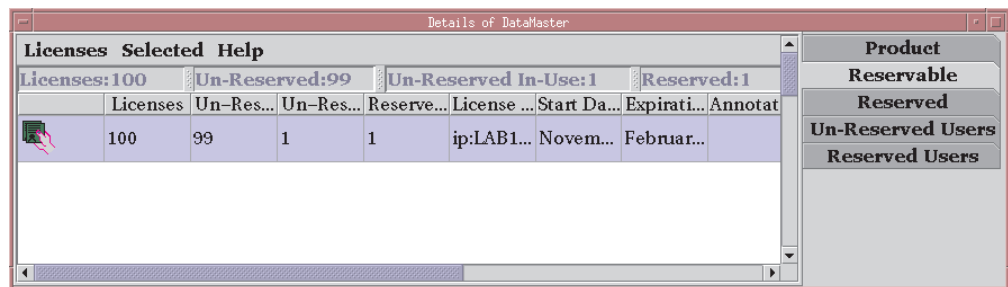

Figure 52. Reservable Page with Unreserved Licenses in Use

Command-Line Equivalent

The GUI scenario shows how to reserve ten DataMaster licenses for the user Paula, display information about that user's use of the reserved licenses, display information about the use of the unreserved licenses by other users, and display detailed information about the product.

The equivalent CLI commands are as follows.

- To get the timestamp of the licenses to be reserved:

```
i4blt -lp -i -v "'IBM Corporation'" -p "DataMaster"
```

- To reserve ten DataMaster licenses for the user Paula in group staff on node louise.rome.tivoli.com using the license password identified by timestamp 899460562:

```
i4blt -R r -v "'IBM Corporation'" -p "DataMaster 2.1a"
-t 899460562 -A 10 -g 11/10/2004 -H 16:44 -u "Paula staff
 louise.rome.tivoli.com"
```

- To display information about the users of reserved licenses:

```
i4blt -s -lrr -v "'IBM Corporation'" -p "DataMaster"
```

- To display information about the users of unreserved licenses:

```
i4blt -s -lru -v "'IBM Corporation'" -p "DataMaster"
```

- To display detailed information about the product, including the number of reserved and unreserved licenses:

```
i4blt -lp -i -v "'IBM Corporation'" -p "DataMaster"
```

Exercising Customer-Controlled Policies

The scenarios in this section explain how to exercise the customer-controlled policies outlined in "Customer-Managed Use Products" on page 5. The scenarios show how to:

- Switch from per-server to per-seat licenses ("Scenario 3: Switching from Per-Server to Per-Seat Licenses")
- Use the hard stop/soft stop policy ("Scenario 4: Using the Hard Stop/Soft Stop Policy" on page 113)
- Update the number of licenses of a customer-managed use product ("Scenario 4: Using the Hard Stop/Soft Stop Policy" on page 113)
- Control the set of users who are permitted to use a specific application ("Scenario 5: Restricting User Access" on page 115)

Scenario 3: Switching from Per-Server to Per-Seat Licenses

In this scenario, the administrator switches the policy for the product e-MailVision from per-server to per-seat. The per-server license has already been enrolled. The enrollment process is the same as for customer-managed concurrent licenses, as described in "Enrolling the Product" on page 103.

The icon for e-MailVision shows per-server licenses.

To do the switch:

1. Install the per-seat license, importing the per-seat enrollment certificate emailvpt.lic and enrolling 50 licenses. The enrollment process is the same as for customer-managed concurrent licenses, as described in "Enrolling the Product" on page 103.

2. Select e-MailVision in the main window (Figure 38 on page 102). Select the **Update licenses...** option from the Selected menu. The Details notebook is displayed.

3. Click the **Per-Seat tab**. Now right-click the product, and select **Update licenses...** from the pop-up menu. The Update Licenses window is displayed. Check the box **Enable per-seat licensing**, as shown in Figure 53 on page 113.
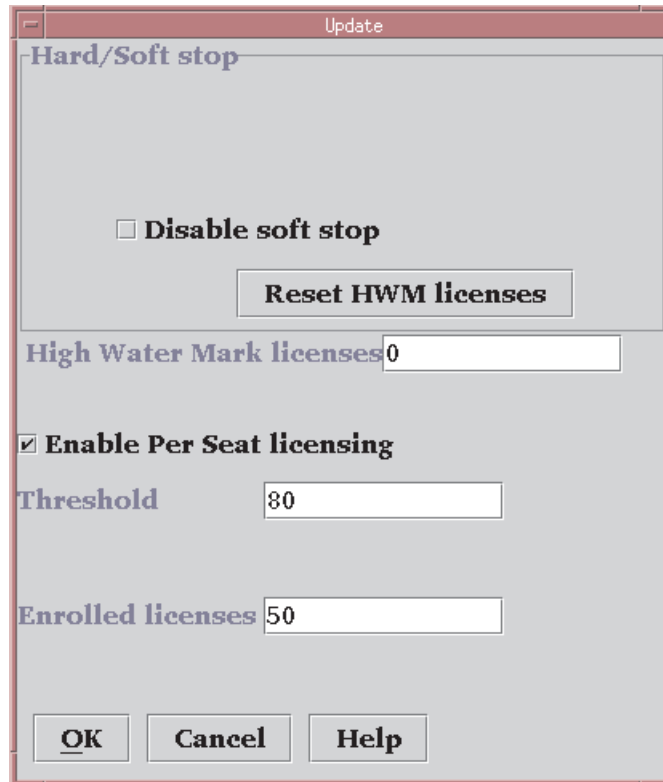
Figure 53. Enabling Per-Seat Licensing

4. Click **OK** to make the change.
5. In the main window the icon changes to show per-seat licenses. The license server will remove the old per-server licenses.

Command-Line Equivalent

The GUI scenario shows how to switch the e-MailVision product from per-server licensing to per-seat licensing.

The equivalent command line interface commands are as follows.

- To enroll the per-seat licenses for e-MailVision:

  `i4blt -a -f /emailvpt.lic -T 2 -R "'Keith Hogson'"`

- To switch e-MailVision from per-server to per-seat licensing:

  `i4blt -U -v "'IBM Software Solutions'" -p "e-MailVision 1.2" -S yes`

Scenario 4: Using the Hard Stop/Soft Stop Policy

In "Scenario 1: Managing a Licensed Product" on page 103, the administrator enrolled the SMARTJava product and distributed five licenses from a network compound password. Now the administrator has distributed the remaining 15 enrolled licenses. Because the vendor enabled this product to use the hard stop/soft stop policy, and the administrator is running it with soft stop set, it is possible that more than 20 licenses are being used at any given time.

In this scenario, the administrator:

- Checks the current number of licenses in use and the maximum number of licenses that have been granted beyond the 20 enrolled (the high-water mark, which in this example is 10)

- Decides to acquire ten more licenses
- Updates the number of enrolled licenses to 30
- Resets the high-water mark to 0
- Distributes ten more licenses from the compound password

Follow these steps:

1. Check the current license usage in the main window (Figure 54).

Figure 54. Main Window with Soft-Stop Licenses in Use

The window shows that ten licenses beyond the 20 enrolled are in use.

2. Check the longer-term license usage:

   Select SMARTJava in the main window. Select **Update licenses...** from the **Selected** menu. The Details notebook is displayed.

3. Go to the Concurrent page, right-click an entry for the product, and select **Update licenses...** from the pop-up menu. The Update Licenses window is displayed (Figure 55).

Figure 55. Update Licenses Window with High-Water Mark

The **High Water Mark licenses** field shows that **10** is the maximum number of soft-stop licenses ever in use at one time since the high-water mark was last reset.

4. Decide to acquire ten more licenses, and pay the vendor for them.

5. Returning to the Update Licenses window, enter **30** in the **Enrolled licenses** field. At the same time, reset the high-water mark to **0** by clicking on **Reset HWM Licenses**.

6. Click OK. A confirmation message is displayed.

7. Distribute the ten newly enrolled licenses (see "Distributing the Licenses" on page 105). In the main window (Figure 38 on page 102), the Licenses column for SMARTJava is updated to 30.

Command-Line Equivalent

The GUI scenario shows how to check usage of soft stop licenses for SMARTJava, enroll and distribute an additional ten licenses, and reset the high-water mark to **0**.

The equivalent command line interface commands are as follows.

- To display information about usage of soft stop licenses of SMARTJava:

  ```
  i4blt -lp -p "SMARTJava" -i
  ```

- To update the enrollment, enrolling ten more licenses on the central registry license server:

  ```
  – i4blt -U -v "'IBM Software Group'" -p "SMARTJava 2.3" -T 30 -I
      "'Margaret using root'"
  ```

- To reset the high-water mark to 0:

  ```
  i4blt -U -v "'IBM Software Group'" -p "SMARTJava 2.3" -M
  ```

- To distribute the ten licenses to network license server **lab15099**:

  ```
  i4blt -E -n lab15099 -v "'IBM Software Group'" -p "SMARTJava 2.3"
   -A 10 -w lab15099
  ```

Scenario 5: Restricting User Access

There are two methods for restricting user access;
- "Restricting User Access With the User File"
- "Restricting User Access Using the Basic License Tool" on page 116

Restricting User Access With the User File

This scenario explains how to create a user file to designate that certain users may or may not use certain products. You could also use a previously created user file as a base. To create a user file, follow these steps:

1. Using a text editor, open a file named:
   - user_file on AIX and UNIX
   - userfile on Windows

2. Within the file, to restrict access to a product, use the **VENDOR** keyword, followed by the name of the vendor, followed by either **ALL** (meaning all products of this vendor) or the name of a product. Enclose vendor names and product names in quotation marks if they contain embedded blanks. For example:

   ```
   VENDOR "IBM Software Group" SMARTJava
   VENDOR Grafix,Inc. all
   ```

   You need one VENDOR statement for each product of the same vendor, unless ALL is sufficient for your purposes.

3. After each vendor statement, code either an `ALLOW` or a `DISALLOW` statement. `ALLOW` and `DISALLOW` are mutually exclusive.

   - `ALLOW`

     Specifies that the user names that follow this keyword are allowed to use the product. If no user names follow this keyword, no users can use the product. The user name is the login user name. For example:

     ```
     VENDOR "IBM Software Group" SMARTJava
     ALLOW fritz harry monique penny
     ```

     This specifies that only four users can use the SMARTJava product: Fritz, Harry, Monique, and Penny. `ALLOW` and `DISALLOW` are mutually exclusive.

   - `DISALLOW`

     Specifies that the user names that follow this keyword are not allowed to use the product. If no user names follow this keyword, all users can use the product. The user name is the login user name. For example:

     ```
     VENDOR Grafix,Inc. all
     DISALLOW heather jason
     ```

     This specifies that all users *except* Heather and Jason can use all Grafix, Inc software products.

4. Store the file in the `/var/lum/` directory on UNIX, the `/var/ifor/` directory on AIX and the `ifor\ls\conf\` directory on Windows. Store it on the workstation where the licenses to be restricted are installed. In this example, the complete user file is:

   ```
   % This line is a comment
   % ******
   VENDOR "IBM Software Group" SMARTJava
   ALLOW fritz harry monique philip
   % ******
   VENDOR Grafix,Inc. all
   DISALLOW heather jason
   ```

5. *policyensuring consistency in user authorization*For a consistent user authorization policy, store the same user file on all network license servers and nodelocked license servers in your environment, including the central registry license server. User file is processed until first matching rule is found. If matching rule is found, processing of user file ends. For example:%

   ```
   This line is a comment
   VENDOR Grafix, Inc. all
   DISALLOW heather jason harry
   % ******
   VENDOR Grafix, Inc. "Special Product"
   ALLOW jason
   ```

   User jason will not get any license for vendor Grafix, Inc., because the first rule in the user file does not allow it. The second rule is not processed.

   Default policy for licenses which are not listed in user file is ALLOW.

6. When adding a new product, remember to update user files at all the license servers accordingly.

Restricting User Access Using the Basic License Tool

This scenario explains how to restrict user access using the basic license tool. This scenario is applicable only to concurrent licenses. This scenario shows you how to:

- Modify the default authorization.
- Create authorization records for a specific concurrent license. This allows the authorized users to take the license.
- Modify a created authorization.

To manage this kind of license:

1. Double-click the product in the administration tool main window, then click the **Concurrent** tab. The Concurrent Licenses - Details window is displayed (Figure 56).
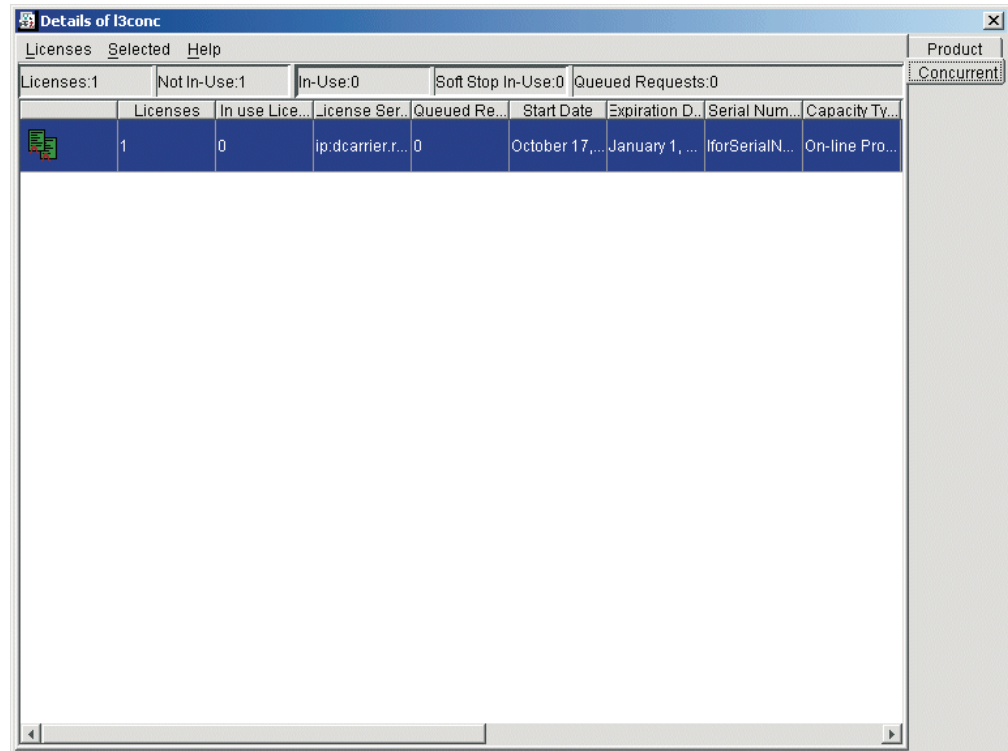

Figure 56. Concurrent Licenses - Details window

2. To modify the default authorization, from the Concurrent Licenses - Details window, right-click on the product and select **Selected ▶ Authorizations**. The Authorization window is displayed (Figure 57 on page 118). The default setting is for all licenses to be allowed.
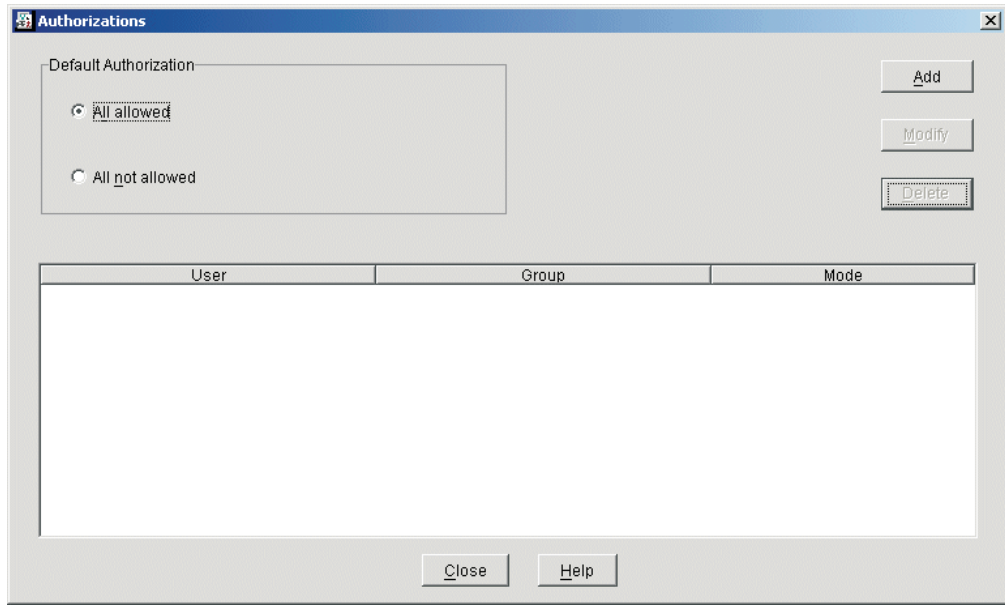
Figure 57. Authorization window

For the selected license, the window shows the mode in which each user can use the license. There are two different modes for license use:

- **All allowed**. Any user can start using the license.
- **All not allowed**. No users are allowed to use the license. A user requires specific authorization to use the license.

3. You decide that users can use the license only if they have a specific authorization, so you click **All not allowed**. Create the first authorization record for a specific user. From the Authorization window (Figure 57), click **Add**. The Add Authorization window is displayed (Figure 58 on page 119). Fill in the details for the new User and Group (alone, or in any combination). Set the Mode for the use of the license for this specific user. The modes are similar to the modes in the Authorization window (Figure 57), except that they are for the individual user and group being authorized. Click **OK**

Note: Group name field has length limited to 31 characters. Any name longer than 31 characters will be truncated.

.

Figure 58. Add Authorization window

The authorization record has now been created (as shown in Figure 59).

Figure 59. Authorization window — showing new authorization

4. When you have created an authorization record, you can change the settings for the authorization record. Select the authorization record file in the Authorization window (Figure 59) and click on **Modify**. The Modify Authorization window is displayed. You decide to change the Group for the user Sandra from Guest to System. Type in the new group, then click **OK**.

   Note: Group name field has length limited to 31 characters. Any name longer than 31 characters will be truncated.
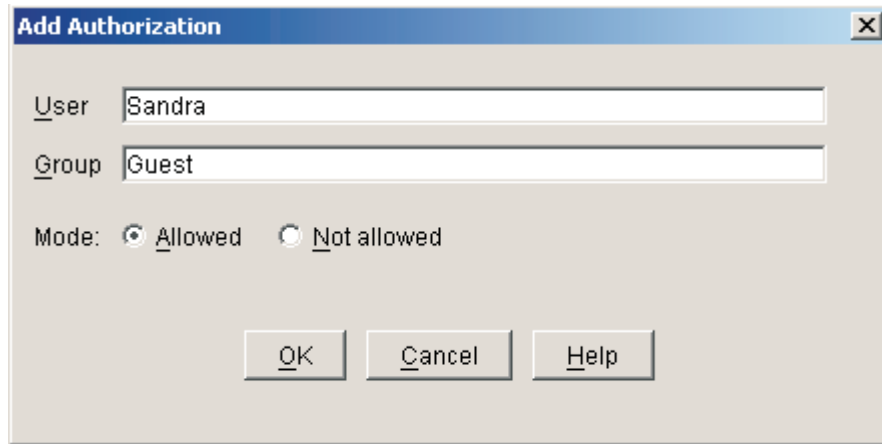
Figure 60. Modify Authorization window
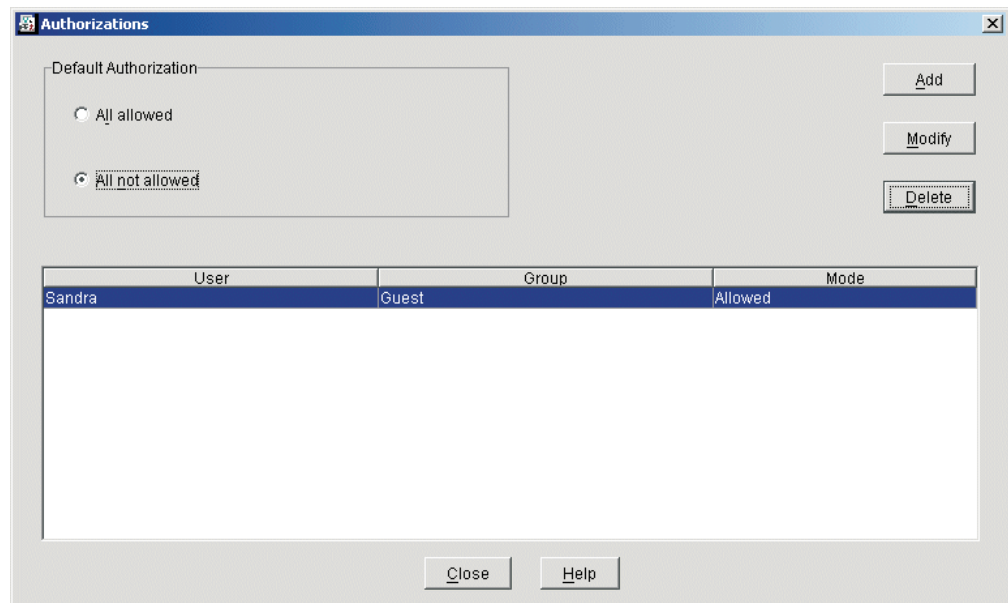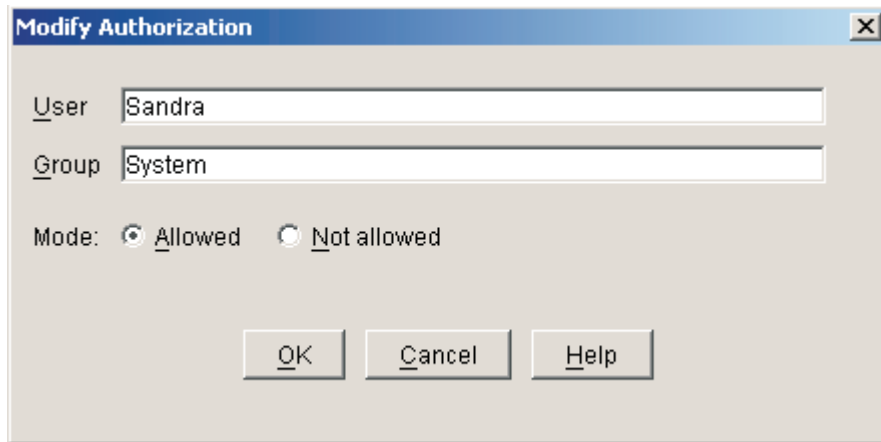
The authorization record has now been modified.

Note: The user is not allowed to change license authorization from remote machine, even if **RemoteAdministration** is enabled. Authorization change is possible only on local machine.

Managing High-Availability Licensing (HAL)

The scenario in this section shows how to set up and manage a cluster of network license servers to ensure high availability of concurrent licenses.

When you create a cluster, License Use Management Runtime generates the cluster ID. For a software vendor to create passwords that are bound to a cluster rather than to a single server, you must provide the cluster ID to the vendor. Therefore, you must create the cluster before you can request licenses bound to the cluster from a software vendor.

Scenario 6: Creating and Managing a Cluster

In this scenario, the administrator:
- Creates a cluster consisting of four network license servers (members)
- Activates all the servers in the cluster
- Adds a fifth server to the cluster
- Deactivates a cluster member

Note: Due to incompatibility between HAL cluster members 4.6.4 and HAL cluster members of previous releases of License Use Management Runtime, you cannot create a cluster using a mixed server environment.

Different license servers (both in terms of License Use Management versions and operating systems) can coexist in the same cluster, provided they are either at License Use Management 4.5.5 to 4.6.2, or License Use Management 4.6.4 or higher.

Creating a Cluster
To create a cluster:

1. Before you begin, be sure all the servers you intend to put in the cluster are configured to communicate with each other through direct binding or that they have all joined the same namespace binding cell.

2. In the main window (Figure 38 on page 102), select **Servers** from the menu bar and the **Clusters...** option from the menu. The Clusters window is displayed, as shown in Figure 63.

3. Select **Clusters** from the menu bar and **Create...** from the menu. The Create Cluster window is displayed.

4. Use the **Initial Number** spin box to specify how many servers you intend to put in the cluster. Make sure that this number is what you want, because it determines the maximum number of members the cluster can ever have and the minimum number of activated members necessary to run the cluster. In this example, as shown in Figure 61, the administrator specifies that the initial number of members is 4.
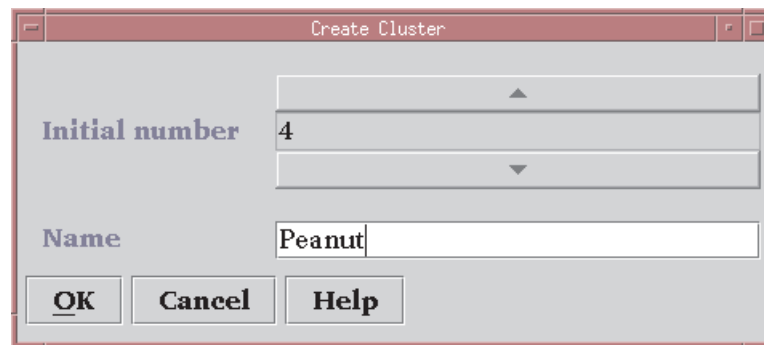

Figure 61. Create Cluster Window

Use the **Name** field to give the cluster a unique name. You will use the name to specify the cluster in GUI and command-line operations. In this example, the administrator names the cluster *Peanut*.

5. Click **OK**. A pop-up message gives you a chance to change the initial number of members.

6. After you confirm your choices, the Define Cluster Members window is displayed.

   Potential members of the cluster are shown in the **Available NetworkLS** box. These are network license servers that are up and running, and are not activated in any other cluster.

   Select a server from the **Available NetworkLS** list and use the **Add>>** push button to move it to the **Cluster Members** list. Continue until the **Cluster Members** list has exactly the number of servers you specified in **Initial Number**. You can move servers between the **Available NetworkLS** and **Cluster Members** boxes until your selections are final. In the example shown in Figure 62, the administrator adds servers **coyote**, **rocky**, and **sprint**.

   Different license servers (both in terms of License Use Management versions and operating systems) can coexist in the same cluster, provided they are either at License Use Management 4.5.5 to 4.6.2, or License Use Management 4.6.4 or higher.
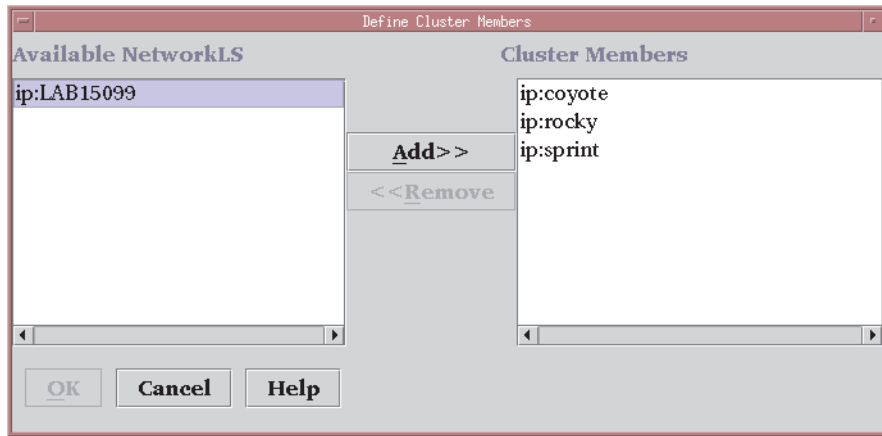
Figure 62. Define Cluster Members Window

7. Click **OK**. A pop-up window is displayed to give you the chance to change your selections. The Clusters window is redisplayed with an entry for the newly-defined cluster, as shown in Figure 63.
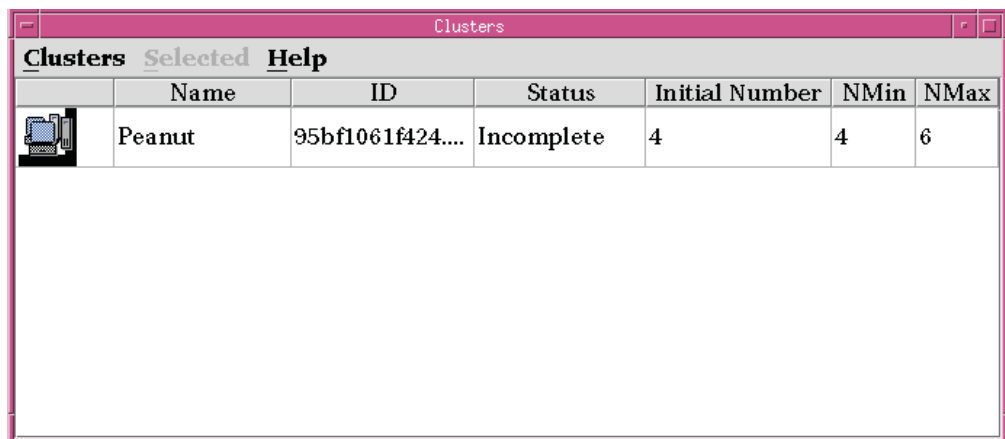
Figure 63. Clusters Window with New Cluster Added

Note that the status of the cluster is **Incomplete**, because not enough servers have been activated yet.

Activating Cluster Members

For a cluster to be able to serve licenses, a minimum number of servers must be activated, and more than half of the activated servers in the cluster must be available. When you create a cluster, only the first server (in this example, **coyote**) is activated. To activate the other servers, follow these steps:

1. In the Clusters window, highlight the cluster *Peanut*. Select **Selected** from the menu bar and **Open As Details...** from the menu. The Details of Peanut window is displayed, as shown in Figure 64.

Figure 64. Details of New Cluster

>   The status of server **coyote** is **Waiting**, because it has been activated but the cluster does not yet have enough activated members to start serving licenses. Next to each activated server is displayed the number of the port on which the server performs its high-availability licensing activities.

2. Select server **rocky**. Right-click it and select **Activate** to activate the server. Similarly activate servers **sprint** and **lab15099**.

   While each activation is being processed and all the servers in the cluster are being updated, the cluster goes into **Change Pending** status for a time that depends on the number of activated servers in the cluster. While the cluster is in this status, you cannot perform any administration on the cluster. Return to the main window and use F5 to refresh the display, and then view the Clusters window again until the cluster has exited from **Change Pending** status.

3. Return to the Details of Peanut window, where:
   - The status of servers **coyote**, **rocky**, and **sprint** has changed to **Serving**.
   - Each server is serving 33% of the licenses, as shown in Figure 65.
   - Server **lab15099** is in **Reserve** status, ready to take over if **coyote**, **rocky**, or **sprint** goes down.

Figure 65. Details of Cluster after Three Activations

4. Return to the Clusters window (Figure 64). The status of the cluster has changed to **Active**. The cluster ID has been generated and is displayed.

The administrator can now give the cluster ID to software vendors who will create passwords bound to the cluster.

	Name	ID	Status	Number	NMin	NMax
	Peanut	95bf1061f42...	Active	4	4	6

Figure 66. Clusters Window with Cluster ID

5. Return to the main window, where the HAL test product has been enrolled on the cluster. You can use it to test operation of the cluster.

Adding a Cluster Member

If a cluster has fewer than its maximum number of members, you can add members, one at a time, up to the maximum number. In this example, you can add additional members to **Peanut**, as follows:

1. In the Clusters window, highlight the **Peanut** cluster. From the **Selected** menu, select **Add cluster member...**. The Add Cluster Members window is displayed. The window is similar to the Define Clusters window shown in Figure 62.

   Potential members of the cluster are shown in the **Available NetworkLS** box. These are network license servers that are not activated in any other cluster. Although Windows Millennium servers or servers older than License Use Management 4.5.0 release may be listed, do not select them. They can not be members of a cluster.

   Select a server (in this example, **thelma**) from the **Available NetworkLS** list and use the **Add>>** push button to move it to the **Cluster Members** list. You can move servers between the **Available NetworkLS** and **Cluster Members** boxes until your selection is final.

2. Click **OK**. A pop-up window is displayed to give you the chance to change your selection. The Clusters window is redisplayed. Return to the main window and use F5 to refresh the display, and then view the Clusters window again until the cluster has exited from **Change Pending** status. The number of servers in **Peanut** is now **5**, and the status of the cluster is **Active**. The newly added server is automatically activated.

3. Open the Details of Peanut window (Figure 64). The status of servers **coyote**, **rocky**, and **sprint** is **Serving**, and each is serving one-third of the licenses. The status of **lab15099** and **thelma** is **Reserve**.

Deactivating a Server

If you want a server to stop functioning as part of a cluster, you can deactivate it. You can then activate the server in another cluster. You cannot, however, substitute another server for the deactivated server in the cluster where it is deactivated.

To deactivate a server in a cluster:

1. In the Details of Peanut window, select server **thelma**. Click it with the right mouse button and select **Deactivate** to deactivate the server. While the deactivation is being processed and all the servers in the cluster are being updated, the cluster goes into **Change Pending** status for a time that depends on the number of activated servers in the cluster. While the cluster is in this
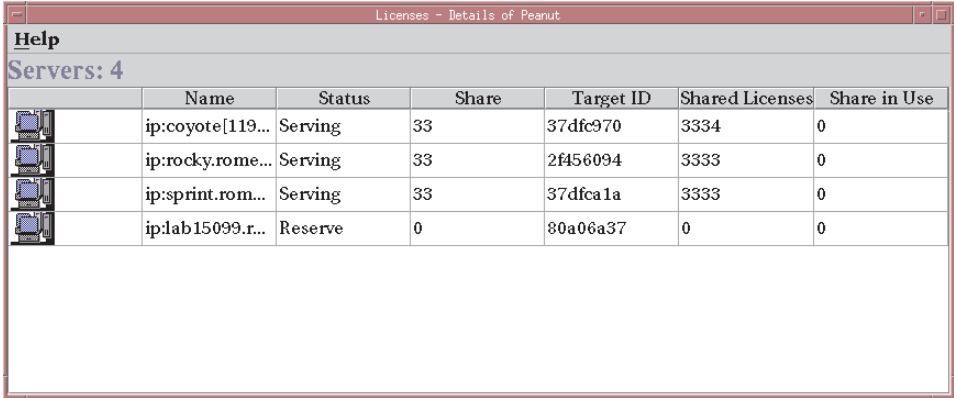
status, you cannot perform any administration on the cluster. Return to the main window and use F5 to refresh the display, and then view the Clusters window again until the cluster has exited from **Change Pending** status.

2. Return to the Details of Peanut window. The status of server **thelma** has changed to **Not Activated**.

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

Viewing Licenses Being Served

For a view of a cluster by product, go to a high-availability product (in this case, the HAL Test Product) in the main window. Open the product's Details notebook. On the **Concurrent** page, right-click the product and select the **Show Servers** option. The Details of Cluster window is displayed with additional information about the number of licenses being served by each server, as shown in Figure 67.

	Name	Status	Share	Target ID	Shared Licenses	Share in Use
	ip:coyote[119...	Serving	33	37dfc970	3334	0
	ip:rocky.rome...	Serving	33	2f456094	3333	0
	ip:sprint.rom...	Serving	33	37dfca1a	3333	0
	ip:lab15099.r...	Reserve	0	80a06a37	0	0

Servers: 4

Figure 67. Details of Servers Serving HAL Test Product

Enrolling and Removing Licenses on a Cluster

Checking the Clusters window, you can see the cluster ID of any cluster. Give the cluster ID to the software vendor, who can use it to create passwords that are bound to the cluster, rather than to an individual server. Enroll the licenses on only one server of the cluster, as shown in "Enrolling the Product" on page 103.

If you later decide to remove licenses enrolled on a cluster, make sure that all activated servers in the cluster be up and running at the time you do the removal.

Command-Line Equivalent

* To create a cluster named *Peanut* that has members **coyote**, **rocky**, **sprint**, and **lab15099**:

  ```
  i4blt -H c -N Peanut -T 4 -n "coyote rocky sprint lab15099"
  ```

* To activate the servers **rocky**, **sprint**, and **lab15099**:

  ```
  i4blt -H a -N Peanut -n rocky
  i4blt -H a -N Peanut -n sprint
  i4blt -H a -N Peanut -n lab15099
  ```

  The server **coyote**, which is the first in the list, is automatically activated.

* To add the server **thelma** to the cluster:

  ```
  i4blt -H a -N Peanut -n thelma
  ```

* To deactivate the server **thelma**:

  ```
  i4blt -H d -N Peanut -n thelma
  ```

- To get an overall report of cluster status:

    `i4blt -H s -N Peanut`

- To get a report of cluster status from the perspective of one of the activated servers in the cluster:

    `i4blt -H s -n coyote`

Other scenarios

Scenario 7: Upgrading a Custom Configuration

The scenario in this section shows you how to upgrade a custom configuration by adding a product to a current custom configuration.

To ensure that the products used are up-to-date, functionally suitable, and competitive, you will occasionally need to add new product components, increase the number of licenses, or extend the license period. To do this, you request from the vendor a new custom configuration password and supply the serial number of the current license. This serial number identifies your current custom configuration. You then pay for only the difference between the cost of the current license and that of the new license. Next, you install the upgraded license as shown in the following procedure.

Before you start the procedure, look at the current entries on the **Concurrent** page of the Details notebook. Double-click the product in the Basic License Tool main window, then select the **Concurrent** tab of the Details notebook. The Details page looks similar to that shown in Figure 68.

Figure 68. Initial State of the Concurrent Page for a Custom Configuration License

To upgrade your current license:

1. In the main window, select **Enroll** from the Products menu, and then **Single product...**. The Enroll Product window is displayed.

2. Select **Import**.

    The Import window is displayed. (See Figure 39 on page 103)

    In the **Look in** drop-down list, select the path to the directory in which the enrollment certificate is stored (in this example, `home/prince/certif`). Then, from **Files**, select the custom configuration enrollment certificate (in this example, **m2update.lic**), as shown in Figure 69.

3. Click **Open**.

The Enroll Product window is redisplayed, filled in with information from the enrollment certificate file, as shown in Figure 69.

Figure 69. Enroll Product Window for Custom Configuration

4. In the **Server name** field, select the server on which the initial key is installed. In this example, because the product is vendor-managed and has network licenses, the custom configuration license is enrolled on the network license server that has the target ID for which the license was created.

   Notes:
   a. If a specific target ID is set in the enrollment certificate, the licenses must be enrolled on the network license server of that target workstation.
   b. If the target ID in the enrollment certificate file is set to **ANY**, select a network license server.

   The license serial number is the same as it was for the previously enrolled license.

5. Click **OK**.

   The main window is displayed.The **Concurrent** page of the Details notebook now contains the updated license information. To display this page, double-click the product name, then select the **Concurrent** tab of the Details notebook. Scroll to the right to see the serial number and the annotation.

Command-Line Equivalent
To upgrade a custom configuration license, using the license certificate file **m2update.lic**, on server **lab15099**:

```
i4blt -a -f m2update.lic -n lab15099
```
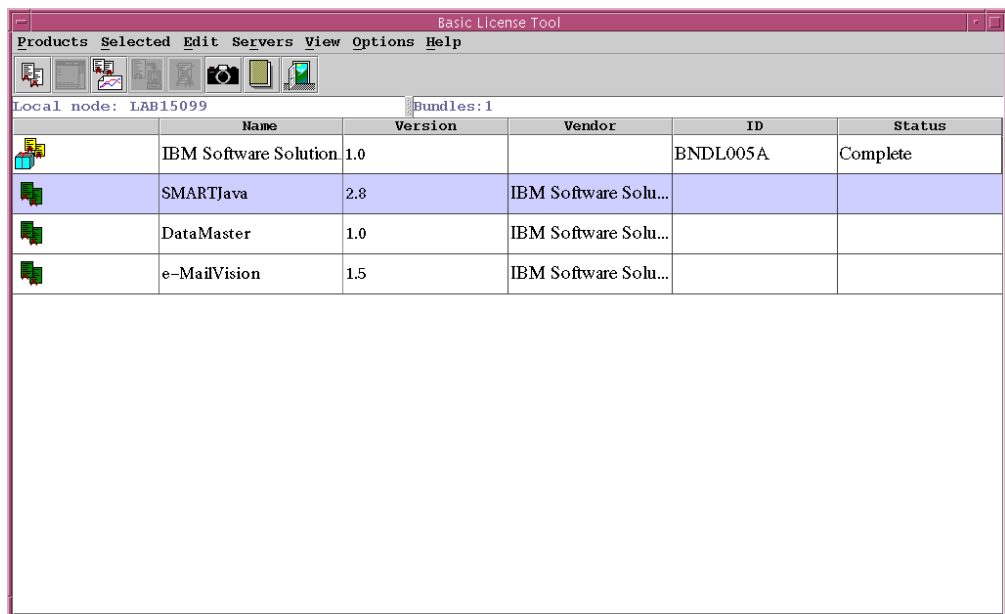
Scenario 8: Managing Bundles

In this scenario, the administrator manages licenses belonging to a bundle. This scenario shows you how to:

- Enroll multiple certificates belonging to a bundle
- Monitor bundles and licenses within them
- Manage groups of licenses belonging to a bundle
- Delete a bundle

To enroll multiple certificates, the administrator performs the following steps:

1. Select **Enroll** from the menu, and then **Multiple certificates...** The Import window is displayed. For an example of the Import window, see Figure 39 on page 103.
2. From the **File Name** list, select the enrollment certificates `smrtjava.lic`, `hicomtel.lic`, `mailvps.lic` and `datamst.lic`.
3. Click **OK** . The Enroll Product window is displayed for each certificate, filled in with information from the enrollment certificate file.
4. For each certificate, perform the following steps:
   a. In the **Server name** field, select the server on which licenses are to be installed.
   b. Click **OK**. The Enroll Product window is redisplayed, filled in with information about the next selected certificate (Figure 70).
5. Repeat step 4 for all certificates.

Figure 70. Bundle Enrolled

Once all certificates are enrolled, shift the Basic License Tool window to bundles view by selecting **Bundles** from the **View** pull-down menu. The Mechanical Products bundle, whose bundle ID is **BNDL005A**, is displayed.

Licenses inside the bundle can be shown by double-clicking on the bundle, or by selecting **Expand bundle** from the **Selected** pull-down menu. It is possible to view license details by double clicking on the license or by selecting **Open as details** from the **Selected** pull-down menu.

To hide licenses within a bundle, double-click the expanded bundle, or select **Collapse bundle** from the **Selected** pull-down menu.

To delete the IBM Software Solutions bundle, perform the following steps:

1. Select the bundle.
2. Choose **Remove bundle** from the **Selected** pull-down menu.

Command-Line Equivalent

To accomplish the same results (bundle enrollment, management, and deletion), use the following commands.

To enroll multiple certificates:

```
i4blt —a —f  "smrtjava.lic hicomtel.lic emailvps.lic datamst.lic"
```

To view bundles, with all licenses within them:

```
i4blt —lb
```

To delete the bundle:

```
i4blt —d —b BNDL005A
```

Scenario 9: Displaying Value Units

This scenario explains how to view the Value Units Policy outlined in Chapter 1 of this book. This policy allows products to be priced by introducing a common unit of a measure called Value Units.
This scenario shows you how to:

- View the information about the metrics used.
- Start the application and check the number of Value Units in use.

Viewing the Value Units

In the basic license tool main window select the line corresponding to the MailVision product in the main window. For information on how to enroll the product see "Enrolling the Product" on page 103, for information on how to distribute the licenses see "Distributing the Licenses" on page 105.

To view the Value Units, perform the following steps:

1. Select the **Open as Details** option from the **Selected** menu. The Details notebook is displayed.
2. Select the **Concurrent** tab. The Concurrent page is displayed.
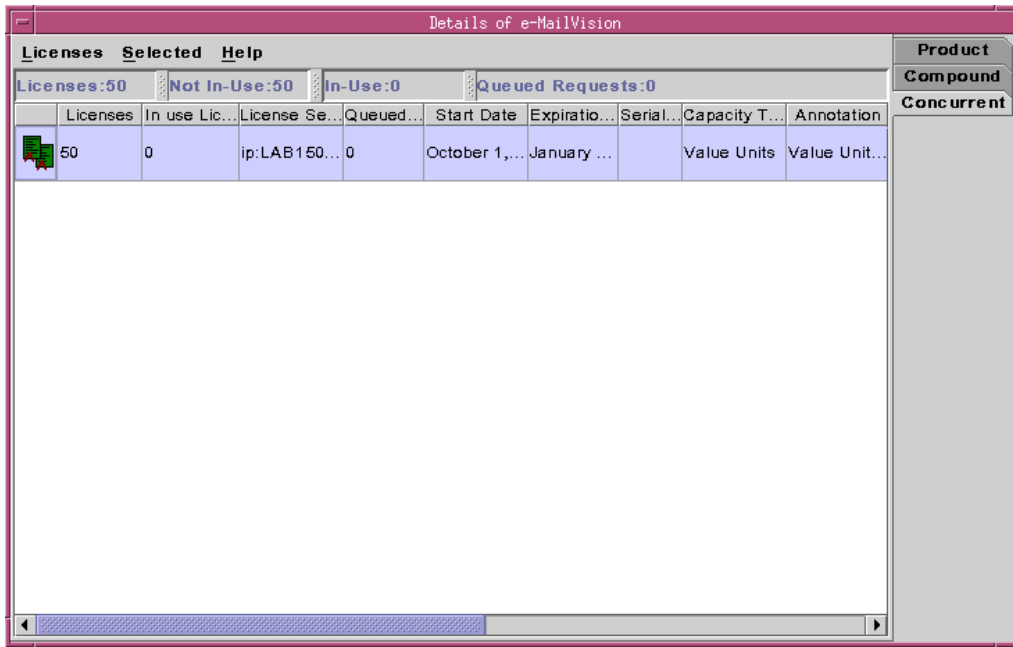
Figure 71. Concurrent Page

> In this page the number of licenses represents the Value Units available, while the capacity type is *Value Units* which indicates that this license uses the value units policy.
>
> 3. Select **Selected** from the menu bar and **Value Units details**. The Value Units Details window is displayed, as shown in Figure 72

Capacity type	Value Units	Value Units Calculat...	From	To
On-line Processors	12.0	Within Range	1	10
Physical Disks	1.5	Linear Increment		

Figure 72. Value Units Details Window

> This table describes the different system capacity measurements used to calculate the number of value units assigned for a capacity type.
>
> The **Capacity Type** column shows a list of items for which a capacity type has been assigned, such as:
>
> 103 Physical Disks
> 104 Physical Memory Mb
> 108 RISC 32–bit Processors
> 109 INTEL 64–bit Processors
> 110 RISC 64–bit Processors

For a complete list of possible values for capacity types, see the *-F* parameter for the .

The **Value Units** column shows the number of units assigned to a specific capacity type required for an application to start.

The **Value Units Calculation** column indicates the method by which the units are calculated: *Within Range* specifies the allowable range. For example, if the range is 4 to 28, 29 x 45 value units for RISC processors would be out of range. For *Linear Increment*, the number of needed value units is proportional to the number of licenses required. For example, if one Intel 64–bit processor is 6 value units, then two would be 12 value units.

Start the Application and Check the Number of In-use Value Units

While running the product the Concurrent Users page is added to the Details notebook for e-MailVision product, as shown in Figure 73. Check the number of Value Units in use: it is calculated as the sum of the different metrics used.
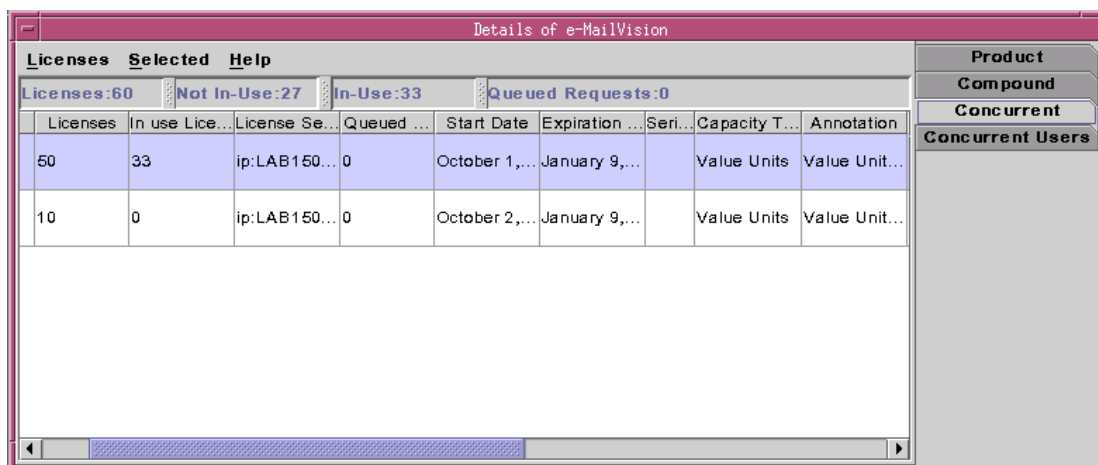
Figure 73. Concurrent Page with Value Units in use

Command-Line Equivalent

To accomplish the same results (viewing the annotation details, starting the application), use the following commands.

To display the annotation table:

```
To show the table of the metrics:
i4blt -ll -Z -p "EmailVision"
The following table is shown
---------------------- Value Units - Details -------------------------
  Capacity                   Value     Value Units
  Type                       Units     Calculation          from     to
------------------------    -------    -------------------  ------  -----
On-line Processors          12.00      Within Range         1        10
Physical Disks              1.50       Linear Increment
```

To show the number of Value Units in use:

```
I4blt -lp -p "EmailVision"
```

Scenario 10: Managing Concurrent-Offline Licenses

This scenario explains how to manage the use of concurrent-offline licenses, as outlined in "Concurrent-Offline Licenses" on page 8. Using this type of license, an administrator allows users to take a concurrent license for a certain number of days and to use it on a portable computer. This scenario shows you how to:

- Reduce the maximum number of days a product can be reserved.
- Modify the default authorization.
- Add a password for the default authorization.
- Create authorization records for a specific concurrent-offline license. This allows the authorized users to take the license.
- Modify a created authorization.
- Display details about the offline users.

In this scenario, 100 concurrent-offline licenses for DataMaster have already been enrolled. For information on how to enroll the product see "Enrolling the Product" on page 103. After enrollment, the product is displayed in the administration tool main window. Note that the icon indicates that the licenses are concurrent-offline.

To manage this kind of license:

1. Double-click the product in the administration tool main window, then click the **Concurrent-Offline** tab. The Concurrent-Offline Licenses - Details window is displayed (Figure 74). Note that the maximum offline period of the license is 120 days.
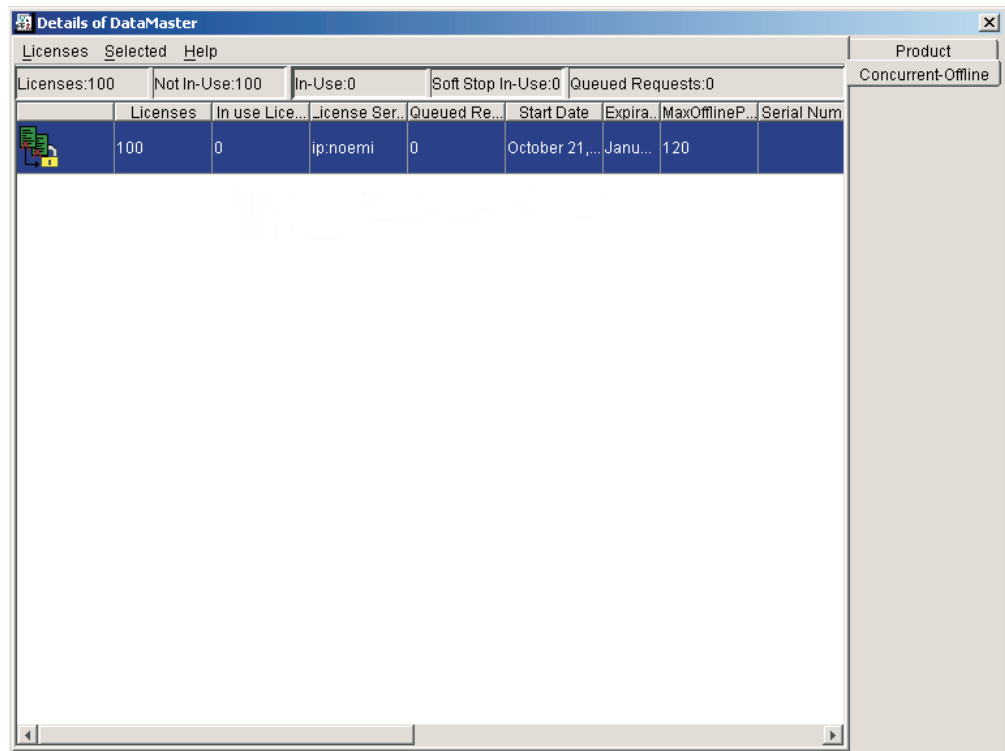
Figure 74. Concurrent-Offline Licenses - Details window

2. To reduce the maximum number of days a product can be reserved, from the Concurrent-Offline Licenses - Details window, right-click on the product and select **Licenses ▸ Set maximum offline period**. The Update Offline Period

window is displayed (Figure 75). You can modify the maximum number of days that a user can reserve a license. For example, you can change the value from 120 to 5 days. Click **OK** to save the new value and close the window.

Note: When the upgrade of the Concurrent-Offline License is enrolled, the value of the maximum offline period remains unchanged. If the upgraded license has a longer maximum offline period than the previous license, and the administrator wants to extend number of days a product can be reserved, the value of maximum offline period must be changed manually.
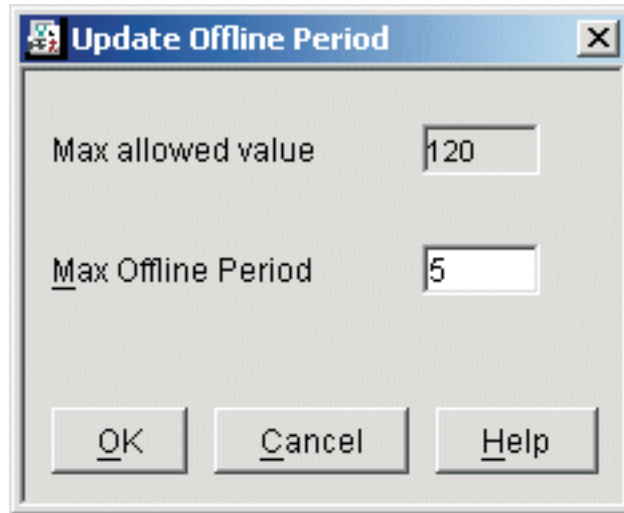
Figure 75. Update Offline Period window

3. To modify the default authorization, from the Concurrent-Offline Licenses - Details window (Figure 74 on page 132), right-click on the product and select **Selected ► Authorizations**. The Authorizations window is displayed (Figure 76 on page 134). The default authorization setting is for all concurrent licenses to be allowed.
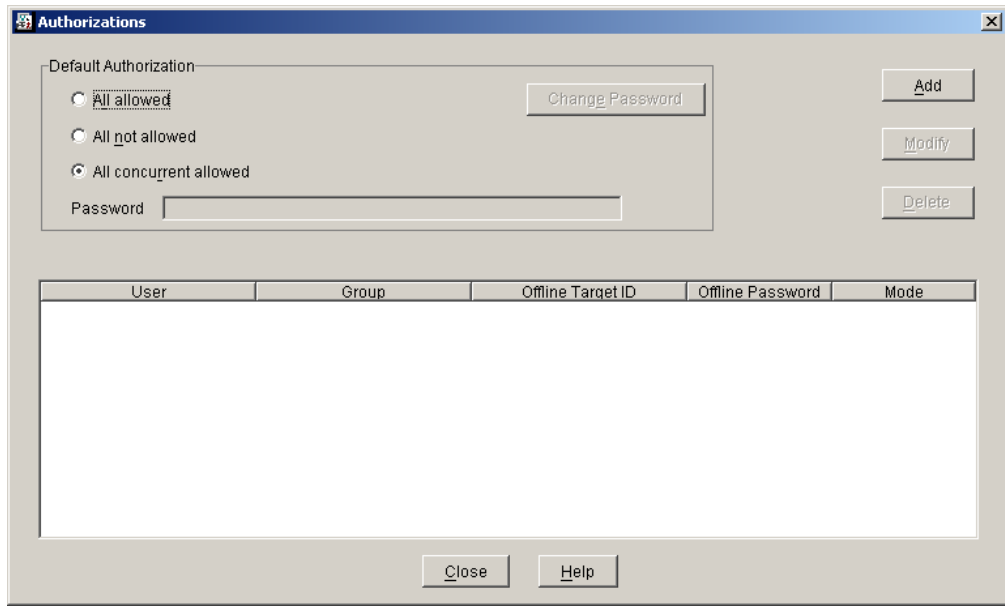
Figure 76. Authorizations window

For the selected license, the window shows the mode in which each user can use the license. There are three different modes for license use:

- **All allowed**. A user can start using the license as a concurrent license or can reserve it for nodelocked-offline use.
- **All not allowed**. The license cannot be granted in any mode. A user requires specific authorization to be granted to use the license.
- **All concurrent allowed**. A user can take the license as a concurrent license, but the license cannot be reserved in the nodelocked file.

4. To add a password to the default authorization, from the Concurrent-Offline Licenses - Details window (Figure 74 on page 132), right-click on the product and select **Selected ▶ Authorizations**. The Authorizations window is displayed (Figure 76). The default setting is **All concurrent allowed**, no password assigned, and for **Change Password** to be disabled. Select **All allowed**. **Change Password** is now enabled. Click **Change Password** and type in the new password. Click **Ok** to set the new password.

   Note: Group name field has length limited to 31 characters. Any name longer than 31 characters will be truncated. You cannot define a password for the **All not allowed** and **All concurrent allowed** authorizations. If you have set a password for the **All allowed** authorization, and you change the authorization to one of the other two settings, the password is deleted, and **Change Password** is disabled.

5. You decide that users can use the license only if they have a specific authorization, so you click **All not allowed**. Now create the first authorization record for a specific user:

   a. From the Authorizations window (Figure 76), click **Add**.

      The Add Authorization window is displayed (Figure 77 on page 135).

   b. Fill in the details for the new User, Group, and Offline Target ID (alone, or in any combination), and the Offline Password.

   c. Set the Mode for the use of the license for this specific user.

The modes are similar to the modes in the Authorization window (Figure 76 on page 134), except that they are for the individual user, group, offline target ID, or offline password that is being authorized.

d. Click **OK**.
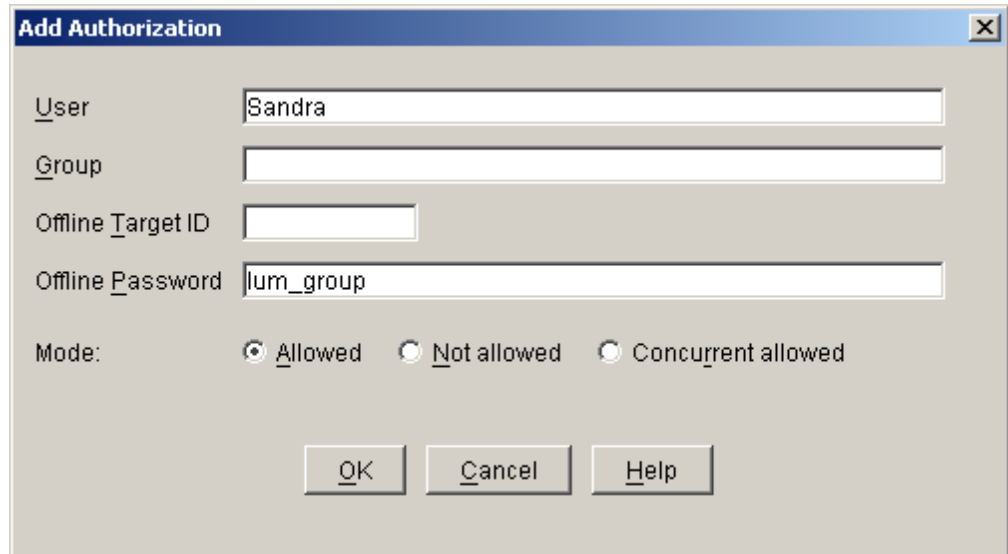
Figure 77. Add Authorization window

The authorization record has now been created (as shown in Figure 78).
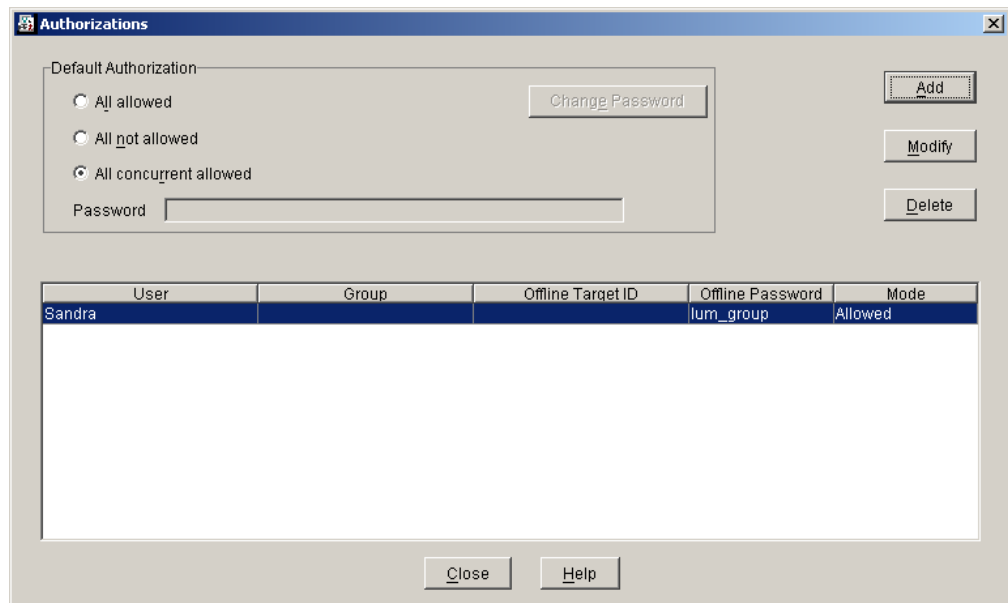
Figure 78. Authorizations window — showing new authorization

6. When you have created an authorization record, you can change the settings for the authorization record. Select the authorization record file in the Authorizations window (Figure 76 on page 134) and click on **Modify**. The Modify Authorization window is displayed. You decide to add the Offline Target ID of a specific machine (57f48e93 in this example) for the user Sandra. Click **OK**. The authorization record has now been modified (as shown in

Figure 79).
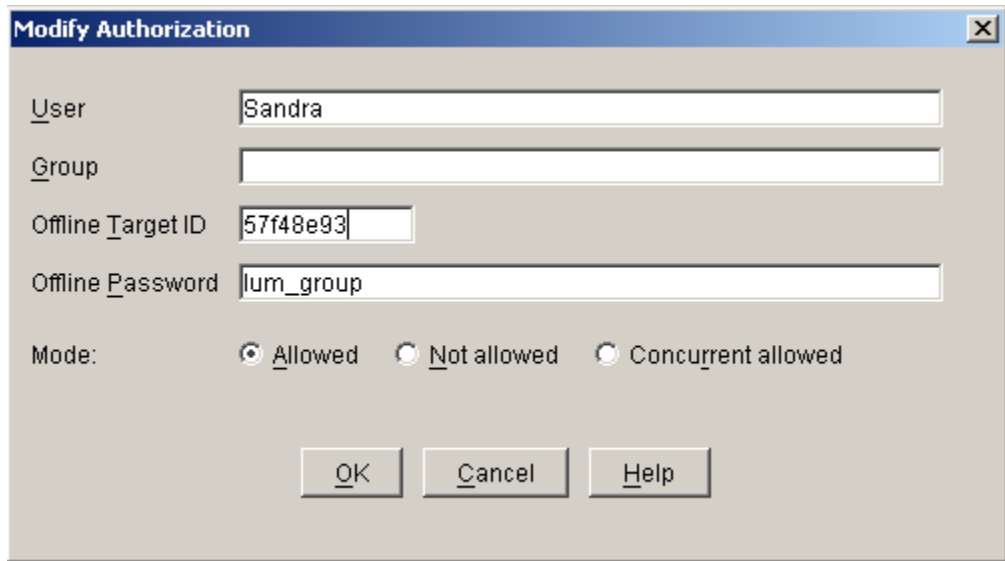
Figure 79. Modify Authorization window

7. User Sandra requires an offline-nodelocked license from her portable computer and reserves it for five days. In the administration tool, the display changes to show that there is a license in use (see Figure 80).
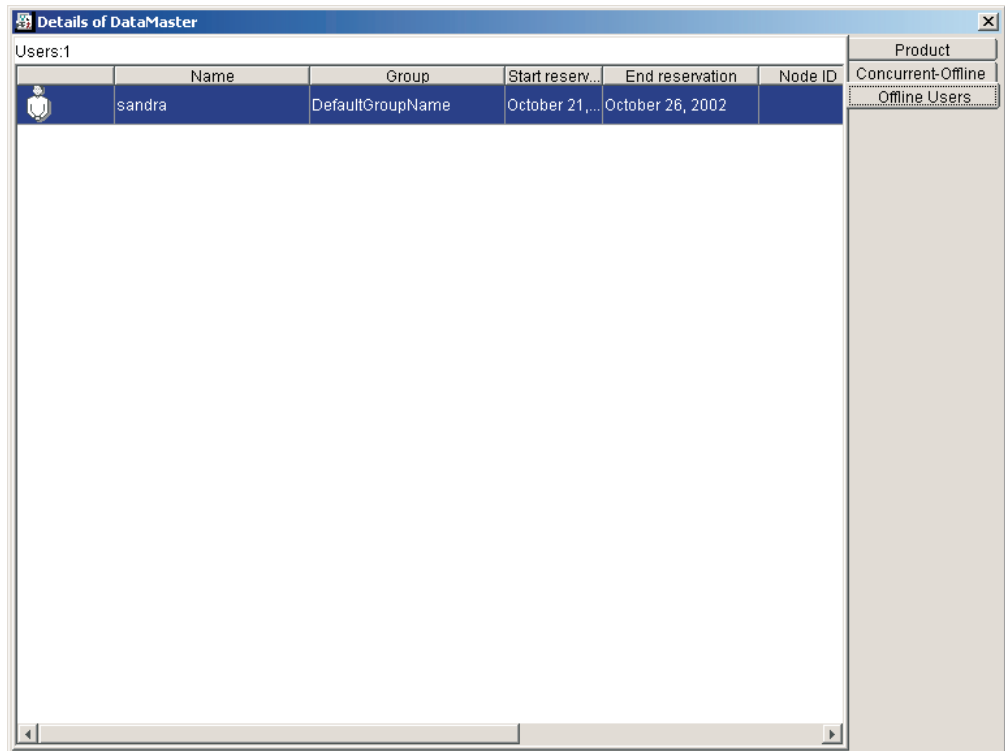
Figure 80. Administration tool, showing an in-use license

8. After three days, user Sandra decides that she no longer needs a license for her portable. Using the hand back procedure, she releases the license.

The administration tool displays that the license is no longer in use.

Note: If the user Sandra did not hand back the license, when the five-day reservation period expires, the license is handed back. The administration tool displays that the license is no longer in use.

Command-Line Equivalent

The equivalent CLI commands are as follows.

To update (to 50) the maximum number of days a product can be reserved.

```
i4blt -U -v "'IBM Corporation'" -p "DataMaster" –O 50
```

To modify the default authorization for DataMaster licenses:

```
i4blt -A m -v "'IBM Corporation'" -p "DataMaster 3.1" -t 899460562 -T no
-w "password"
```

To create a reservation record for the DataMaster license for the user Sandra, for any group and target ID:

```
i4blt -A a -v "'IBM Corporation'" -p "DataMaster 3.1" -T yes
-u "Sandra * *" –w "lum_group"
```

It is not possible modify using a unique command an authorization record other than the default record. The administrator needs to delete the old record and create a new record. To delete a reservation record for DataMaster licenses for the user Sandra:

```
i4blt -A d -v "'IBM Corporation'" -p "DataMaster 3.1"  -u "Sandra * *"
-t 899460562
```

To create a authorization record for the DataMaster license for the user Sandra for a specific machine with target id 57f48e93:

```
i4blt -A a -v "'IBM Corporation'" -p "DataMaster 3.1"   -T yes -u
"Sandra * 57f48e93" –w "lum_group"
```

To list all the authorization records for the DataMaster license3:

```
i4blt -A l -v "'IBM Corporation'" -p "DataMaster 3.1"
```

To display information about the users of concurrent-offline licenses for the DataMaster product:

```
i4blt -s –l co -v "'IBM Corporation'" -p "DataMaster"
```

Chapter 6. License Use Management Runtime Command Line Interface

This chapter describes how to use the License Use Management Runtime CLI.

In the descriptions of command syntax, the following conventions are used:

Bold type	Code this item exactly as shown.
Italic type	Replace this item with your own value.
Brackets []	The item or items in brackets are optional.
Braces {}	Choose one item from the list of alternative items inside the brackets.
Vertical line |	This OR sign separates the alternative items.
Three dots	An ellipsis following an item means you can include more than one of the item. For example, [*filename*]**...** specifies an optional list of files.

The following commands are available on *all* operating systems:

i4blt	Basic License Tool (page 140)
i4cfg	Configuration Tool (page 167)
i4tv	Test Verification Tool (page 173)
i4target	Target View Tool (page 174)
i4lmd	Network License Server (page 176)
i4llmd	Nodelocked License Server (page 178)
i4gdb	Central Registry (page 180)

The following commands are available on *both* UNIX and AIX:

lb_admin	Local Broker Administration (page 182)
drm_admin	GLBD Replicas Administration (page 186)
lb_find	GLBs List (page 189)
uuid_gen	ID Generator (page 190)
llbd	Local Location Broker Subsystem (page 191)
glbd	Global Location Broker Subsystem (page 192)
i4glbcd	Global Location Broker Data Cleaner Subsystem (page 194)

i4blt - Basic License Tool

Primary Command Options

ALL

The i4blt command has the following primary command options:

(No option)	Start the basic license tool GUI.
i4blt -a Enroll a Product	Add products to a license database.
i4blt -A Authorize	Authorize an offline-nodelocked license.
i4blt -C Clean Up Stale Licenses	Update the number of concurrent, reservable, per-server, and concurrent nodelocked licenses in use.
i4blt -d Delete a Product License or Bundle	Delete bundles, delete products from a license database, or delete Application Client Identifiers from the central registry.
i4blt -E Extract and Distribute Licenses	Extract and distribute licenses from a network compound password of a given product to the servers.
i4blt -h Display Help	Display command syntax and usage information about the basic license tool CLI.
i4blt -H Administer High-Availability Licensing	Create a cluster of network license servers; add servers to an existing cluster; display cluster status; activate and deactivate servers in a cluster.
i4blt -l Display a List	List license database information about servers, vendors, bundles, products, and licenses.
	Check the state of databases and repair them if needed.
	List all the databases and their states (provide the information if they are corrupted).
i4blt -m Monitor and Log Threshold Events	Monitor and log the threshold messages.
i4blt -r Generate a Report	Report on information recorded in the log file of a license server.
i4blt -R Reserve Licenses; Delete or Update Reserved Licenses	Reserve reservable licenses for use by a specific user, group, or node; deletes reserved licenses; updates reservation status.
i4blt -s Display Product License Status	Gather status information about license usage.
i4blt -S Set the Stress Threshold	Set the network license server load threshold.
i4blt -T Activate or Deactivate Tracing	Display information about current product usage on specified license servers.
i4blt -U Update a Product	Update the number of licenses you enrolled, update the hard stop/soft stop policy and high-water mark when enabled on the product, switch from per-server to per-seat licenses, set the threshold value of a customer-managed product, and set the maximum allowed period that a concurrent-offline license can be reserved.
i4blt -x Delete Server Log Entries	Delete license server and central registry log file entries.

General Rules for the i4blt command

1. The root user (UNIX), or the administrator (Windows), can run the i4blt command with all its options.

   Other users can run the **-l** (Display a List), **-s** (Display Product License Status), **-r** (Generate a Report), **-Hs** (check a cluster status), and **-h** (Display Help) options and the equivalent GUI actions, allowing them to display the content of the license and license users databases.

2. The parameters within any of the following name specifications are positional:
   - vendor_information (*vendor_name vendor_id vendor_password*)
   - product_information (*product_name product_version license_password license_annotation*)
   - administrator_information (*administrator_name company_name address additional_info*)
   - user_information (*user_id user_group user_node*)

3. All the following name specifications must be enclosed within double quotation marks (for example: *"vendor_name vendor_id vendor_password"*).
   - vendor_information (*vendor_name vendor_id vendor_password*)
   - product_information (*product_name product_version license_password license_annotation*)
   - administrator_information (*administrator_name company_name address additional_info*)
   - user_information (*user_id user_group user_node*)
   - signature_information

4. When a list of values (such as server names, vendor names, product names, or user names) is entered as a parameter, the list must be enclosed in double quotation marks. For example:

   ```
   i4blt -r3 -u "katie dustin emily adam"
   ```

5. A name that contains character spaces must additionally be enclosed within single quotation marks. If multiple blanks within the name must be preserved, each must be preceded by a backslash. For example:

   ```
   -v "'IBM Corporation'"
   ```
   ```
   -p "'Corel\ \ - System' 1.1"
   ```

6. The parameters you specify in any of the command options (for example, server names, vendor names, and product names) are case-sensitive.

7. The following command parameters have a limited length.

   group_name   31 characters

   > **Note:** License Use Management Runtime fix pack 12 supports Windows groups with names longer than 31 characters by truncating the group name to 31 characters; this ensures backward compatibility. If group names are different only after the 31st character they will be treated as the same group.

   server_name   31 characters

   > **Attention:** The first three character positions are reserved for the string `ip:` Any server name that is longer than 28 characters is truncated on the right.

   user_name   31 characters

8. You can display help on i4blt command options as follows:
   - To get help on just the -a, -U, -E, -d, -m or -H option:

```
i4blt -option
```
- To get help on just the -R, -l, -r, or -x option:
```
i4blt -optionh
```
Examples
- Display the i4blt -E syntax:
```
i4blt -E
```
- Display the i4blt -r syntax:
```
i4blt -rh
```

i4blt -a Enroll a Product

This option adds a product to the license database on the license server that you specify. Use the `i4blt -a` command to add a new product and its initial licenses to a license server database. You can also use this command to add licenses for existing vendor-managed products.

You can add product license information to a server in two ways:
- If you got the product license information in the form of an enrollment certificate file, you can install the product importing the enrollment certificate. You can enroll more than one product at the same time.
- If you got the product license information in a format other than an enrollment certificate file, you must enter the product information manually.

Note: You can only enroll licenses that have an associated additional license annotation file if you have the enrollment certificate file (that is, enroll using the **-f** option). In this case, make sure that the additional license annotation file is present in the directory indicated by the enrollment certificate file before you enroll the product (see the AdditionalLicenseData tag of the certificate).

Syntax: If you have the enrollment certificate file:

i4blt -a
[**-n** *server_name*] **-f** *filename* [*filename*]...
[**-R** *administrator_name* [*company_name address additional_info*]]
[**-T** *enrolled_licenses*]
[**-I** *signature_information*]

If you do not have the enrollment certificate file:

i4blt -a
[**-n** *server_name*]
-v *vendor_name vendor_id vendor_password*
-p *product_name product_version license_password* [*license_annotation*]
[**-R** *administrator_name* [*company_name address additional_info*]]
[**-T** *enrolled_licenses*]
[**-I** *signature_information*]
[**-S** { *serial_number* | *bundle_id* }]

Parameters:

-n *server_name*
> Specifies the name of the license server to which you intend to add the product. If **-n** is omitted:
> - If the product is customer-managed, and the licenses are network licenses, they are enrolled on the central registry.

- If the product is customer-managed, and the license is nodelocked, it is enrolled on the local workstation.
- If the product is vendor-managed, and the enrollment certificate file specifies a target ID, the licenses are enrolled on that workstation.
- If the product is vendor-managed, and the enrollment certificate file does not specify a target ID, the licenses are enrolled on the local workstation.

-f *filename [filename]...*
: (Only if you have the enrollment certificate) The complete path and file name of the enrollment certificate file containing the product license information that you intend to add. You can specify a list of file names if you want to enroll more than one certificate at a time.

-v *vendor_name vendor_id vendor_password*
: (If you do not have the enrollment certificate)

vendor_name
: The name of the vendor that manufactured the product that you intend to add.

vendor_id
: The unique vendor ID string for the vendor that you specify in the *vendor_name* argument.

vendor_password
: The unique vendor password string for the vendor that you specify in the vendor_name argument.

-p *product_name product_version license_password* **[** *license_annotation* **]**
: (If you do not have the enrollment certificate)

product_name
: The name of the product that you want to install.

product_version
: The version of the product that you specified in the product_name parameter.

license_password
: The unique license password string associated with the product.

license_annotation
: The license annotation information (if any) provided by the vendor.

-R *administrator_name* **[** *company_name address additional_info* **]**
: (for customer-managed use products only)

administrator_name
: The name of the administrator who performs the operation. This parameter is required.

company_name
: The name of your company.

address The address of your company.

additional_info
: Comments, notices to future users, or other information about the initial enrollment of this product.

If you enroll several products, the value you specify for these options apply to all the license certificates.

-T *enrolled_licenses*
> (for customer-managed use products only) The number of licenses you have acquired from the software supplier. This parameter is required.
>
> If you enroll several products, the value you specify for this option applies to all the license certificates.

-I *signature_information*
> (for customer-managed use products only) Information about the user issuing the command, to be stored with the signature stamp.

-S { *serial_number* | *bundle_id* }
> The serial number of a custom configuration license, or the bundle identifier for a bundle. The number is a string of up to 31 alphanumeric characters that uniquely identifies a custom configuration or a bundle.

Examples: Add a customer-managed use product:

```
i4blt -a
-v "Venus 4ca0fd5cf000.0d.00.02.1a.9a.00.00.00 kz5esmu69hzyw"
-p "timer 1.1 wzx3ewdfrvu4v64d53bbrkzhheaaaaa"
-R "Alex IBM Rome" -T 100
-I "'Alex using root userid'"
```

Add a vendor-managed use product:

```
i4blt -a -n thelma
-p "scena 1.0 suf0fpeixfi5v78a22xxrkzhheaaaaa"
-v "Operatix 7gp4ac8jj000.0d.00.02.1a.9a.00.00.00 lb7usud93jdna"
```

i4blt -A Authorize

Use this option to create an authorization for either a concurrent, or a concurrent-offline license. i4blt must be run on the same machine where the license is enrolled.

Note: To add an authorization for a high-availability license, you must repeat the command on each license server in the cluster.

Syntax:

i4blt -A *action_type*

{ { **a -v** *"vendor_name"* **-p** *"product_name product_version"*
-u *"user_name user_group target ID"* [**-w** *"password"*] **-T** *mode* [**-t** *timestamp*]}
|
  { **l -v** *"vendor_name"* **-p** *"product_name product_version"* [**-t** *timestamp*]}
|
{ **d -v** *"vendor_name"* **-p** *"product_name product_version"*
-u *"user_name user_group target ID"* **-t** *timestamp*}
|
{ **m -v** *"vendor_name"* **-p** *"product_name product_version"* **-t** *timestamp*
-T *mode* [**-w** *"password"*] } }

Parameters:

action_type
> Specifies the action to be taken:
>
> **a**   Adds an authorization for a product. With action type **a**, the **-v**, **-p**, **-u**, and **-T** parameters are all required. The **-t** and **-w** parameters are optional.

> To add an authorization for a high-availability license, you must repeat the command on each license server in the cluster.

l Lists the authorizations for a product. With action type **l**, the **-v** and **-p** parameters are required. The **-t** parameter is optional.

d To delete an authorization for a product. With action type **d**, the **-v**, **-p**, **-u**, and **-t** parameters are all required.

> To delete an authorization for a high-availability license, you must repeat the command on each license server in the cluster.

m To modify an authorization for a product. With action type **m**, the **-v**, **-p**, **-t**, and **-T** parameters are all required. The **-w** parameter is optional.

> You can modify only the default authorization from the CLI. To modify any authorization that is already created, you must use the GUI.

-u *"user_name user_group target ID"*
> The user name and user group for the user to be authorized, and target ID of the machine where the license is to be authorized. The target ID applies only to concurrent-offline licenses that are to be used in the offline mode. The target ID is ignored for concurrent licenses, and also for concurrent-offline licenses that are used as concurrent licenses.

> The wildcard character (*) can also be used to denote any.

-w *"password"*
> The initial password for the authorized user. This is for concurrent-offline licenses only. The password cannot contain spaces. The password must be enclosed in double quotation marks. It is for concurrent-offline licenses that are to be used for offline mode. The password is ignored for concurrent licenses, and also for concurrent-offline licenses that are used as concurrent licenses.

> The wildcard character (*) can be used to clear (unset) the password.

-v *"vendor_name"*
> The vendor of the licensed product that you intend to update. Enclose the *vendor_name* with single quotation marks (') when the *vendor_name* contains spaces, for example `"'IBM Corporation'"`.

-p *"product_name product_version"*
> The information on the licensed product that you intend to update.

> *'product_name'*
> > The name of the product for which you have acquired the new licenses. Enclose the *product_name* with single quotation marks (') when the *product_name* contains spaces.

> *product_version*
> > The version of the product that you specified in the product_name parameter.

-T *mode*
> Set to one of:

yes A user can start using the license as a concurrent license or can reserve it for nodelocked-offline use.

no The license cannot be granted in any mode.

> **concurrent**
>> This is for concurrent-offline licenses only. A user can start using the license in a concurrent mode only. The user cannot reserve the license for nodelocked-offline use.

> **-t** *timestamp*
>> Unique timestamp of the product license that you intend to create, modify, list, or delete. To get the timestamp, issue the following command:
>>
>> ```
>> i4blt -lp -p "product_info" -i
>> ```

Examples: To create an authorization record for the DataMaster license for the user Sandra, for any group and target ID:

```
i4blt -A a -v "'IBM Corporation'" -p "DataMaster 3.1" -T yes
-u "Sandra * *" —w "lum_group"
```

To create an authorization record for the DataMaster license for the user Sandra for a specific machine with target id 57f48e93:

```
i4blt -A a -v "'IBM Corporation'" -p "DataMaster 3.1" -T yes
-u "Sandra * 57f48e93" —w "lum_group"
```

i4blt -C Clean Up Stale Licenses

Use **i4blt -C** to update the number of in-use concurrent, reservable, per-server, and concurrent nodelocked licenses.

When you issue this command, License Use Management Runtime polls all the license servers that have granted licenses of these types and verifies that the licenses are still in use. If any stale licenses are found, they are removed from the number of in use licenses.

Syntax:

i4blt -C
[**-F** *server_type* { **l** | **w** | **a** }]
[**-n** *server_names*]
[**-v** *vendor_names*]
[**-p** *product_names*]

Parameters:

-F *server_type*
> A filter on the type of server to be searched. Specify **l** for nodelocked license servers, **w** for network license servers, or **a** (the default) for both network license servers and nodelocked license servers.

-n *server_names*
> The names of the servers where the products are enrolled.

-v *vendor_names*
> The name of the vendors that manufactured the products whose licenses are in use.

-p *product_names*
> The names of the products whose stale licenses you want to clean up.

Examples: Clean up stale licenses for the Graphics product of vendor Alpha on servers **Thelma**, **Hall**, and **Louise**:

```
i4blt -C -n "Thelma Hall Louise" -v "Alpha" -p "Graphics"
```

Clean up stale licenses for the Graphics product of vendor Alpha on all nodelocked license servers in the network.

```
i4blt -C -F l -v "Alpha" -p "Graphics"
```

i4blt -d Delete a Product License or Bundle

This option deletes a product license or bundle from the license database on the license server that you specify, or an Application Client Identifier from the central registry.

Syntax:

i4blt -d
-n *server_name*
-v *vendor_name*
-p *product_name product_version*
{ **-t** *timestamp* | **-A** *ACID* | **-b** *bundle_id* }
[**-I** *signature_information*]

Parameters:

-n *server_name*
> Either of the following:
> * The name of the license server from which you intend to delete the product license.
> * To delete a high-availability license, the name of one of the servers in the cluster on which the license is enrolled.
>
> To delete a high-availability license, issue the command:
> ```
> i4blt -d -n server_name -v vendor_name -p product_name
> product_version -t timestamp
> ```
> where server_name identifies one of the servers in the cluster on which the license is enrolled. You must repeat the command for each license server in the cluster.

-v *vendor_name*
> Name of the vendor whose product license you intend to delete.

-p *product_name product_version*
> The information on the licensed product whose licenses you intend to delete.
>
> *product_name*
> > Name of the product whose license you intend to delete.
>
> *product_version*
> > Version of the product whose license you intend to delete.

-t *timestamp*
> Unique timestamp of the product license that you intend to delete. To get the timestamp, issue the following command:
> ```
> i4blt -lp -p "product_info" -i
> ```
> Do not specify the timestamp when you delete an Application Client Identifier.

-A *ACID*
> Unique identifier of the Application Client Identifier of an application client you want to delete from the central registry. After deletion the

application client no longer has the license to use the specified product. To get the Application Client Identifier, issue the following command:

```
i4blt -s -lpt -v "vendor_name" -p "product_info"
```

Do not specify **-A** when you delete a product.

-b *bundle_id*
> The identifier of the bundle you want to delete.

-I *signature_information*
> (for customer-managed use products only) Information about the user issuing the command, to be stored with the signature stamp. Use this parameter when deleting a product license.

When the last license for the only remaining product of a vendor is deleted, the vendor is automatically deleted from the license database. Vendor-managed compound passwords and use-once licenses cannot be deleted until they expire.

Examples: Delete a bundle whose ID is AZ57561C:

```
i4blt -d -b AZ57561C
```

Delete an expired license for a VectorComp Corporation product called EZ-Vectors Version 1.0. The unique timestamp of the license to be deleted from the database on server **saturn** is 781401788:

```
i4blt -d -n saturn -v "'VectorComp Corporation'" -p
"EZ-Vectors 1.0" -t 781401788
```

Delete an application client whose Application Client Identifier is **thelma** from the central registry. After this command the application client will no longer have licenses for the product EZ-Vectors Version 1.0 of VectorComp Corporation vendor:

```
i4blt -d -v "'VectorComp Corporation'" -p "EZ-Vectors 1.0"
-A thelma
```

i4blt -E Extract and Distribute Licenses

Use the i4blt -E command to extract licenses from an installed network compound password and distribute them to the network license servers.

Syntax:

i4blt -E
-n *origin_server_name*
-v *vendor_name*
-p *product_name product_version*
-A *license_number_per_server*
-w *target_server_names*
[**-I** *signature_information*]

Parameters:

-n *origin_server_name*
> The name of the server where the network compound password is enrolled.

-v *vendor_name*
> The name of the vendor that manufactured the product whose licenses you want to distribute.

-p *product_name product_version*

> The information on the licensed product whose licenses you intend to distribute.

> *product_name*
>> The name of the product whose licenses you want to distribute.

> *product_version*
>> The version of the product that you specified in the product_name parameter.

-A *license_number_per_server*

> The number of licenses for the specified product you want to distribute on each of the servers specified after the -w parameter.

-w *target_server_names*

> The servers on which you want to distribute the licenses.

-I *signature_information*

> (For customer-managed use products only) Information about the user issuing the command, to be stored with the signature stamp.

Examples: Extract and distribute 10 licenses to each of the servers Louise and Hall, for the Test Compiler product, Version 1.1 of vendor Psychosync, installed on server Thelma:

```
i4blt -E -n "Thelma" -v "Psychosync" -p "'Test Compiler' 1.1"
-A 10 -w "Louise Hall" -I "'Paula'"
```

i4blt -h Display Help

This option displays general syntax information for the basic license tool CLI.

Syntax:

i4blt -h

Examples: Display the i4blt syntax:

```
i4blt -h
```

i4blt -H Administer High-Availability Licensing

This option creates a cluster of network license servers; adds servers to an existing cluster; displays cluster status; and activates and deactivates servers in a cluster.

Syntax:

i4blt -H *action_type*
{ { **c -N** *cluster_name* **-n** *"server_names"* **-T** *initial_number_of_servers* }
|
{ **a** | **d** } **-N** *cluster_name* **-n** *"server_names"* }
|
{ **s** { **-N** *cluster_name* | **-n** *"server_names"* } } }

Parameters:

action_type

> Specifies the action to be taken:

> **c**   To create a cluster. With action type **c**, the **-N**, **-T**, and **-n** parameters are all required.

a   To add a server to a cluster, or to activate a server in a cluster. With action type **a**, the **-N** and **-n** parameters are required.

d   To deactivate a server in a cluster. With action type **d**, the **-N** and **-n** parameters are required.

s   To request cluster status. With action type **s**, the **-N** or **-n** parameter is required. Use the **-N** parameter to get an overall view of cluster status. If you are having problems with cluster operation and you want to see a view of the cluster from the perspective of an individual server, use action type **s** with the **-n** parameter. In either case, the cluster status display includes the following information about the cluster:

- Cluster name
- Cluster ID
- Cluster status:

    Active   Running, serving licenses

    Change Pending

          Waiting for a change in the status of a server, or of the cluster, to be propagated to all the servers

    Inactive   Not enough servers up and running

    Incomplete   Not enough servers activated

- Initial number of servers
- Minimum number of members, maximum number of members, and minimum up and running for the cluster to work
- For each server in the cluster:
    - Server name
    - Server status:

        Serving   Running, serving licenses

        Waiting   Server is ready, but cluster is in incomplete or inactive state

        Unavailable   Not started or not reachable

        Reserve   In reserve in case a serving server becomes unavailable

        Not activated   Defined as a member of the cluster but administrator has not yet activated the server or has deactivated the server

    - Percentage of licenses being served
    - Target ID

-N *cluster_name*
          The name of the cluster to which the command is directed. The **-N** parameter is required if action type is **c**, **a**, or **d**, and it is recommended when action type is **s**.

-T *initial_number_of_servers*
          The initial number of servers in the cluster that you are creating. The **-T** parameter is required if action type is **c**, and is not valid if action type is not **c**. The **-n** parameter must specify a number of servers equal to the value of **-T**.

-n *server_names*

> The names of the servers to which the command is directed. The **-n** parameter is required if action type is **c**, **a**, or **d**, and it can be used when action type is **s**. If action type is **c**, this is the list of initial members of the cluster. You must specify a number of servers equal to the value of **-T**. After this command has been processed, the first server in the list is automatically activated. Issue `i4blt -H` again, using action type **a**, to activate each additional server. At the time the `i4blt -H c` command is processed, all the servers specified must be up and running. If not, the command fails. If action type is **a** or **d**, this is the name of the server to be activated or deactivated. You must specify exactly one server. If action type is **a**, the server must be up and running when the command is processed. If not, the command fails. If action type is **s**, this is the name of any server that is currently activated in the cluster. The command returns cluster status from the perspective of this server.

> **Note:** When the cluster is in **Change pending** status, different servers may return different data.

Examples: Create a cluster named **ruth** that has three members: anthony, germaine, and costanza:

```
i4blt -H c -N ruth -T 3 -n "anthony germaine costanza"
```

Activate the servers **germaine** and **costanza**:

```
i4blt -H a -N ruth -n germaine
i4blt -H a -N ruth -n costanza
```

Add the server **sandra** to the cluster:

```
i4blt -H a -N ruth -n sandra
```

Deactivate the server **germaine**:

```
i4blt -H d -N ruth -n germaine
```

Get an overall report of the status of the cluster *nobel*:

```
i4blt -H s -N nobel
```

Get a report of the status of the cluster *nobel* from the perspective of one of its activated members, **pirandello** (recommended only for troubleshooting purposes):

```
i4blt -H s -n pirandello
```

i4blt -l Display a List

All users can use this option to display a list of servers, vendors, bundles, products, or licenses. You can also use it to display details about individual products or individual licenses.

Syntax:

i4blt -l *list_type* { **n** | **s** | **v** | **p** [**-i**] | **b** | **l** [**-k**] [**-Z**] **d** | **r** }
[**-b** *bundle_id*]
[**-F** *server_type* { **l** | **w** | **a** }]
[**-n** *server_names*]
[**-v** *vendor_names*]
[**-p** { *product_name*[*product_version*] } ...]
[**-u** *user_names*]

i4blt - Basic License Tool

Parameters:

list_type

Indicates the type of information that you want to list. You can specify any one of the following list types:

-ln or -ls

To display a list of active license servers. Also displays information about trace level, load percentage and load threshold percentage. The load percentage and load threshold percentage are not applicable for a nodelocked license server or for a central registry license server or for any server older than License Use Management Runtime Version 4.6.8.

Filters:
- You can use the **-F** filter option to display a list of active network license servers or nodelocked license servers.
- Do not specify the **–b**, **-n**, **-v**, **-p**, or **-u** filter option with this parameter.

-lb To display a list of bundles.

Filters:
- To list only the contents of a specified bundle, use the **–b** filter option, followed by the bundle identifier.
- To list bundles or bundle components for a specific type of license server, use the **-F** filter option to specify nodelocked license servers or network license servers.
- To list bundles or bundle components for servers that you specify, use the **-n** filter option followed by one or more server names.
- To list bundles or bundle components from particular vendors, use the **-v** filter option, followed by one or more vendor names.
- Do not specify the **-p**, **-u**, or **-t** filter option with this parameter.

-lv To create a vendor list.

Filters:
- To list vendor information gathered from a specific type of license server, use the **-F** filter option to specify nodelocked license servers or network license servers.
- To list vendor information gathered from servers that you specify, use the **-n** filter option followed by one or more server names.
- Do not specify the **-b**, **-v**, **-p**, or **-u** filter option with this parameter.

-lp [-i] To create a product list.

Filters:
- To list product information gathered from a specific type of license server, use the **-F** filter option to specify nodelocked license servers or network license servers.
- To list product information gathered from servers that you specify, use the **-n** filter option followed by one or more server names.

- To list product information on products from particular vendors, use the **-v** filter option, followed by one or more vendor names.
- To list product information on particular products, use the **-p** filter option, followed by one or more product names.
- To list information on users who are currently using the products that you specify, use the **-u** filter option, followed by one or more user names.
- Do not specify the **-b** filter option with this parameter.
- Specify the **-k** option to remove expired licenses from the list.

Specify the **-i** option to display detailed information about each product (for example, the number of reserved and unreserved licenses) in a product list.

i4blt -ld

To display a list of databases and their states (provides information about the databases that are corrupted).

i4blt -lr

To display a list and repair databases. However, database corruption occurs very rarely during day-to-day operation and this command should be used sparingly.

-ll [-k][-Z]

To create a list of individual licenses. The output includes all the information you get by specifying **lp** with the **-i** option, plus, for products with concurrent licenses that are administered in a high-availability environment, information about the cluster and servers within the cluster.

Filters:
- To list only the contents of a specified bundle, use the **–b** filter option, followed by the bundle identifier.
- To list license information gathered from a specific type of license server, use the **-F** filter option to specify nodelocked license servers or network license servers.
- To list license information gathered from servers that you specify, use the **-n** filter option followed by one or more server names.
- To list license information on products from particular vendors, use the **-v** filter option, followed by one or more vendor names.
- To list license information on particular products, use the **-p** filter option, followed by one or more product names.
- To list information on users who are currently using the licenses that you specify, use the **-u** filter option, followed by one or more user names.
- Specify the **-k** option to remove expired licenses from the list.
- Specify the **-Z** option to display a table containing value units annotation information.

Value Units Output:
- **Capacity Type**. Shows a list of items for which a capacity type (metric) has been assigned.
- **Value Units**. Shows the number of value units assigned to a specific capacity type.

- **Value Units Calculation**. Indicates the method by which the value units are calculated from capacity units:
  - *Within Range*. Specifies the same value units for all capacity units in the specified range.
  - *Linear Increment*. The number of value units required is proportional to the number of capacity units requested.
- **From**. Indicates the beginning of the range.
- **To**. Indicates the end of the range (if any).

High-Availability Output:
- Cluster name
- **For each server in the cluster**:
  - Server name
  - Server status:

    Serving
    > Running, serving licenses

    Waiting
    > Server is ready, but the cluster is in incomplete or inactive state

    Unavailable
    > Server is unavailable to other members of the cluster

    Reserve
    > In reserve in case a serving server becomes unavailable

    Not activated
    > Defined as a member of the cluster but the administrator has not yet activated the server or has deactivated the server

  - Percentage of licenses being served by this server
  - Target ID
  - Number of licenses served by this server
  - Number of in-use licenses served by this server

-F *server_type*
> A filter on the type of server to be searched. Specify **l** for nodelocked license servers, **w** for network license servers, or **a** (the default) for both network license servers and nodelocked license servers.

-n *server_names*
> Names of the servers about which you want to display information in a vendor or product list.

-v *vendor_names*
> Names of the vendors about whose products you want to display information in a product list.

-p { *product_name*[**?***product_version*] } ...
> Names of the products and, optionally, their versions about which you want to display information in a product list. If *product_name* or *product_version* contains blank characters, enclose the product name and product version in quotation marks, for example:
>
> ```
> -p 'product name?product_version'
> ```

-b *bundle_id*

To list the contents of a specific bundle, use the **–b** filter option, followed by the bundle identifier.

-u *user_names*

Names of users about whom you want to display license usage information in a product list.

-i   Specify the **-i** option in conjunction with a product list (**i4blt -lp**) to display the following detailed license usage information about an individual licensed product in a product list:

- Vendor name
- Vendor ID
- Product name
- Product version
- Product ID
- Licenses (total on all the selected servers)
- In-use licenses (total on all the selected servers)
- **For each license instance:**
  - Number of licenses
  - License type
  - Capacity type
  - Bundle component
  - Server on which the license is installed
  - License annotation (if any)
  - Serial number or bundle ID (if any)
  - Start date
  - Expiration date
  - Time stamp
  - Password use control level
- **For products with customer-managed use control and per-server, per-seat, or concurrent nodelocked licenses**, the following information is also displayed:
  - High-water mark licenses
  - Threshold value
  - Soft stop
  - Soft stop enabled
- **For products with customer-managed use control and use-once nodelocked licenses**, the following information is also displayed:
  - Threshold
- **For per-seat licenses**, the following information is also displayed:
  - Enablement flag
- **For reservable licenses**, the following information is also displayed:
  - Number of reserved licenses
  - Number of unreserved licenses
- **For reserved licenses**, the following information is also displayed:
  - User for whom licenses are reserved
  - Group for which licenses are reserved

       – Node for which licenses are reserved
- **For concurrent, concurrent nodelocked, and per-server licenses**, the following information is also displayed:
    – Multiuse rules (if any)

    For try-and-buy licenses, the following information is also displayed:
    – Try-and-buy flag
- **For compound passwords**, the following information is also displayed:
    – Derived license type
    – Aggregate duration
    – Derived start type
    – Derived expiration date
    – **For products with customer-managed use control and concurrent or reservable licenses**, the following information is also displayed:
        - Enrolled licenses
        - Distributed licenses
        - To be distributed licenses
        - High-water mark licenses
        - Threshold value
        - Soft stop
        - Soft stop enabled
    – **For products with customer-managed use control and use-once licenses**, the following information is also displayed:
        - Enrolled licenses
        - Distributed licenses
        - To be distributed licenses
        - Threshold value

Examples: List all servers:

```
i4blt -ln
```

List all vendors on all servers:

```
i4blt -lv
```

List all vendors on all network license servers:

```
i4blt -lv -F w
```

List all vendors on server **Hall**:

```
i4blt -lv -n Hall
```

List all products on server **Hall**:

```
i4blt -lp -n Hall
```

List all products on server mercury provided by vendors Opticon, Inc. and Cybertronics Ltd.:

```
i4blt -lp -n mercury -v "'Opticon, Inc.' 'Cybertronics Ltd.'"
```

List detailed information for the product PsychoSynch on server **venus**:

```
i4blt -lp -n venus -p PsychoSynch -i
```

List information for the 1.2 version only of the product PsychoSynch and any
version of the Megamail product:

```
i4blt -lp -p"PsychoSynch?1.2 Megamail"
```

List information for the 1.2 version only of the product Sync Psycho and any
version of the Megamail product:

```
i4blt -lp -p"'Sync Psycho?1.2' Megamail"
```

List all of the products on the server **neptune** that are currently being used by the
user **Alex**:

```
i4blt -lp -n "neptune" -u Alex
```

i4blt -m Monitor and Log Threshold Events

This option displays the threshold messages and logs them if the threshold logging
option is specified.

Syntax:

i4blt -m
[**-T** *percentage* [**1...100**]]
[**-A** *periodic_mode* [**yes** | **no**]]
[**-X** *frequency* [**1...1440**]]
[**-l** *log* [**yes** | **no**]]

Parameters:

-T *percentage*

> Specifies the level of threshold value. It can be any number between 1 and
> 100. The default is 80. This is the percentage over which you want to log
> the level of usage of each product installed on all the servers. For instance,
> if you have 100 licenses of the product Icon Editor Version 1.5, and you set
> the level of threshold to 10, a message appears in the vendor messages
> report only if more than 10 licenses are in use. If 20 licenses are being
> used, the message will say:
>
> ```
> The 20% of licenses of Icon Editor 1.5 is in use.
> ```
>
> Note that a customer-managed product may have its own threshold value,
> set with the **i4blt -U** command. Such a threshold value overrides the **-T**
> value.

-A *periodic_mode*

> Specifies whether to check the license usage of the products only once, or
> periodically. Its values can be:
> - no To check the threshold conditions on the products once, immediately.
>   If **-A** is omitted, this is the default.
> - yes To check the threshold conditions on the products periodically, with
>   the frequency specified with the **-X** parameter.

-X *frequency*

> Specifies the number of minutes between one license usage check and the
> next. Enter a value between 1 and 1440. It is mandatory if you set the **A**
> parameter to **yes**.

-l *log*   Specifies whether or not the threshold messages must be logged on the
> license server to be reviewed with the report function.

Examples: Set the threshold percentage to 50% and set the check on the products' usage to every 4 hours:

```
i4blt -m -T 50 -A yes -X 240
```

i4blt -r Generate a Report

All users can use this option to list server, event, vendor, product, and user information on specified license servers.

Syntax:

i4blt -r *report_type* [**1** | **2** | **3** | **4** | **5** | **6**]
[**-b** *start_date*]
[**-g** *end_date*]
[**-e** *event_filter*]
[**-F** *server_type* { **l** | **w** | **a** }]
[**-n** *server_names*]
[**-v** *vendor_names*]
[**-p** { *product_name*[?*product_version*] } ...]
[**-u** *user_names*]

Parameters:

-r *report_type*

Specifies the type of report to generate. The following report types are available:

1 - Standard Event Report

Displays detailed information about significant events occurring on the license servers that you specify. Available for all license types.

2 - License Request by Product Report

Displays statistical information about the use of the licenses of a product, on the license server that you specify, for the time interval you specify. For each product, it reports the licenses requested, the licenses granted, and the percentage of rejections. Not available for simple nodelocked or use-once (nodelocked or network) licenses. You can specify only one server for each report.

3 - License Request by User Report

Displays statistical information about the use of products by users, on the license server that you specify, in the time interval you specify. For each user, it reports the licenses requested, the licenses granted, and the percentage of rejections for each product the person is using. Not available for simple nodelocked or use-once (nodelocked or network) licenses. You can specify only one server for each report.

4 - License Use by Product Report

Displays statistical information about the use of the licenses of a product, on the license server that you specify, in the time interval you specify. For each product, it lists the maximum number of concurrent nodes that used the product, the maximum number of concurrent users, and the average time of use of the product. Statistical information about events appears after the license release. Not available for per-seat, simple nodelocked, or use-once (nodelocked or network) licenses. You can specify only one server for each report.

5 - License Use by User Report

Displays statistical information about the use of the licenses of a product, on the license server that you specify, in the time interval you specify. For each user, it lists the number of times each product was invoked, and the average time the user used each product. Statistical information about events appears after the license release. Not available for per-seat, simple nodelocked, or use-once (nodelocked or network) licenses.You can specify only one server for each report.

6 - Customer-Managed Use Audit

Reports the following information for customer-managed use product transactions:

- Vendor name
- Product name
- Product version
- Administrator information
- Time stamp of the event
- Number of licenses involved in the transaction
- Event list (product enrolled, license distributed, license deleted, license updated, per-server/per-seat license migrated)
- Signature stamp (user, group, and node)
- Signature information

Available for all license types.

-b *start_date*

Specifies the start date of a report. Be sure to express the date using the mm/dd/yyyy format. If you specify a start date and do not specify an end date, the report will include all information logged from the specified start date until the present.

-g *end_date*

Specifies the end date of a report. Be sure to express the date using the mm/dd/yyyy format. If you specify an end date and do not specify a start date, the report will include all information logged prior to (and including) the specified end date.

-e *event_filter*

You can use the event_filter argument to generate a Standard Event report on the following types of events which you specify, by number, on the command line. (Separate multiple event type arguments with a comma.)

1   All events

2   License-related events

3   Database-related events

4   Errors — general errors, but not including unrecoverable errors

5   Server-related events — start/stop events, threshold events, vendor events, unrecoverable errors

-F *server_type*

A filter on the type of server to be searched. Specify **l** for nodelocked license servers, **w** for network license servers, or **a** (the default) for both network license servers and nodelocked license servers.

-n *server_names*

Names of the servers about which you want to display information. If you want to generate a report for report types 2, 3, 4, or 5, you can specify only one server name.

-v *vendor_names*

Names of the vendors about whose products you want to display information.

-p { *product_name*[**?***product_version*] } **...**

Names of the products and, optionally, their versions about which you want to display information. If *product_name* or *product_version* contains blank characters, enclose the product name and product version in quotation marks, for example:

```
-p 'product name?product_version'
```

-u *user_names*

Names of users about whom you want to display license usage information.

Examples: Standard Event Report:

Report on license-related events (2) and server start and stop events (5) that were logged on server **neptune** since May 21, 2004:

```
i4blt -r1 -n "neptune" -b 05/21/2004 -e 2,5
```

License Use by Product Report:

Report current license usage information on server **saturn** for the products NetLS Test Product, Compiler, PsychoSynch, Megamail/2, EZ-Vectors, and DataVision:

```
i4blt -r4 -n "saturn" -p "'NetLS Test Product' 'Compiler'
PsychoSynch Megamail/2 EZ-Vectors DataVision"
```

License Request by User Report:

Report current license usage information on server **mercury** for users alex, ann, mary, christine, paul, and alby:

```
i4blt -r3 -n "mercury" -u "alex ann mary christine paul alby"
```

Customer-Managed Use Audit Report:

Report information about customer-managed use product transactions on all nodelocked license servers from May 1, 2004 to July 31, 2004:

```
i4blt -r6 -F l -b 05/01/2004 -g 07/31/2004
```

i4blt -R Reserve Licenses; Delete or Update Reserved Licenses

Use **i4blt -R** to reserve reservable licenses and to delete or update the reservation status of reserved licenses.

Syntax:

i4blt -R *action_type* [**r** | **d** | **u**]
-n *server_name*
-v *vendor_name*
-p *product_info*
[**-t** *timestamp*]
[**-A** *license_number*]

[**-g** *end_date*]
[**-H** *end_time*]
[**-u** *user_id user_group user_node*]

Parameters:

action_type

To reserve licenses, **r**; to delete licenses, **d**; to update an existing reservation, **u**.

-n *server_name*

The name of the server where the product license is enrolled.

-v *vendor_name*

The name of the vendor that manufactured the product.

-p *product_name product_version*

The information on the licensed product whose licenses you intend to reserve, delete, or update.

product_name

The name of the product.

product_version

The version of the product.

-t *timestamp*

Unique timestamp of the product license from which you intend to reserve, or that you intend to delete or update. To get the timestamp, issue the following command:

```
i4blt -lp -p "product_info" -i
```

If you are reserving licenses (option **-R r**), the timestamp is optional. If it is omitted, the first usable reservable license is used.

-A *license_number*

The number of licenses you intend to reserve. If you are updating a reservation (-R -u) or deleting licenses (-R -d), do not specify -A.

-g *end date*

The end date of the new or updated reservation (mm/dd/yyyy). The latest allowed expiration date of a reservation is 12/31/2037. If you are deleting licenses (**-R -d**), do not specify **-g**.

-H *end time*

The end time of the new or updated reservation (hh:mm). If you are deleting licenses (**-R -d**), do not specify **-H**.

-u *user_id user_group user_node*

The identification of the user, group, and node for which a license is being reserved or a reservation is being changed. Any of these values may be *,
meaning ″any″, unless you are using the named user policy. In this case the *user_id* must be explicitly declared. If you are deleting licenses (**-R -d**),
do not specify **-u**.

Examples: Reserve three licenses for Test Compiler product, taken from the reservable license identified by the timestamp 389588975, Version 1.1 of vendor Psychosync for any member of the testers group. They expire March 2, 2004, at 11:00.

```
i4blt -R r -v "Psychosync" -p "'Test Compiler' 1.1"
-t 389588975 -A 3 -g 03/02/2004 -H 11:00 -u "* testers *"
```

i4blt -s Display Product License Status

All users can use this option to display information about current product usage on specified license servers.

Syntax:

i4blt -s
[**-l** *list_type* [**c** | **pt** | **ps** | **ru** | **rr** | **cn** | **co**]]
[**-n** *server_names*]
[**-v** *vendor_names*]
[**-p** { *product_name*[**?***product_version*] } ...]
[**-u** *user_names*]

Parameters:

-l *list_type*

> Indicates the type of license usage you want to list. You can specify one of the following list types:

> **-c**   To display information related to concurrent users of concurrent and concurrent-offline licenses. Because concurrent-offline licenses can also be used as a concurrent license, you can see which users are using concurrent-offline licenses in a concurrent mode.

> **-pt**   To display information related to application clients that use per-seat licenses.

> **-ps**   To display information related to users of per-server licenses.

> **-ru**   To display information related to users of unreserved reservable licenses

> **-rr**   To display information related to users of reserved licenses.

> **-cn**   To display information related to users of concurrent nodelocked licenses. If you omit **-l**, its default value is **c**.

> **-co**   To display information related to users who have reserved concurrent-offline licenses in offline mode.

-n *server_names*

> The name of each of the license servers for which you want to display product usage information.

> If you omit the -n parameter, the display defaults to all servers in your cell. This parameter is not used if you use pt, ps, rr, or cn as the list type.

-v *vendor_names*

> The name of the vendor (or vendors) about whose products you want to display information.

-p { *product_name*[**?***product_version*] } **...**

> Names of the products and, optionally, their versions about which you want to display information. If *product_name* or *product_version* contains blank characters, enclose the product name and product version in quotation marks, for example:

> -p 'product name?product_version'

-u *user_names*

> Use the optional **-u** user_names argument to display product usage information for the specified products that are currently in use by the named users.

This command displays the following information for the servers, vendors, products, and users that you specify:

- Vendor name
- Product name
- Product version
- Total number of installed licenses
- Number of licenses currently in use
- Number of soft stop licenses currently in use
- Number of licenses not in use
- Number of queued users

For each user who currently holds a license, the following information is displayed:

- User name
- Node name
- Group name
- Number of licenses the user has been granted
- Check-out date for each granted license

Examples: Display current license availability and usage information for concurrent licenses of the Monolith Inc. product Megamail/2 on server **uranus**:

```
i4blt -s -lc -n "uranus" -v "'Monolith Inc.'" -p "Megamail/2"
```

i4blt -S Set the Stress Threshold

Use this option to set the network license server load threshold.

Syntax:

i4blt -S *threshold_value* **-n**   *server_name*

Parameters:

threshold_value
>Indicates the level of the load threshold. The range for this parameter is from 1 to 100.

-n *server_name*
>The name of the license server for which you want to set the load threshold.

Examples: Set the workload threshold to 70% for network license server **uranus**.

```
i4blt -S 70 -n uranus
```

i4blt -T Activate or Deactivate Tracing

Use this option to set the trace level on a specified network license server, or to change to a different level of tracing. Traces are written to output files whose file names have the format:

```
i4lmd.trcxx
```

where *xx* identifies a particular file in the sequence. For example, the first file would be named i4lmd.trc1, the last file would be named i4lmd.trc99. When i4lmd.trc99 is full, the license server begins writing traces to i4lmd.trc1 again. The current trace file has an underscore appended (for example, i4lmd.trc4_). By default the trace files are stored in the following directories:

AIX   /var/ifor

HP-UX, Solaris
     /var/lum

Windows
     C:\ifor\ls\conf\

Syntax:

i4blt -T *level-of-tracing* **-n**   *server_name*

Parameters:

level-of-tracing
     Indicates the level of tracing, as follows:

         0 – No trace

         1 – High (high level trace). Server logic and database transactions.

         2 – Low (low level trace). Network communication.

         3 – Low & high (low and high level trace)

-n *server_name*
     The name of the network license server for which you want to activate or deactivate the trace. You can activate or deactivate tracing for one server at a time. If you omit the **-n** *server_name* parameter, the tracing will be activated or deactivated on the local network license server (if there is one).

Examples: Set a high trace level for the network license server **plato**.

```
i4blt -T 1 -n plato
```

i4blt -U Update a Product

This option is valid only for customer-managed use products. It is issued for the following purposes:

- To update the number of licenses you enrolled. Use it when you acquire new licenses for an already enrolled customer-managed use product, to update the total number of licenses you are entitled to use. In the case of a network compound password, the licenses must be distributed after the update to make them available to end users.

- For a product with per-server/per-seat licenses, to switch from per-server to per-seat licenses.

- For a product with the hard stop/soft stop policy enabled, to change the hard stop/soft stop policy and to reset the high-water mark.

- To update the threshold value of a product.

- Set the maximum allowed period that a concurrent-offline license can be reserved.

Syntax:

i4blt -U
-v *vendor_name*
-p *product_name product_version*
[**-n** *server_name*]
[**-T** *enrolled_licenses*]
[**-S** *enable_switch* [**yes**]]
[**-H** *hard_soft_mode* [**yes** | **no**]]

[**-M** *hwm_reset*]
[**-t** *threshold* [**1...100**]]
[**-I** *signature_information*]
[**-O** *MaxOfflinePeriod*]

Parameters:

-v *vendor_name*

> The name of the vendor that manufactured the product that you intend to update.

-p *product_name product_version*

> The information on the licensed product that you intend to update.

> *product_name*

>> The name of the product for which you have acquired the new licenses.

> *product_version*

>> The version of the product that you specified in the product_name parameter.

-n *server_name*

> Name of the license server on which you want to update product information. This parameter is required if the product has nodelocked licenses and you are updating the product on a remote nodelocked license server. It is the name of the nodelocked license server. If you are updating the product on the local nodelocked license server, omit the -n parameter. If the product has network licenses, this parameter need not be specified, because the server is the central registry license server.

-T *enrolled_licenses*

> The total number of licenses you have for the specified product; that is, the number of licenses you had, plus the new ones.

-S *enable_switch*

> Use this parameter to migrate the license from per-server to per-seat. To use the per-seat license remember also to enroll the per-seat certificate. The only allowed value for -S is yes. When the licenses have been changed to per-seat, you cannot go back to per-server licenses.

-H *hard_soft_mode*

> Use this parameter to switch the product behavior from hard stop to soft stop and vice versa. You can do it only on products the vendor has enabled to allow hard stop/soft stop switching. Allowed values for -H are:

> **no**   Set the soft stop

> **yes**   Set the hard stop

-M *hwm_reset*

> Use this parameter to reset the high-water mark to 0. You can do it only on products the vendor has enabled to soft stop.

-t *threshold*

> Use this parameter to set a specific value for the threshold value of a customer-managed product. Allowed values are 1 to 100.

-I *signature_information*

> Information about the user issuing the command, to be stored with the signature stamp. Use this parameter with the -T parameter.

-O *MaxOfflinePeriod*
> The maximum allowed period that a concurrent-offline license can be reserved. Use this parameter with the -v and -p parameters.

Examples: The Test Compiler product (version 1.1, vendor Psychosync) has network licenses. Update the number of licenses to 50.

```
i4blt -U -v "Psychosync" -p "'Test Compiler' 1.1"
-T 50 -I "'Paula'"
```

For the same product, set the soft stop policy and reset the high-water mark.

```
i4blt -U -v "Psychosync" -p "'Test Compiler' 1.1" -H no -M
```

For the ScreenPic product (version 2, vendor ArtTools), update to 5 the number of nodelocked licenses on server *Virginia*:

```
i4blt -U -n Virginia -v "ArtTools" -p "ScreenPic 2" -T 5
```

To update (to 50) the maximum number of days a product can be reserved:

```
i4blt -U -v "'IBM Corporation'" -p "DataMaster"-O 50
```

i4blt -x Delete Server Log Entries

This option deletes all entries before a specified delete date from the log file of the license servers that you specify. If one of the specified license servers has the central registry, the central registry log entries are also deleted. If the specified license server is the local node, the nodelocked license server log entries are also deleted.

Syntax:

i4blt -x *delete_date*
[**-F** *server_type* { **l** | **w** | **a** }]
[**-n** *server_names*]

Parameters:

-x *delete_date*
> Specifies an end date for the delete operation. All log entries recorded before the delete date are removed from the log file. You must specify a delete date in the *mm/dd/yyyy* format. If you do not specify a *delete_date* value, all entries in the log file are deleted.

-F *server_type*
> A filter on the type of server to be searched. Specify l for nodelocked license servers, **w** for network license servers, or **a** (the default) for both network license servers and nodelocked license servers.

-n *server_names*
> Specifies the license servers from whose log file you want to delete the entries.

Examples: Delete all the log file entries recorded on server neptune before August 25, 2004:

```
i4blt -x 08/25/2004 -n neptune
```

Delete all the log file entries recorded on all nodelocked license servers before August 25, 2004:

```
i4blt -x 08/25/2004 -F l
```

i4cfg - Configuration Tool

ALL

Use the i4cfg command as an alternative to the Configuration Tool GUI or to the Configuration Tool script to configure your workstation. Before coding the i4cfg command, see "Before You Configure" on page 73 to plan your configuration requirements. If issued with no options, the i4cfg command starts the Configuration Tool GUI.

Syntax

i4cfg [**-a** { { **c,n,s,r** } | { **C,N,S,R** } }]
[**-b** { *"binding_list"* | **null** }]
[**-d** { *option_string* | **all** }]
[**-e** { **a** | **A** | { **e,t,w,c,g,v,m,p,s** } }]
[**-h**]
[**-l** *logfile_path*]
[**-list**]
[**-R** { **a,n,s** }]
[**-S** { **a,n,s** }]
 [**-start**]
[**-stop**]
[**-t** *"transport_list"*]

AIX

[**-c** { **d** | **a** | *cell_uuid* }]
[**-G** { *"site_list"* | **null** }]
[**-n** { **c** | **l** | **g** | **n** }]
[**-r** { **first** | **from:ip**:*host_name* }]
[**-script**]

Windows

[**-g** *group_name*]

Parameters

ALL

-a　　　The roles the workstation is to play in your licensing environment. Code any combination of these values, optionally separated by commas:

　　　c　　　Reset the current role of the workstation to network license client.

　　　n　　　Reset the current role of the workstation to nodelocked license server.

　　　s　　　Reset the current role of the workstation to network license server.

　　　r　　　Reset the current role of the workstation to central registry license server.

　　　C　　　Update the current role of the workstation to include network license client.

　　　N　　　Update the current role of the workstation to include nodelocked license server.

　　　S　　　Update the current role of the workstation to include network license server.

　　　R　　　Update the current role of the workstation to include central registry license server.

-b *"binding_list"*
　　　The complete list of servers (network license servers, nodelocked license servers, and central registry license server) with which this workstation

> will communicate in a direct binding environment. Enclose the complete list in double quotes. Specify the network license servers, nodelocked license servers, and central registry license server as follows:
>
> ```
> 'network ip:network_address1 [ port_number1 ] ip:network_address2
> [ port_number2 ] ... '
> 'nodelocked ip:network_address1 [ port_number1 ]
> ip:network_address2 [ port_number2 ] ... '
> 'registry ip:network_address [ port_number ]'
> ```
>
> Code -b null to delete all previously specified entries from the binding list.

-d *option_string*

> Display the current configuration settings for the i4cfg options specified in *option_string*. Code all to see the current settings of all the options.
>
> For example, i4cfg -d elS requests a display of which events are being logged, the path to the log databases, and a list of startup options showing which are enabled and which are disabled.

-e

> The list of events you want to be logged. Code **a** to log all events, **A** to disable logging of all events or any combination of these values, optionally separated by commas:
>
> **c - 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.
>
> **w - 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.
>
> **t - 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.
>
> **e - Errors**
>> Describes all types of server errors that do not stop the server. This is logged by default.
>
> **g - License grant/release**
>> Tells you when a license was granted or released. This is not logged by default.
>
> **v - Vendor added/deleted**
>> Tells you when a product of a new vendor was registered or deleted. This is logged by default.
>
> **m - Vendor messages**
>> Provides the log messages the vendor inserted in the enabled product. This is logged by default.
>
> **p - Product added/deleted**
>> Tells you when a new product was registered or deleted. This is logged by default.
>
> **s - Server start/stop/threshold**
>> Logs the successful start or stop of the license server, and when the threshold on the workload is exceeded or regained. This is not logged by default.

-h Displays command syntax and usage information about the Configuration Tool CLI.

-l *logfile_path*
  The path in which you want log files to be stored.

-list Displays a list of active subsystems.

-R Startup options that you want to disable. Code any combination of these values, optionally separated by commas:

  a Automatic startup of subsystems at system startup (disabled by default).

  n Remote administration of nodelocked license server (disabled by default).

  s Remote administration of network license server (enabled by default).

  Note: This parameter is not valid for network clients.

-S Startup options that you want to enable. Code any combination of these values, optionally separated by commas:

  a Automatic startup of subsystems at system startup (disabled by default).

  n Remote administration of nodelocked license server (disabled by default)

  s Remote administration of network license server (enabled by default)

  Note: This parameter is not valid for network clients.

-start Starts all the subsystems you have configured to run on the workstation.

  Note: If you are running in a Solaris environment and are using a non-Korn shell, you can start the server using `i4cfg -x -start`. This will close the terminal window from which this command was run and the server will continue to work in background.

-stop Stops all the subsystems that are running on your workstation.

-t *"transport_list"*
  Use this parameter to change the default port numbers, as follows:

  `"ip 'netls_port,crls_port,nodls_port'"`

  The three subparameters for port numbers are positional. If you omit one, its value is reset to the default. For example:

  `"ip "`
  `"ip '1516,10999,1215'"`
  `"ip ',,11999'"`

AIX

-c The NCS cell the workstation is to join. This parameter is meaningful only if namespace binding support is enabled (see the **-n** parameter). Code one of the following:

  d The default cell.

  a A new alternate cell. The Configuration Tool creates the UUID. You can retrieve the UUID from the glb_obj.txt file.

cell_uuid
> An alternate cell with the specified UUID.

> If you are configuring as a GLB replica (**-r from**), code this parameter to specify which cell this server is to join.

> **Note:** The glbd replica is not available on Linux operating systems.

-G *"site_list"*
> This parameter is meaningful only if namespace binding support is enabled (see the **-n** parameter). If your system does not support broadcasting or if the global location broker is running on a workstation in a different subnetwork, use this parameter to set the list of hosts running the global location broker. Clients can contact the servers using the site_list. List each server that runs the global location broker, in the form:
>
> `ip:network_address`
>
> Separate the entries with spaces, and enclose the entire list in double quotes. Code `-G null` to delete a previously-specified site list. In this case, clients must locate global location brokers by broadcasting. Before configuring a workstation to join an existing cell, check that there is no glb_site.txt file, or, if the file exists, that it includes a server that is in the cell being joined. Otherwise, use `-G null` to delete the existing site list.

-n
> Specifies namespace binding support. Code one of the following:
>
> **c**   Namespace binding support as a network license client only.
>
> **l**   This workstation is to run the local location broker but not the global location broker.
>
> **g**   This workstation is to run the global location broker and the local location broker.
>
> **n**   No namespace binding support (direct binding only).

-r
> This parameter is meaningful only if namespace binding support is enabled and this workstation is to run the global location broker (see the **-n** parameter). Code **first** if this is to be the first global location broker in a cell. Code **from:ip:***host_name* to replicate the global location broker that already exists on host_name. If you code **-r** from, you must also code the **-c** parameter to specify which cell this server is to join.

> **Note:** The glbd replica is not available on Linux operating systems.

-script Starts the interactive script to configure your workstation using a guided step-by-step procedure.

Windows

-g *group_name*
> The name of the group to which the user of this workstation belongs.

Examples

1. Configure a standalone nodelocked license server, specifying automatic startup of the server and customizing the path to the log files and the selection of events logged.

   For example, on UNIX and AIX:

   `i4cfg -a n -S a -e evmps -l /home/maveret`

For example, on Windows:

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

2. Configure a nodelocked license server in a network. Specify automatic startup of the server, make it possible to administer licenses on another nodelocked licensed server (**louise**) remotely, and customize the path to the log files and the selection of events logged.

   With direct binding on UNIX and AIX:

   ```
   i4cfg -a n -S a,n -e evmps -l /home/maveret -b 'nodelocked ip:louise' -n n
   ```

   With direct binding on Windows:

   ```
   i4cfg -a n -S a,n -e evmps -l c:\ifor\ls\maveret -g Infotech
   -b 'nodelocked ip:louise
   ```

3. With namespace binding on UNIX and AIX, joining an existing cell that has UUID 456b91c50000.0d.00.00.87.84.00.00.00:

   ```
   i4cfg -a n -S a,n -e evmps -l /home/maveret -b null -n l
   -c 456b91c50000.0d.00.00.87.84.00.00.00
   ```

   The nodelocked license server *louise* must belong to the same cell.

4. Configure a network license server (**thelma**). Specify automatic startup of the server, and customize the path to the log files and the selection of events logged. Configure to communicate with:
   - Network license server **louise**
   - Nodelocked license server **louise**
   - Nodelocked license server **speedy**
   - Central registry license server **speedy**

   With direct binding on UNIX and AIX:

   ```
   i4cfg -a s -S a,s -e cegvp -l /home/maveret -b "'network
    ip:thelma ip:louise' 'nodelocked ip:speedy ip:louise'
    'registry ip:speedy'" -n n
   ```

   With direct binding on Windows:

   ```
   i4cfg -a s -S a,s -e cegvp -l c:\ifor\ls\maveret -g Infotech
   -t "ip" -b "'network ip:thelma ip:louise'
   'nodelocked ip:speedy ip:louise'
   'registry ip:speedy'"
   ```

   With namespace binding on UNIX and AIX, starting a new alternate cell:

   ```
   i4cfg -a s -S a,s -e cegvp -l /home/maveret -b null -n g -r first
   ```

   Note that **speedy** and **louise** must join this new cell.

5. Configure a network license client that will communicate with a workstation named **thelma** that is configured as both a network license server and the central registry license server. For Windows specify the user group.

   With direct binding on UNIX and AIX:

   ```
   i4cfg -a c -b "'network ip:thelma' 'registry ip:thelma'" -n n
   ```

   With direct binding on Windows:

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

   With namespace binding, joining an existing cell that has UUID 456b91c50000.0d.00.00.87.84.00.00.00:

   ```
   i4cfg -a c -b null -n c -c 456b91c50000.0d.00.00.87.84.00.00.00
   ```

   Note that **thelma** must belong to the same cell.

6. Configure a workstation named **thelma** as the central registry license server and a network license server. Configure to communicate with a network license server named **hydra**. Specify automatic startup of the servers, and for Windows specify the user group.

    With direct binding on UNIX and AIX:

    ```
    i4cfg -a s,r -S a,s -b "'network ip:thelma ip:hydra'
    'registry ip:thelma'" -n n
    ```

    With direct binding on Windows:

    ```
    i4cfg -a s,r -S a,s -g Infotech -b "'network ip:thelma ip:hydra'
    'registry ip:thelma'"
    ```

    With namespace binding on UNIX and AIX, joining an existing alternate cell that has UUID 789b91c50000.0d.00.00.87.84.00.00.00 and replicating the global location broker at the server **hydra**:

    ```
    i4cfg -a s,r -S a,s -b null -n g -r from:ip:hydra
    -c 789b91c50000.0d.00.00.87.84.00.00.00
    ```

    Note: The glbd replica is not available on Linux operating systems.

7. Cancel all entries previously made in the direct binding servers list:

    ```
    i4cfg  -b null
    ```

8. Display the command syntax and usage:

    ```
    i4cfg  -h
    ```

i4tv - Test Verification Tool

ALL

Use the i4tv tool after the license servers are started to verify that they are running properly.

A message describing a completed license transaction and a list of all license servers will be displayed. If an HAL cluster has been configured and activated, i4tv tries to request all the 10,000 licenses of the HAL Test Product in order to contact all the servers in serving state.

Syntax

i4tv { [**-n** *hostname*] [**-z**] [**-v**] | { **-h** | **-usage** | **-version** } [**-p** *number_of_transactions*] }

Parameters

-n *hostname* Checks that the specified workstation is running a network license server. It returns 0 if the hostname is running a network license server and 1 if the hostname is not running a network license server.

-z Turns on NCS remote procedure call (RPC) tracing messages, which can be used to diagnose problems.

-v Displays progress messages during the license request operation.

-h Displays command usage information (same as -usage). This parameter is valid only when issued without other parameters.

-usage Displays command usage information (same as -h). This parameter is valid only when issued without other parameters.

-version Displays command version information. This parameter is valid only when issued without other parameters.

-p [*number_of_transactions*]
Specifies the number of transactions to be completed before performance information is displayed. This information provides averages for the specified period. It can be used for tuning the system and for troubleshooting performance problems. The default value is 1000.

Examples

Run the i4tv test and verification tool:

```
i4tv
```

Check for the presence of the license server **pluto**:

```
i4tv -n pluto
```

i4target - Target View Tool

ALL

Use the i4target tool to display the target ID of your workstation.

Syntax

i4target [**-O** | **-V** | **-o** | **-l** | **-h**]
(on AIX, and Solaris)

i4target [**-O** | **-V** | **-o** | **-l** | **-h** | **-v**]
(on Linux)

i4target [**-O** | **-V** | **-o** | **-c** | **-h**] [**-v**] [**-q**]
(on HP-UX)

i4target [**-O** | **-V** | **-o** | **-l** | **-h** | **-d** *devicename*] [**-z**] [**-v**]
(on Windows)

Parameters

-O Displays the target identifier of the workstation on which you issue the command, in the form that the license creation tool accepts.

-V Displays command version information.

-o Displays the operating system name of your workstation.

-c Displays multiple target IDs for workstations, that have more than one network connection, and enables you to change from one that is currently in use to another (HP-UX only).

-h Displays command usage information.

-v Displays information in verbose mode. On Windows, the information that this parameter displays includes the NDIS level of the device driver.

-q Displays information in quite mode, comments are not displayed before the output (HP-UX only).

-d *devicename*
Specifies that the device driver *devicename* is to be used to retrieve the target ID (Windows only).

-z Displays all the devices on the workstation; use it to check the available network devices before using the -d parameter to select a device driver manually (Windows only).

-l Displays all target IDs of the workstation, starting with the most secure (the one based on the network adapter, if it is available).

Windows

For Windows, the hardware-based (win32mac) target ID is retrieved from the device driver that controls the network adapter, rather than from the operating system. This mechanism results in the highest level of security and makes it possible for the target ID to work even when the workstation is not connected to the network.

For this mechanism to work, the following must be installed on the workstation where the license will be installed:

• A device that complies with NDIS 4.
• The latest level of the device driver. For IBM device drivers, you can check at http://www.networking.ibm.com to determine the latest level.

- TCP/IP.
- If License Use Management Runtime is not yet installed on the workstation, you can download it from the License Use Management Web site (http://www.ibm.com/software/lum), or the vendor of the license-enabled products can supply it to customers.

The target ID is retrieved from the device driver that controls the network adapter. You may want to specify explicitly the device driver to be used to determine the target ID. You would do this if either of the following points were true:

- Your workstation has multiple network adapters and you prefer to use an adapter other than that automatically selected as the basis for the target ID.
- The **target -Ol** command did not return a target ID.

Perform the following steps:

1. Get the device name of the adapter you want to use, in one of the following ways:
   - For Windows 2000, Windows XP, and Windows Server 2003, to get the device name of the adapter you want to use, enter the command:

     ```
     ipconfig /all
     ```

     For example, for IBM token ring adapter IBMTRP1, the information returned includes the line:

     ```
     Token ring adapter IBMTRP1:
     ```
   - Alternatively for Windows, enter the command:

     ```
     i4target -z
     ```

     This displays all the devices on the workstation, and be sure the network device you want to use is listed.

2. Enter the command:

   ```
   i4target -d devicename
   ```

   For example, `i4target -d IBMTP1`.

Note: The AIX PREFERRED LUM Target ID is only compatible with License Use Management Version 4.6.6 or later. If you need a license to be installed on an older license server, or a nodelock license for an application enabled with an older License Use Management toolkit, then request the license using the LUM Target ID shown second in the output of the **i4target -Ol** command.

i4lmd - Network License Server

ALL

The i4lmd subsystem starts the network license server on the local node. If the workstation is not configured to run the network license server, i4lmd has no effect.

The parameters of i4lmd override the corresponding settings in the i4ls.ini file.

Syntax

startprc i4lmd [**-no** *event_list*] [**-v**] [**-z**] [**-l** *log_name*]
[**-s**] [**-r**] [**-c**] [**-p**]
 (on UNIX)

startsrc -s i4lmd [**-a** ″ [**-no** *event_list*] [**-v**] [**-z**] [**-l** *log_name*]
[**-s**] [**-r**] [**-c**] [**-p**] ″
] (on AIX)

i4ls [**-no** *event_list*] [**-v**] [**-z**] [**-l** *log_name*] [**-s**] [**-r**] [**-c**] [**-p**]
 (on Windows)

Parameters

-no event_list Turns off logging of the events specified in event_list. Any combination of events is valid, but items in the list of events must not be separated by spaces or other characters. Following are the event types that you can specify:

l Grant and release licenses.

c Check in licenses. (Licensed products usually check in with the license server at regular intervals while a user is using the product.)

w Waiting events: these include wait events (a user was waiting for a license), wait grant events (a user was waiting for and then was granted a license), and wait remove events (a user was waiting for a license and then asked to be removed from the queues before a license was granted.)

v Vendor events: a vendor was added, renamed or deleted.

p Product events: a product was added, renamed, or deleted.

e Errors.

m Messages.

s Logs the successful start or stop of the license server, and when the threshold on the workload is exceeded or regained.

t License timeout events. (When a licensed product fails to check in with the license server, it may stop running after it times out. The vendor of the product sets the timeout interval, which is how long a product can run after it has lost contact with the license server.)

-v License Use Management Runtime library verbose mode.

-z Debugging flag. Prints RPC debugging information.

-l *log_name*   Overrides the default name and location of the file used to store log information. This allows the I/O activity to the files used by the license server to be spread across multiple file systems that may become important for large installations.

-s   Instructs the license server to ignore attempts from administrators on remote systems to modify the license database. Records in the database remain readable by all instances of the License Use Management Runtime Administration Tool.

-r   Recovers files from the automatic backup version.

-c   Specifies that this is a cold start, meaning that the license server restarts from scratch, as if it had granted no licenses before stopping.

-p   Specifies that i4lmd is to display performance information at specified intervals. The default and maximum interval is 1000 calls received from clients. To change the frequency of reporting, change the environment variable I4_POLL_FREQ. This option is linked to the DebugProc tag in the i4ls.ini file.

Examples

Start a license server on UNIX and do not log checkin, vendor, product, timeout, or message events:

```
startprc i4lmd -no cvptm
```

To start a license server, changing the default log-file on AIX:

```
startsrc -s i4lmd -a "-l /home/smith/my_log"
```

To start a license server, changing the default log-file on Windows:

```
i4ls -l \ifor\ls\my_log
```

i4llmd - Nodelocked License Server

ALL

The i4llmd subsystem starts the nodelocked license server on the local node. If the workstation is not configured to run the nodelocked license server, i4llmd has no effect.

The parameters of i4llmd override the corresponding settings in the i4ls.ini file.

Syntax

startprc i4llmd -b [**-no** *event_list*] [**-v**] [**-l** *log_name*] [**-s**]
[**-r**] [**-c**] (on UNIX)

startsrc -s i4llmd [**-a** ″ [**-no** *event_list*] [**-v**] [**-l** *log_name*] [**-s**]
[**-r**] [**-c**] ″] (on AIX)

i4ls -b [**-no** *event_list*] [**-v**] [**-l** *log_name*] [**-s**] [**-r**] [**-c**] (on Windows)

Parameters

-b   Starts License Use Management Runtime on a nodelocked license server.

-no *event_list*   Turns off logging of the events specified in the event list. Any combination of events is valid, but items in the event list must not be separated by spaces or other characters. Following are the event types that you can specify:

   l   Grant and release licenses.

   v   Vendor events: a vendor was added, renamed or deleted.

   p   Product events: a product was added, renamed, or deleted.

   e   Errors.

   m   Messages.

   s   Starts and stops of this license server.

   t   Time out.

-v   License Use Management Runtime library verbose mode.

-l *log_name*   Overrides the default name and location of the file used to store log information. This allows the I/O activity to the files used by the license server to be spread across multiple file systems that may become important for large installations.

-s   Instructs the license server to ignore attempts from administrators on remote systems to modify the license database. Records in the database remain readable by all instances of the basic license tool.

-r   Recovers files from the automatic backup version.

-c   Specifies that this is a cold start, meaning that the license server restarts from scratch, as if it had granted no licenses before stopping.

Examples

Start a nodelocked license server on UNIX and do not log checkin, vendor, product, or message events:

```
startprc i4llmd -b -no cvpm
```

Start a nodelocked license server changing the default log file on AIX:

```
startsrc -s i4llmd -a "-l /ifor/ls/my_log"
```

Start a nodelocked license server on Windows, disabling remote administration from instances of the basic license tool on other workstations:

```
i4ls -b -s
```

i4gdb - Central Registry License Server

ALL

The central registry is a License Use Management Runtime subsystem that provides a mechanism for storing data pertaining to licensing information. There must be one and only one central registry license server running per cell. This ensures that the data is accurate and complete.

The basic license tool requires a central registry license server up and running to administer customer-managed use products.

In namespace binding, if more than one i4gdb is found in a given cell, the newly started i4gdb automatically shuts down.

In direct binding there is no such control, and you must double-check that you have started one and only one central registry license server in your licensing environment by issuing the following command on every License Use Management Runtime server:

```
i4cfg -list
```

If the workstation is not configured to run the central registry license server, i4gdb has no effect.

The parameters of i4gdb override the corresponding settings in the i4ls.ini file.

Syntax

startprc i4gdb -a [**-no** *event_list*] [**-v**] [**-l** *log_name*]
[**-r**] [**-c**] [**-z**] (on UNIX)

startsrc -s i4gdb [**-a** ″ [**-no** *event_list*] [**-v**] [**-l** *log_name*]
[**-r**] [**-c**] [**-z**] ″] (on AIX)

i4ls -a [**-no** *event_list*] [**-v**] [**-l** *log_name*] [**-r**] [**-c**] [**-z**] (on Windows)

Parameters

-no *event_list*   Turns off logging of the events specified in the event list. Any combination of events is valid, but items in the event list must not be separated by spaces or other characters. Following are the event types that you can specify:

        l   Grant and release licenses.

        c   Check in licenses. (Licensed products usually check in with the license server at regular intervals while a user is using the product.)

        v   Vendor events: a vendor was added, renamed or deleted.

        p   Product events: a product was added, renamed, or deleted.

        e   Errors.

        m   Messages.

        s   Starts and stops of this license server.

        t   Time out.

-v   License Use Management Runtime library verbose mode.

-l *log_name*   Overrides the default name and location of the file used to store log information. This allows the I/O activity to the files used by the license server to be spread across multiple file systems that may become important for large installations.

-r   Recovers files from the automatic backup version.

-c   Specifies that this is a cold start, meaning that the license server restarts from scratch, as if it had granted no licenses before stopping.

-z   Debugging flag. Prints RPC debugging information.

lb_admin - Local Broker Administration

AIX

The Local Broker Administration tool (lb_admin) administers the registrations of NCS-based servers in global location broker (GLB) or local location broker (LLB) databases. A server registers universal unique identifiers (UUIDs) specifying an object, a type, and an interface, along with a socket address specifying its location. A client can locate servers by issuing lookup requests to GLBs and LLBs.

Use the Local Broker Administration tool (lb_admin) to look up information, add new entries, and delete existing entries in a specified database.

The Local Broker Administration tool is useful for inspecting the contents of location broker databases and for correcting database errors. For example, if a server terminates abnormally without unregistering itself, use Local Broker Administration (lb_admin) to manually remove its entry from the GLB database.

When accepting input or displaying output, Local Broker Administration (lb_admin) uses either character strings or descriptive textual names to identify objects, types, and interfaces. A character string directly represents the data in a UUID in the format:

```
xxxxxxxxxxx.xx.xx.xx.xx.xx.xx.xx.xx
```

where each x is a hexadecimal digit.

Local Broker Administration (lb_admin) will examine or modify only one database at a time. This is referred to as the current database. The use_broker command selects the type of location broker database, GLB or LLB. The set_broker command selects the host whose GLB or LLB database is to be accessed. Of course, if one replica of a replicated GLB database is modified, the modifications will be propagated to the other replicas of that database.

Note: The glbd replica is not available on Linux operating systems.

Syntax
lb_admin [**-nq**] [**-version**]

Parameters

-nq Do not query for verification of wildcard expansions in unregister operations.

-version Display the version of NCS that this lb_admin tool belongs to, but do not start the tool.

Commands
When you type:

```
lb_admin
```

you are prompted with the following line, where you can enter the lb_admin commands:

```
lb_admin:
```

In lookup, register, and unregister commands, the *object*, *type*, and *interface* arguments can be either character strings representing UUIDs or textual names corresponding to UUIDs, as described earlier.

a[dd] Synonym for register.

c[lean]

Find and delete obsolete entries in the current database. When issuing this command, lb_admin attempts to contact each server registered in the database. If the server responds, the entry for its registration is left intact in the database. If the server does not respond, lb_admin looks up its registration in the LLB database at the host where the server is located, tells the result of this lookup, and asks if the entry is to be deleted. If a server responds, but its UUIDs do not match the entry in the database, lb_admin tells this result and asks if the entry is to be deleted. Entries that meet either of these conditions are probably safe to delete:

- The server does not respond. The lb_admin succeeds in contacting the LLB at the host where the server is located, but the server is not registered with that LLB. The server is probably no longer running.

- A server responds, but its UUIDs do not match the entry in the database. The server that responds is not the one that registered the entry.

Entries that meet either of these conditions are probably safe to delete. In other situations, it is best not to delete the entry unless it can be verified directly that the server is not running (for example, by listing the processes running on its host). When lb_admin asks to delete an entry, you can respond in four ways:

- A **y[es]** response deletes the entry.

- A **n[o]** response leaves the entry intact in the database. After a **yes** or a **no**, lb_admin proceeds to check the next entry in the current database.

- A **g[o]** response invokes automatic deletion, in which all eligible entries are deleted and all ineligible entries are left intact, without the user being queried, until all entries have been checked.

- A **q[uit]** response terminates the clean operation.

d[elete]

Synonym for unregister.

h[elp] [*command*] **or ?** [*command*]

Display a description of the specified command or, if none is specified, list all the lb_admin commands.

l[ookup] *object type interface*

Look up and display all entries with matching object, type, and interface fields in the current database. You can use an asterisk (*) as a wildcard for any of the parameters. If all the parameters are wildcards, lookup displays the entire database.

q[uit] Exit the lb_admin session.

r[egister] *object type interface location annotation* [*flag*]

Add the specified entry to the current database. Use an asterisk (*) to represent the null UUID in the *object*, *type*, and *interface* fields.

The *location* is a string in the format *family:host*[*port*], where *family* is an address family, *host* is a host name, and *port* is a port number. A leading # can be used to indicate that a host name is in the standard numeric form.

The following are sample location specifiers:

```
ip:vienna[1756]
ip:#192.5.5.5[1791]
```

The *annotation* is a string of up to 255 characters annotating the entry. Use double quotation marks to enclose a string that contains a space or contains no characters. To embed a double quotation mark in the string, precede it with a backslash.

The *flag* is either **local** (the default) or **global**, indicating whether the entry should be marked for local registration only or for registration in both the LLB and GLB databases. The flag is a field that is stored with the entry but does not affect where the entry is registered. The **set_broker** and **use_broker** commands select the particular LLB or GLB database for registration.

s[et_broker] [*broker_switch*] *host*

Set the host for the current LLB or GLB. If specifying **global** as the broker switch, **set_broker** sets the current GLB, otherwise it sets the current LLB. The host is a string in the format *family:host*, where *family* is an address family and *host* is a host name. Use a leading # to indicate that a host name is in the standard numeric form. The following are sample location specifiers:

```
ip:prague
ip:#192.5.5.5
```

Issue **use_broker**, not this command, to determine if subsequent operations will access the LLB or the GLB.

set_t[imeout] [short | long]

Set the timeout period used by lb_admin Administration for all of its operations. With an argument of **short** or **long**, **set_timeout** sets the timeout accordingly. With no argument, it displays the current time-out value.

u[nregister] *object type interface location*

Delete the specified entry from the current database. The location is a string in the format *family:host*[*port*], where *family* is an address family, *host* is a host name, and *port* is a port number. Use a leading # to indicate that a host name is in the standard numeric form. The following are sample location specifiers:

```
ip:vienna[1756]
ip:#192.5.5.5[1791]
```

You can use an asterisk as a wildcard in the *object*, *type*, and *interface* fields to match any value for the field. Unless queries have been suppressed by invoking lb_admin with the **-nq** option, **unregister** allows deletion of each matching entry.

- A **y[es]** response deletes the entry.
- A **n[o]** response leaves the entry in the database.
- A **g[o]** response deletes all remaining database entries that match, without querying.
- A **q[uit]** response terminates the unregister operation, without deleting any additional entries.

us[e_broker] [broker_switch]

Select the type of database that subsequent operations will access, GLB or LLB. The *broker_switch* is either **global** or **local**. If a broker switch is not supplied, **use_broker** determines if the current database is global or local.

Use **set_broker** to select the host whose GLB or LLB is to be accessed.

Examples

1. Set the global location broker as the default database.

   ```
   lb_admin
   lb_admin: use global
   ```

2. Find and delete obsolete entries in the global location broker database.

   ```
   lb_admin: clean
   ```

   This is the output, if there are no entries to be cleaned:

   ```
   0 Entries deleted of 8 processed
   ```

3. Exit the tool:

   ```
   lb_admin: quit
   ```

drm_admin - GLBD Replicas Administration

The GLBD Replicas Administration tool (drm_admin) administers servers based on the NCS location brokers such as glbd, the replicated version of the global location broker. With the GLBD Replicas Administration tool (drm_admin), the replica lists can be inspected or modified, databases can be merged to force convergence among replicas, servers can be stopped, and replicas can be deleted.

Note: The glbd replica is not available on Linux operating systems.

The role of the GLBD Replicas Administration tool (drm_admin) is to administer the databases, not to change the data they contain. For instance, you can use GLBD Replicas Administration (drm_admin) to merge two replicas of the global location broker database, but the Local Broker Administration (lb_admin) must be used to add a new entry to the database.

Also, although GLBD Replicas Administration (drm_admin) can stop or delete a global location broker replica, glbd must be invoked directly to start or create a replica. After you start it, GLBD Replicas Administration (drm_admin) enters an interactive mode in which it accepts the following commands.

Syntax
drm_admin [-version]

Parameters

-version　　　　Displays the version of NCS that this GLBD Replicas Administration (drm_admin) belongs to, but does not start the tool.

Commands
When you type:
```
drm_admin
```

you are prompted with this line:
```
drm_admin:
```

where you can enter the drm_admin commands. Most drm_admin commands operate on a default object (**default_obj**) at a default host (**default_host**). Together, **default_obj** and **default_host** specify a default replica. Defaults are established by the **set** command and are remembered until changed by another set. Currently, the only known object is glb.

Some drm_admin commands operate on a host other than the default. Identify this host as **other_host**. The host name supplied as a default_host or an other_host takes the form *family:host*, where the host can be specified either by its name or by its network address. The following are examples of acceptable host names:
```
ip:bertie
ip:#192.5.5.5
```

addrep other_host
　　　　Add other_host to the replica list at default_host. The replica at default_host will propagate other_host to all other replica lists for default_obj.

chrep -from other_host -to new_other_host
　　　　Change the network address for other_host in the replica list at

default_host to new_other_host. The replica at default_host will propagate this change to all other replica lists for default_obj. The chrep command will fail if a replica of default_obj is running at other_host or if other_host is not on the replica list at default_host.

delrep other_host

Delete the replica of default_obj at other_host. The delrep command tells the replica at other_host to do the following:

1. Propagate all of the entries in its propagation queue.
2. Propagate a delete request to all other replicas, causing other_host to be deleted from all other replica lists for default_obj.
3. Delete its copy of default_obj.
4. Stop running.

The delrep command returns you immediately to the GLBD Replicas Administration prompt, but the actual deletion of the replica can take a long time for configurations that are not stable and intact. Check to see if the daemon for the deleted replica has stopped by listing the processes running on its host.

info   Get status information about the replica for default_obj at default_host.

lrep [-d] [-clocks] [-na]

List replicas for default_obj as stored in the replica list at default_host.

The **-d** option lists deleted as well as existing replicas.

The **-clocks** option shows the current time on each host and indicates the time difference between the replicas.

The **-na** option lists the network address of each host.

merge { -from | -to } other_host

The merge command copies entries in the default_obj database and replica list from one replica to another. It copies an entry if no corresponding entry exists in the destination database or if the corresponding entry in the destination database bears an earlier time stamp. A merge does not cause entries to be propagated. The database and replica list at the origination are not changed.

The **-from** parameter copies entries from the default_obj database and replica list at other_host to the default_obj database and replica list at default_host.

The **-to** parameter copies entries from the database and replica list at default_host to the database and replica list at other_host.

A `merge -from` followed by a `merge -to` causes the replicas at the two hosts to converge.

merge_all

This command uses default_host as the hub for a global merge of all replicas for default_obj. For each host on the replica list at default_host, a `merge_all` first runs a `merge -from`, then runs a `merge -to`. All replicas of default_obj are thereby forced into a consistent state. The merge_all operation does not cause any entries to be propagated. You should run a merge_all when:

- A replica is purged
- A replica is reset
- A replica has been not communicating for two weeks or more

- A replica stops (for example, when its database is destroyed by a disk failure)

monitor [-r n]

This command causes drm_admin to read the clock of each replica of default_obj every *n* minutes and to report any clock skews or non-answering replicas. If **-r** is not specified, the period is 15 minutes.

purgerep other_host

This command purges other_host from the replica list at default_host. The replica at default_host then propagates a delete request to the replicas at the hosts remaining on its list, thereby removing other_host from all other replica lists for default_obj. The delete request is not sent to other_host. A purgerep can cause data to be lost and should only be used when a replica has stopped. It is strongly recommended that a **merge_all** operation be run after the purgerep to prevent the remaining replicas of the default_obj database from becoming inconsistent. If the purged replica is still running, it should be reset. It is recommended that you use **chrep** (rather than **addrep** and **purgerep**) to change entries on the replica list.

quit   Quit the drm_admin session.

reset other_host

Reset the replica of default_obj at other_host. The reset command tells the replica at other_host to delete its copy of default_obj and to stop running. It does not cause other_host to be deleted from any other replica lists. This command can cause data to be lost unless a successful **merge_all** is run first.

set [-o *obj_name* **] -h** *host_name*

Set the default object and host. All subsequent commands will operate on obj_name. Subsequent commands that do not specify a host will be sent to *host_name*. If the **-o** option is not specified, drm_admin keeps the current *default_obj*. If set is used with the **-o** option, drm_admin checks the clocks at all hosts with replicas of the object.

stop   Stop the server for *default_obj* that is running at *default_host*.

Examples

The following example starts drm_admin, sets the default object to **glb**, and sets the default host to **ip:mars**:

```
drm_admin
drm_admin: set -o glb -h ip:mars
```

This is the output:

```
Default object: glb default host:
ip:mars state: in service
Checking clocks of glb replicas
ip:mars 2004/04/09.17:09
ip:pluto 2004/04/09.17:09
ip:mercury 2004/04/09.17:07
```

lb_find - GLBs List

The GLBs List (lb_find) lists global location broker processes and their attributes. It sends out inquiries to the NCS location broker processes and gathers the responses.

If on the workstation where you issue the command there are not the two files:

```
glb_site.txt
glb_obj.txt
```

lb_find finds all the GLB processes of the same subnet, or all the GLB processes it can reach via broadcast.

If you want to see GLB processes of a different subnetwork you must have those two files. In such a case **lb_find** finds all the GLB processes that run on the hosts whose addresses are in the `glb_site.txt`, situated in the cell specified in the `glb_obj.txt`. If the `glb_obj.txt` is not on the workstation, the default cell is taken.

The results are analyzed to determine whether or not the global location broker can be replicated, and which cell each daemon serves. After 10 seconds, the results are summarized, showing the server host's network address, the port number, the global location broker type, a cell name of either `default` or `alternate_n`, where *n* is a number greater than or equal to 1, and the cell's UUID.

Note: The glbd replica is not available on Linux operating systems.

Syntax

lb_find [**-dl**] [**-f ip** | **-q**] [**-h**] [**-v**]

Parameters

-dl   Turn on RPC remote procedure call (RPC) debugging while searching for GLB servers.

-f   Query for the global location broker servers that communicate with the specified protocol in all the cells.

-q   Query for a global location broker server using the standard RPC mechanism. At most, one global location broker server is printed, and only servers in the current workstation's cell are searched. The program exits with a status of 0 if a global location broker server is found; otherwise, the status is nonzero.

-h   Print out the help for the command.

-v   Print out the NCS version string.

Examples

A network contains two global location broker processes (glbd) in the default NCS cell.

```
lb_find
```

This is the output:

```
sent to broadcast address ip:#9.87.220.255
      waiting for replies
      ip:server5(9.87.220.5) 1024 replicatable default
      333b91c50000.0d.00.00.87.84.00.00.00
      ip:server3(9.87.220.3) 1072 replicatable default
      333b91c50000.0d.00.00.
      87.84.00.00.00
```

uuid_gen - UUID Generator

Use the uuid_gen tool to generate the UUID (universal unique identifier) for an NCS cell. The UUID is a string of 28 hexadecimal characters string, and is contained in the glb_obj.txt file.

Syntax
uuid_gen

Example
To generate the UUID:

```
uuid_gen
```

This is an example of the output:

```
54c7874546ae.0.2.81.87.92.34.0.0.00.00
```

llbd - Local Location Broker Subsystem

The local location broker subsystem (llbd) is part of the network computing system (NCS). It manages the local location broker (LLB) database, which stores information about NCS based server programs running on the local host.

A host must run llbd if it is to support the location broker forwarding function or to allow remote access, for example by the Local Broker Administration (lb_admin) to the LLB database. In general, any host that runs NCS-based servers should run an llbd, and llbd should be running before any such servers are started. Additionally, any network or internet supporting NCS activity should have at least one host running a global location broker subsystem (glbd).

Syntax

startprc llbd [**-dl**] (on UNIX)

startsrc -s llbd [**-a** ″ **-dl** ″] (on AIX)

Parameters

-dl   Prints debugging information.

glbd - Global Location Broker Subsystem

The global location broker (GLB) subsystem (glbd) helps clients to locate servers on a network or internet. The GLB database stores the locations (that is, the network addresses and port numbers) where server subsystems are running. A process maintains this database and provides access to it.

You can replicate the GLB database to increase its availability. Copies of the database can exist on several hosts, with glbd running on each of those hosts to maintain the consistency of the database replicas. (In an internet, at least one glbd must run in each network.) Each replica of the GLB keeps a list of all the other GLB replicas. The GLBDs Replicas Administration (drm_admin) tool administers the replication of the GLB database and of the replica list.

Note: The glbd replica is not available on Linux operating systems.

The glbd command writes diagnostic output to the file `/etc/lum/ncs/glb_log` (on UNIX) and `/etc/ncs/glb_log` (on AIX).

Syntax

startprc glbd [**-create** { **-first** [**-family** *family_name*]
| **-from** *host_name* }] [**-debug**] [**-log_stdout**] (on UNIX)

startsrc -s glbd [**-a** ″ [**-create** { **-first** [**-family** *family_name*]
| **-from** *host_name* }] [**-debug**] [**-log_stdout**] ″] (on AIX)

Parameters

-create   Creates a replica of the GLB. This option creates a GLB database in addition to starting a broker subsystem. It must be used with either -first or -from.

-first   Use this option only with -create. Use it to create the first replica (that is, the very first instance) of the GLB on the network or internet.

-family *family_name*
            Use this option only with **-first**, to specify the address family that the first replica will use to identify itself on the replica list. Any subsequently created replicas must use this family to communicate with this replica. The family name can only be **ip**.

-from *host_name*
            Use this option only with **-create**, to create additional replicas of the global location broker. A replica of the global location broker must exist at *host_name*. The database and replica list for the new replica are initialized from those at *host_name*. The replica at *host_name* adds an entry for the new replica to its replica list and propagates the entry to the other global location broker replicas. A *host_name* takes the form *family:host*, where the host can be specified either by its name or by its network address.

            Note: The **-from** parameter is not available on Linux operating systems.

            The following are examples of acceptable host names:
            `ip:bertie`
            `ip:#192.5.5.5`

The new replica uses the same address family as *host_name* in identifying itself on the replica list. For example, if *host_name* is an IP address, the new replica is listed by its IP address on the replica list.

-debug   Prints debugging information.

-log_stdout   Redirects the log and debug printout to standard output instead of `/etc/ncs/glb_log`.

Examples

Create and start for the first time the first replica of the GLB on the network or internet:

```
startprc glbd -a "-create -first -family ip"
```

Start for the first time a subsequent replica of the GLB, initializing its database from host buddy:

```
startprc glbd -a "-create -from ip:buddy"
```

Restart an existing replica of the GLB:

```
startprc glbd
```

i4glbcd - Global Location Broker Database Cleaner Subsystem

Note: The **i4glbcd** subsystem is not available on Linux operating systems.

The i4glbcd subsystem automatically cleans up incorrect entries in the global location broker database. Do not start more than one instance of i4glbcd in an NCS cell.

Syntax

startprc i4glbcd [**-nq**] (on UNIX)

startsrc -s i4glbcd [**-a** ″**-nq**″] (on AIX)

Parameters

-nq   Verbose mode. This causes **i4glbcd** to display debugging information to standard output. Use this information if you need to call IBM support.

Chapter 7. Managing the licensing environment effectively

This chapter provides information that is intended to help you manage your licensing environment better.

Managing Concurrent and Concurrent-Offline licenses on the same server

If you have both the concurrent and concurrent-offline licenses enrolled on the same server, then if the license is requested for concurrent use, the concurrent license will be ranted first if available. If all concurrent licenses are to be in use, the concurrent-offline license will be granted in the concurrent mode.

In this context, the concurrent license is a license for a stand alone server, or a cluster license. The order in which stand alone concurrent licenses and concurrent HAL licenses are granted is not specified.

Managing the Time Zone

Windows

To display the correct time in administration tool panels:

1. For Windows Server 2003 and Window XP, click **Start** > **Control Panel** > **System** > **Advanced** > **Environment Variables**.
2. For Windows Vista, click **Control Panel** > **System** > **Advanced system settings** > **Environment Variables**.
3. In the System Variables box, click **New** to open the New System Variable dialog. In the **Variable Name** entry field, type **TZ**.
4. In the **Variable Value** entry field, type:

   XXXsnnYYY

   where:

   XXX **and** *YYY*
       Characters you choose to remind you of the setting

   s   A sign: + or -

   nn   The difference in hours from Greenwich mean time (GMT). A positive difference means the Greenwich is west of your location; a negative difference means that Greenwich is east of your location. For example, you might specify Italian standard time (one hour east of Greenwich mean time, or GMT+1) as:

   IST-1GMT

5. Click **OK**.

Remember to adjust this time to account for standard time and daylight savings time in different countries.

Monitoring the integrity of the License Use Management Runtime databases

License Use Management Runtime uses a binary format for the database files. To ensure day to day operations run smoothly, it is very important that the integrity of the database files is maintained. Integrity is preserved when data in the database is correct and consistent. As License Use Management Runtime uses the database files dynamically, problems such as the following may cause the database files to become corrupted:

- Hardware failures (media surface errors such as bad sectors on hard disk plates).
- Synchronization errors during writing of data (for example, the loss of electrical power).
- Unusual termination of a License Use Management Runtime server process.

Detecting server crashes

During each startup procedure a check is made to verify if the server has been stopped correctly. If an abnormal termination has been detected, an appropriate message will be logged in the **log server file** file, for example:

```
GRL-2137: Warning: LUM was not terminated properly! (05.02.08 12:30:05).
```

See *Server Log File* for a detailed description of the file.

A crash results in database file corruption. During the startup process, if the server detects a corruption, it will attempt to repair the database. See "Checking database integrity and handling database corruptions" for detailed information on this procedure.

Checking database integrity and handling database corruptions

There are many events and situations when the state of the database files will be checked to ensure database integrity:

- during the startup process
- before the backup procedure is initiated
- before the server's work is terminated
- on demand

There are four types of databases:

- licenses (Licdb/Crpdb)
- users (Usrdb)
- logs (Logdb/Crlog)
- annotation database

Once the corruption of a database file has been discovered the file is automatically copied to an additional, backup folder. The default name of the folder is **corrupted** and its name can be determined in the **i4ls.ini** file. The names of backed up files also contain reparation date and time. The repairing process takes place in the original location of the database and it tries to retrieve the data from the cache and save it as a new database file. Users database and License database are backed up together.

For log databases, a new log database will be created to work with since log databases are created in a cycle with parameters defined in **i4ls.ini** file. For details regarding the log file see Appendix A, "License Use Management Runtime Configuration File," on page 239.

For the server to repair the database automatically the flag in the configuration file **i4ls.ini** has to be set to **AutoRepair=yes**. The possible scenarios are listed below:

- The server will continue working after it has repaired the database (**logAutoRepair=yes**, **licAutoRepair=yes** and **usrAutoRepair=yes**).
- If a log file is corrupted and the server cannot open it, it will create a new one. The old database will be left unchanged (**logAutoRepair=yes**).
- The server will shut itself down if it cannot repair the database (**licAutoRepair=yes** and **usrAutoRepair=yes**)
- The server will shut itself down if the flag is set to **AutoRepair=no** and the database is corrupted.

Each check is logged in server log file for the given server with test results and all details are traced.

It should be noted that the database repair procedure handles database files with extreme care and often databases can only be repaired in case of simple errors.

Checking a database during the startup process

If database corruption is detected during the startup procedure, the server processes will attempt to repair the corrupted database. Based on the configuration settings in the **i4ls.ini** file the server will either repair the failure or it will exit. See Appendix A, "License Use Management Runtime Configuration File," on page 239 for details of the configuration.

Checking a database before the backup procedure

To be sure database is correct and to avoid backing up a database which is corrupted, each database needs to be checked and repaired if necessary just before a server backup procedure starts. If a database corruption is detected before the backup starts, the server process will attempt to obtain data from the server memory to repair the database files. For details regarding the backup procedure see "Using the Built-In Backup and Recovery Procedure" on page 199.

Checking a database before the server's work is terminated

If database corruption is detected before the server's work is terminated, the server process will attempt to obtain data from the server memory to repair the database files.

Checking a database on demand

The current status of the database can be checked by issuing the **i4blt** command:

i4blt -ld
> You may use this command during the server work to check the integrity of the databases.

i4blt -lr
> You may use this command to check the database and repair it only if needed. However, database corruption occurs very rarely during day-to-day operation and this command should be used sparingly.

For example, use these commands to display a database integrity monitoring report for two servers louise and infdev.

The output of i4blt –ld has the following format:

Possible values of status text are:

- OK – database is not corrupted or database was corrupted but the server repaired it
- Corrupted – something is wrong with a database
- N/A – the operation cannot be performed for a database (for example server does not support DIM or it does not use this type of database)
- IO error – there is a problem with IO operations

```
=========================================================================
===       D a t a b a s e   I n t e g r i t y   M o n i t o r i n g      ===
=========================================================================

  Server Name:    ip:louise.rome.tivoli.com
  Target ID:      2f05e220
  Target Type:    Windows Vista
  Server Type:    Network

      Licenses Database: Ok
      User Database:     Ok
      Log Database:      Ok


=========================================================================

  Server Name:    ip:infdev.kraklab.pl.ibm.com
  Target ID:      23f07e330
  Target Type:    AIX
  Server Type:    Network

      Licenses Database: Ok
      User Database:     Corrupted
      Log Database:      Ok


=========================================================================

              ==========================================
              === End of Database Integrity Monitoring ===
              ==========================================
```

The output of i4blt –lr has the following format:

Possible values of status text are:

- OK – database was not corrupted
- Corrupted – database was corrupted and the repairing failed
- N/A – the operation cannot be performed for a database (for example server does not support DIM)
- IO error – there is a problem with IO operations

```
=========================================================================
===       D a t a b a s e   I n t e g r i t y   R e p a i r i n g      ===
=========================================================================

  Server Name:    ip:infdev.kraklab.pl.ibm.com
  Target ID:      23f07e330
  Target Type:    AIX
  Server Type:    Network

      Licenses Database: Ok
      User Database:     Ok
      Log Database:      Ok


=========================================================================

              ==========================================
              === End of Database Integrity Repairing ===
              ==========================================
```

Note: The only option which can be used with both commands is **-n**.

Tthe exact syntax of both commands can be found in Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Using the Built-In Backup and Recovery Procedure

ALL

Because the breakdown of license servers may have a potentially severe impact on production, it is important to ensure you have a backup of your database files to ensure you can recover.

Automatic Backup Procedure

License Use Management Runtime fix pack 12 performs an automatic periodic backup on license servers by copying all files and databases to the following directories:

- `/tmp` on UNIX and AIX
- `c:\ifor\ls\conf\backup` on Windows

You can choose to get the backup on any other device by changing the **BackupPath** parameter in the configuration file (i4ls.ini). You can set the automatic backup to occur daily, at a certain time, (the default), or weekly, on a certain day, or at every change on the license database, according to the **BackupMode** and **BackupParm** parameters specified in the configuration file.

The mode of the License Use Management Runtime database backup procedure has possible values:

- **daily** . It means that the backup is started at the time specified in the **BackupParm** parameter.
- **weekly**. It means that the backup is started at approximately midnight of the day specified in the **BackupParm** parameter.
- **changes**. It means that the backup is made each time the database is changed, when an object is added or deleted.
- **none**. It means that the backup is not started.

You can also disable the automatic backup procedure by setting the **BackupMode** parameter to **none**.

For detailed information on the configuration file (i4ls.ini) see *Appendix A.*Appendix A, "License Use Management Runtime Configuration File," on page 239.

Be sure that the **BackupMode** and **BackupParm** parameters have the same value on all servers in the licensing environment.

The objects listed in:

- `/var/lum/scripts/db_back.sh` on UNIX
- `/var/ifor/scripts/db_back.sh` on AIX
- `c:\ifor\ls\conf\db_back.cmd` on Windows

are backed up if found.

Manual Backup

You can run the backup procedure manually by running:

- /var/lum/scripts/db_back.sh on UNIX
- /var/ifor/scripts/db_back.sh on AIX

On Windows 2000, Windows XP, and Windows Server 2003 use the ntbackup tool ntbackup.exe to perform backup and recovery.

On Windows Vista use Windows Backup tool sdclt.exe.

The following files should be included in the backup:

- c:\Documents and Settings\All Users\Application Data\IBM\LUM\nodelock
- c:\Documents and Settings\All Users\Application Data\IBM\LUM\i4ls.ini
- c:\IFOR\LS\CONF\licdb.dat
- c:\IFOR\LS\CONF\licdb.idx
- c:\IFOR\LS\CONF\logdb01.dat
- c:\IFOR\LS\CONF\logdb01.idx
- c:\IFOR\LS\CONF\logdb00_.da
- c:\IFOR\LS\CONF\logdb00_.idx
- c:\IFOR\LS\CONF\usrdb.dat
- c:\IFOR\LS\CONF\usrdb.idx

Note: The list contains the file names from the default configuration. If you change the file names from those used in the default configuration, substitute the new file name for the one shown in the list. You can also add additional files as needed, according to your environment.

On a workstation configured only as a network license client, only manual backup is available.

This command copies the files and databases:

- to the backup file: /tmp/iforls_bak_DATE_SERVERNAME on UNIX and AIX
- to a backup diskette: the a: drive on Windows 2000, Windows XP, Windows Server 2003, and Windows Vista

Manual Recovery

To start the recovery procedure in case of corruption, use:

- /var/lum/scripts/db_recov.sh iforls_bak_DATE_SERVERNAME for UNIX
- /var/ifor/scripts/db_recov.sh iforls_bak_DATE_SERVERNAME for AIX
- Ntbackup too for Windows 2000, Windows XP, and Windows Server 2003 or Windows Backup tool for Windows Vista

Managing the Report Log Files

There are many ways in which the License Use Management Runtime server can store information about events that have happened and the errors that have occurred. Below is a summary of the three mechanisms that collect the information.

1. **Logs** – it is a database in which events are logged, for example a license request, a license release, a license enrollment or a server start. You can

determine what has to be logged by using **i4cfg** tool. You can create reports on the basis of information recorded in the logs.

2. **Server log file** – it is a low level log file: warnings and errors are recorded in it. The intended users of this file are administrators of License Use Management Runtime server.

3. **Traces**. They are used to trace problems to be sent to the support team. A lot of data is written into traces during day-to-day operation, especially the information how the given server processed each request. By default they are turned off - it is not recommended to have traces turned on all the time because it lowers server performance. They can be turned on either by using the **i4blt -T** command or by editing the appropriate entries in the **i4ls.ini** file (the former method is more advisable). See "Using the Server Tracing Service" on page 206.

There are three types of servers in License Use Management Runtime environment:
- Central registry server
- Network license server
- Nodelocked license server

Below you will find the location for each of the server type log files on each operating system:
- On UNIX:

  `/var/lum/logdbnn` (network license server)

  `/var/lum/crlognn` (central registry license server)

  `/var/lum/llmlgnn` (nodelocked license server)
- On AIX:

  `/var/ifor/logdbnn` (network license server)

  `/var/ifor/crlognn` (central registry license server)

  `/var/ifor/llmlgnn` (nodelocked license server)
- On Windows:

  `c:\ifor\ls\conf\logdbnn` (network license server)

  `c:\ifor\ls\conf\crlognn` (central registry license server)

  `c:\ifor\ls\conf\llmlgnn` (nodelocked license server)

The names of the current log files end with an underscore.

The files `logdbnn`, `crlognn`, and `llmlgnn` contain all the collected License Use Management Runtime events. You can specify which events are to be collected when you configure license servers. See the examples in "Scenario 1: Configuring a Standalone Nodelocked License Server" on page 77, and "Scenario 3: Configuring a Network License Server" on page 84. *nn* can assume values from 00 to 99. When a file is full, the server process moves to the next file (and overwrites the data there). You determine the maximum value for *nn* and the maximum size of each file by setting the **NumberOfLogFile** and **MaxLogFileSize** parameters in the `i4ls.ini` configuration file. When the maximum value for *nn* is reached, License Use Management Runtime wraps to 00. The filled log files are retained so that you can archive them if you wish before the numbering wraps. For details see Appendix A, "License Use Management Runtime Configuration File," on page 239.

The numbering of log files starts from 00. Suppose you have the following files on the workstation:

```
logdb00_
logdb01
logdb02
```

File **logdb00_** is the current one, file **logdb02** is older than the current one, and file **logdb01** is the oldest (will be the next one to be overwritten).

On the machine where the License Use Management Runtime server is run, the space needed for log files is given by the formula `Space needed = (NumberOfLogFile * MaxLogFileSize) *10`, measured in kilobyte units (It should be noted, however, that the real size of the files depends on the operating system disk cashing mechanism and can be much bigger than specified.).

If the log files, marked with the underscore, get too big, do not delete them. You can decrease the size of the current files with the following command:

```
i4blt -x delete_date -n server_name
```

where:

delete_date   Specifies an end date for the delete operation. All log entries recorded up to the delete date are removed from the log files. If you do not specify a date all the entries are deleted.

server_name   Specifies the license server where you want to delete the entries of `logdb`*nn*`_`, `crlog`*nn*`_`, and `llmlg`*nn*`_` if they exist on the server.

When a user displays reports using both CLI and GUI, the i4blt tool creates a temporary file in a default directory. The location of the default directory is dependent on the operating system (for example, `C:\` for Windows platforms, `/tmp` for AIX, or `/var/tmp` for Solaris). The size of the log file depends on the information stored in the LUM database. Use the following formula to calculate the minimum size of log file:

For each server contacted: `Space needed = (NumberOfLogFiles * MaxLogFileSize) *10`, measured in kilobyte units. It should be noted that the size of log files is dependent on operating system caching settings.

Server Log File

This is the file where the server writes warnings and errors. They may help you identify issues that could have impacted the performance of the server. In this file the following information can be recorded:

- the start and stop of the server
- sudden errors
- database corruptions and server crashes
- HAL status changes
- server overload state changes

The names of log files for different server types are:

- I4GDB.log - Central registry
- I4LMD.log - Network server
- I4LLMD.log - Nodelock server

These files are stored in the following locations based on the operating system where the License Use Management Runtime is running:

AIX /var/ifor

HP-UX, Solaris, Linux
 /var/lum

Windows
 :\ifor\ls\conf\

Below are examples of warnings that are recorded in the server log file:

```
GRL-2133: I4LMD service started (05.02.08 10:06:44)

GRL-2137: Warning: LUM was not terminated properly! (05.02.08 12:30:05)

GRL-2137: Warning: Log database (C:\ifor\ls\conf\logdb00_) is corrupted (05.02.
08 12:30:11)

GRL-2134: I4LMD service stopped (13.02.08 10:19:47)
```

Managing Coexistence of NCS and DCE on HP-UX and AIX

If in your network environment you have applications, such as Directory Services and Security (DSS), that use the Distributed Computing Environment (DCE), and configured in namespace binding, read this section.

The default operation of the startup process, as described in this section, will probably be appropriate if both DCE and the License Use Management Runtime subsystems are started at workstation startup and DCE is started first. Check the /etc/inittab file to verify that this is how your workstation is configured.

DCE 3.1 provides a DCE Full Client and a DCE Slim Client. The DCE Full Client installs the DCE daemon. The DCE Slim Client installs the cdsclerk daemon. If you have installed the DCE Full Client, the NCS local location broker will be substituted by the DCE daemon. If you have installed the DCE Slim Client, the cdsclerk daemon is installed.

The NCS local location broker, the DCE daemon, and the cdsclerk daemon use the same TCP/IP port number, 135, which has been assigned to them. Because the NCS local location broker can be replaced by the DCE daemon, when you start services, License Use Management Runtime checks whether DCE is installed before starting the local location broker. If DCE is installed, License Use Management Runtime checks if the DCE daemon is running. If it is not running, License Use Management Runtime waits for 20 seconds (default value), then, if the DCE daemon does not start, the local location broker is started. The local location broker is started if DCE is not installed or if it does not start within the 20 seconds.

The cdsclerk daemon installed by the DCE Slim Client (instead of the DCE daemon) also uses port 135. If the cdsclerk daemon is running, the NCS local location broker cannot start.

If the 20-second delay is either too much or too little for your environment, open the configuration file:

```
i4ls.ini
```

and change the entry:

```
DCEDWAITTIME=
```

in the section:

```
[iFOR/LS NCS-Server]
```

If your workstation is not configured to start the DCE daemon, and therefore you do not want any delay, change this entry to 0.

Tuning the Environment to Manage the Workload

ALL

When a high volume of client/server interactions reaches the server in a short timeframe (for example, 100 license requests per second), the server may not be able to keep up with the volume of workload. The external symptom is that the server seems to hang.

To optimize the license server daemon throughput and better balance its workload, use the information in the following sections:
- "Monitoring Your Environment"
- "Tuning Your Environment" on page 208

Monitoring Your Environment

The Basic License Tool Active License Servers window displays the load percentage and the load threshold percentage for the each license server. The **i4blt -ln** command also displays load percentage and load threshold percentage in the list of active license servers.

Server workload can be directly monitored in five ways:
- "Using the Basic License Tool GUI"
- "Using the Basic License Tool CLI" on page 205
- "Using Standard Reports" on page 206
- "Using the Server Tracing Service" on page 206
- "Using the server log file" on page 207

Using the Basic License Tool GUI

Use the following procedure to monitor the environment using the basic license tool GUI:

1. From the basic license tool main window (Figure 38 on page 102), select:
   Servers → Active License Servers

   The Active license servers window is displayed.

Figure 81. Active license servers window

2. For active license servers, two columns display the load threshold that is currently set and the percentage of server workload:

Load %
> An indicator of the server workload, in the range 0–100%.

Load Threshold %
> The workload threshold set by the administrator, in the range 1–99%. (see "Setting the Load Threshold" on page 208).

When the value of **Load %** exceeds the **Load Threshold %** value, an alert message is displayed indicating that at least one of the network license servers is overloaded. You can now take action to reduce the load on the license server.

Note: In order to see the most up-to-date data, you must refresh the basic license tool main window.

Using the Basic License Tool CLI

To monitor the environment using the basic license tool CLI, you can use the **i4blt -ln** command to display additional information in the list of active license servers. For example, use the following command to display a list for the server louise:

```
i4blt -ln -n louise
```

The command displays a list of active license servers, an example is shown below:

```
=====================================================================
===                         S e r v e r s                      ===
=====================================================================


  Server Name:    ip:louise.rome.tivoli.com
  Target ID:      2f05e220
  Target Type:    Win/NT-MAC
  Server Type:    Network
  Load:           1
  Load Threshold: 85
  Trace:          none


=====================================================================


                       ==========================
                       ===  End of Server List ===
                       ==========================
```

Note: When the active license server is running versions of License Use Management Runtime previous to Version 4.6.8, or it is a nodelocked license server or a central registry license server, the field **Load** displayed in the output of the command will be shown as `Null or Not Available`. This means that the load percentage is not available. When the active license server is a network server, with a load equal to 0%, the field **Load** will be displayed as `Null or Not Available` but the actual meaning is Null.

Using Standard Reports

Use the following procedure to monitor the environment using standard reports:

1. Ensure that the **Server start/stop/threshold** checkbox is selected in the Configuration Tool Notebook - Log Page (Figure 26 on page 79), or use the **i4cfg-es** command (see "i4cfg - Configuration Tool" on page 167). Setting the logging of server and threshold events causes events that exceed the load threshold or regain the load threshold to be logged as `Event Type = SERVER` and `Event Subtype = Threshold Exceeded` or `Event Subtype = Threshold Regained`. An example report is shown below:

```
-------------------------------------------------------------------------
Sequence      : 695                         Event Type  : SERVER
Time Stamp    : Feb 25 2004 3:09:32 PM    Event Subtype: Threshold Exceeded
Server Name   : Network License Server: ip:louise.rome.tivoli.com
-------------------------------------------------------------------------
Sequence      : 696                         Event Type  : SERVER
Time Stamp    : Feb 25 2004 3:15:46 PM    Event Subtype: Threshold Regained
Server Name   : Network License Server: ip:louise.rome.tivoli.com
-------------------------------------------------------------------------
```

2. Use either the basic license tool GUI ("Generating Reports" on page 106) or the **i4blt** command ("i4blt -r Generate a Report" on page 158) to view the events.

Using the Server Tracing Service

Use this service when you have been asked by IBM Support to provide traces, or you want to further investigate traces. The trace level can be customized at runtime. It can be customized using either of the following:

- "Using the Basic License Tool GUI to Customize the Trace Level"
- "Using the Basic License Tool CLI to Customize the Trace Level" on page 207

You must be logged in with administrator privileges to be able to use of traces.

Trace levels: The trace level can be set to:

- None (CLI value is 0)

  No tracing of activity for the selected license server.

- High (CLI value is 1)

  High level tracing. Server logic or database transactions. This consists of entering/exiting major pieces of code, timestamps, the trace level itself, load threshold (displayed in hexadecimal format), and load percentage.

- Low (CLI value is 2)

  Low level tracing. Network communications. This consists of time stamps, entering/exiting minor pieces of code, and thread identifiers (Windows only).

- Low & High (CLI value is 3)

  Selects both the low level and the high level of tracing. This is the maximum level of information

Using the Basic License Tool GUI to Customize the Trace Level: Use the following procedure to start or stop tracing:

1. From the Active license servers window (Figure 81 on page 205), right-click on the server that you want to set the trace level.
2. From the pop-up menu, select **Tracing**, then the required trace level as shown below (see also "Trace levels" on page 206 for more information on the levels of traces that are available). The example below shows a trace level being changed from Low & High to High.
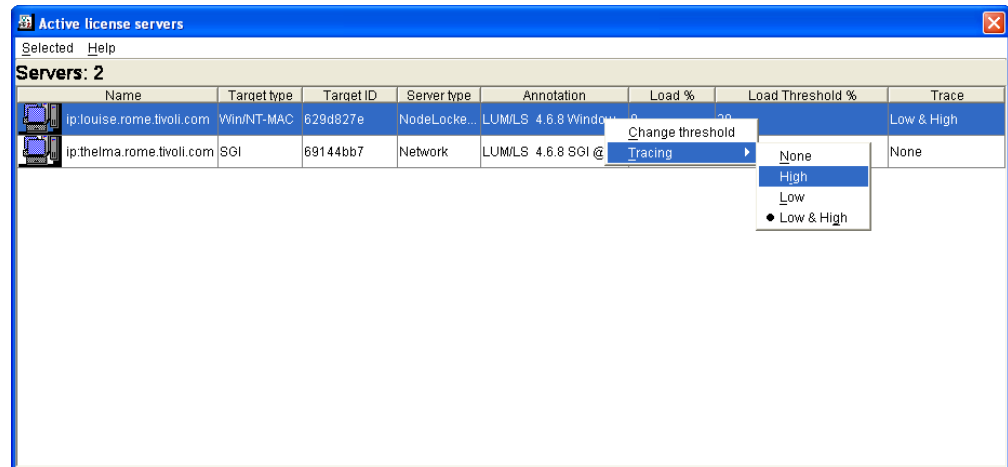
Using the Basic License Tool CLI to Customize the Trace Level: To monitor the environment using the basic license tool CLI, enter the following command:

```
i4blt -T level-of-tracing -n server_name
```

where:

level-of-tracing
  is the value of the level of tracing. See "Trace levels" on page 206.

server_name
  is the name of the network license server for which you want to activate or deactivate the trace. You can activate or deactivate tracing for one server at a time.

If you omit the **-n** *server_name* parameter, the tracing will be activated or deactivated on the local network license server (if there is one). For more information on the **i4blt -T** command, see "i4blt -T Activate or Deactivate Tracing" on page 163.

For example, to activate low & high tracing on the server louise, enter the following:

```
i4blt -T 3 -n louise
```

Using the server log file

When the server overload threshold limit has been overcome License Use Management Runtime writes information about that event into the I4LMD file. The server administrator can check the contents of the file regularly to see if unexpected events have taken place. This task can be automated by associating an action with an event, e.g. in case of an alarming event a notification e-mail can be sent to the administrator or a demon can be started to perform an additional task.

Tuning Your Environment

To determine the maximum number of client sessions that are supported by your servers, and to prevent the servers from going into an overload condition, see the following sections:

- "Displaying Performance Information"
- "Setting the Load Threshold"
- "Determining the Minimum RPC Rate (to be checked after tests have been performed)" on page 209

Displaying Performance Information

To display performance information, use the i4tv test verification tool. Enter the command **i4tv -p**. This command can be used only when License Use Management Runtime has been installed. This tool shows the maximum number of RPCs that are being handled by the server at that instance. It can be used for tuning the system and for troubleshooting performance problems.

To use the **i4tv -p** command, perform the following procedure:

1. Change the configuration of the license server so that it is the only one in the direct binding list.
2. Restart the server
3. Run **i4tv -p** several times on the same computer where you have started the license session in order to have an average of the RPC rate supported by the License Use Management server. An example report is shown below:

```
i4tv -p
Completed license transaction on node  5c35df15 running LUM 4.6.8 AIX
issued RPCs rate: 200 per second

Active License Servers:
    ip:louise.rome.tivoli.com. (IBM/AIX) running LUM 4.6.8 AIX
```

Setting the Load Threshold

Administrators can set the threshold on the license server to prevent it from reaching an overload condition. When the load threshold percentage % is exceeded, no more license requests will be accepted by the server. The server will continue to serve existing clients and HAL cluster synchronization will continue to work, however, no additional clients will be accepted. If the load threshold percentage is set too low, then much of the server's processing power will be unused. You should monitor these values over a period of time and adjust them to suit your needs. You can set the load threshold using either the basic license tool GUI, or the basic license tool CLI.

Setting the Load Threshold Using the Basic License Tool GUI: Use the following procedure to set the load threshold:

1. From the Active license servers window (Figure 81 on page 205), highlight the server that you want to set the load threshold.
2. Right-click to display the pop-up menu.
3. Select **Change threshold**. The Change Load Threshold dialog box is displayed. Type the new threshold level that you require and click **OK**.
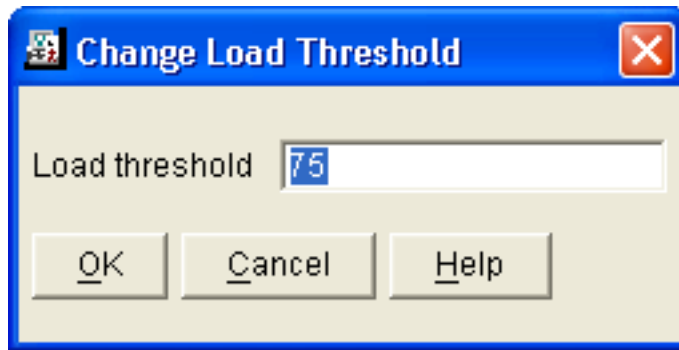
Setting the Load Threshold Using the Basic License Tool CLI: Set the load threshold using the following command:

```
i4blt -S threshold_value -n server_name
```

For example, to set the load threshold on the license server louise to 75%, enter the following:

```
i4blt -S 75 -n louise
```

Determining the Minimum RPC Rate (to be checked after tests have been performed)

The minimum RPC rate required will depend on the number of client sessions that can be simultaneously active. A License Use Management server performing x RPCs per second can support approximately 10 times x client sessions. For example, a server performing 200 RPCs per second can support approximately 2000 client sessions. Over short periods of time the License Use Management server will be able to support more client sessions, but over prolonged periods the 10 times x client sessions value should be used.

However, these values can be influenced by other factors, such as hardware (CPU performance and the amount of installed RAM), and the configuration of the environment.

Conditions which Affect Licensing System Performance

The following information can be useful when tuning your environment:

- Trace Level

  There is a license server performance degradation when the trace level is changed to a more detailed level of tracing. The worst case is when the trace level is changed from None to Low&High. For example, on a 600 MHz Silicon Graphics Octane 2, with 512 MB of RAM, and IRIX 6.5.21, passing from trace level None to trace level Low&High caused a decrease in performance of approximately 70%.

- Heavy administrative tasks

  Heavy administrative tasks can degrade license server performance. Examples of heavy administrative tasks are:

  – Making the license server produce large standard reports while serving client sessions

  – Running frequent batch jobs of i4blt commands

- Level of logging

Producing many large log files causes a degradation of license server performance. The levels of logging which most degrade performance are "license grant" and "license check in". Up to a 30% degradation can be experienced when these two levels of logging are active.

- Hardware configuration

From the tests that IBM has made, results have shown that, for example, on a SUN Ultra-5_10 with 128MB of RAM, the average RPC rate is 210 RPCs per second. Therefore, the license server can support approximately 2100 client sessions. On a SUN Ultra-5_10 with 512MB of RAM, the average RPC rate is 245 RPCs per second. Therefore, the license server can support approximately 2450 client sessions.

- Resource sharing

It is recommended that the license server runs on a dedicated computer. Any other resources running on the license server will degrade its performance.

- Too many client sessions

The performance of the license server will be reduced when the number of client sessions greatly exceeds the suggested 10 x RPCs per second rate and the load on the license server approaches maximum. If this occurs, the administrator can:

  – Change the load threshold (see"Setting the Load Threshold" on page 208)
  – Reduce the number of requesting client sessions
  – Improve the performance of the license server

There is no possibility for the license administrator to revoke active licenses.

The Load Threshold value (corresponding to MaxActivitiesThreshold parameter in the i4ls.ini file) keeps account of the speed of the network connecting servers with clients and the number of requesting client sessions. When the value of this parameter is set too high for the particular licensing system, the server may become easily overloaded. The solution is to reduce the value of the threshold (using the basic license tool, either with the GUI ("Using the Basic License Tool GUI to Customize the Trace Level" on page 206) or the CLI ("Using the Basic License Tool CLI to Customize the Trace Level" on page 207). You can also edit the value of the MaxActivitiesThreshold parameter in the i4ls.ini file, then stop and restart the server. In the case of HAL licensing, this action should prevent the cluster from entering a possible instable condition.

Tuning the Timeout To Detect a Non-Responding server

ALL

Communication problems on the client side may be caused by slow or unreliable network.

The default value of TIMEOUT_FACTOR environment variable can sometimes be too short for work in a distributed environment. You can increase or decrease the timeout using a scaling factor. You can modify the timeout for connections between client and server by changing the TIMEOUT_FACTOR environment variable on the client or server side of the connection.

The default value of TIMEOUT_FACTOR is 1000; values greater than 1000 increase the timeout interval, while values below 1000 decrease it.

The best solution is to try and find values by experimenting with the bandwidth of the network. If the response is longer that 250 milliseconds, you have to increase the value.

License Use Management Runtime in Wide Area Networks

ALL

While we do not recommend to configure License Use Management Runtime to work in Wide Area Network, because of the unpredictability of the network, this scenario can be successfully implemented.

When communication between the client and server is not reliable the timeout period can be changed on both the server and client. This parameter might be changed by setting the TIMEOUT_FACTOR environment variable. See "Tuning the Timeout To Detect a Non-Responding server" on page 210 paragraph for more info.

It is extremely important to configure the firewalls in your network precisely as some packets might get truncated when travelling in your Wide Area Network and this would prevent your licensing environment from working properly.

Managing a Custom Configuration

This section offers advice about custom configurations and their licenses.

Before Requesting a License Upgrade

Before you request an upgrade to your current custom configuration license, double-check the serial number.

Deleting Products or Reducing Numbers

When you upgrade a custom configuration, you can add products and increase the number of seats; however, you can neither delete products nor reduce the number of seats.

Deleting Keys

The initial key is always required. Do not delete it from either the network license server or the nodelock file. You can, however, delete intermediate upgrade keys from a nodelock file. (These intermediate keys are deleted automatically on network license servers.)

Chapter 8. Troubleshooting

This chapter provides suggestions for problem determination, and debugging when you use products managed with License Use Management Runtime.

Troubleshooting the License Use Management Runtime on Windows Vista

If after installing License Use Management Runtime on a Windows Vista workstation you have problems communicating with the server, restart your workstation, even if it is not required by the installer.

Collecting Error Log Data

In order to help IBM assist you in problem determination, you should gather additional information in the form of logs and traces to be able to provide it to an IBM representative when you request support. License Use Management Runtime subsystems and tools can be run in trace mode. For more information see the sections below.

Getting Error Message

This section covers problems that can arise when you try to run License Use management Runtime tools on Windows Vista. If, by starting the License Use Management Runtime tools, the following message appears:

```
Windows cannot access the specified device, path, or file. You may not have the appropriate permis
```

you must switch to the elevated mode and extend your rights to the administrator level. This behavior has been introduced to the Windows Vista operating system as a new concept of User Account Control (UAC) to increase system security. For more information see Windows Vista website.

If the License Use Management Runtime tools do not run when you start your workstation, you might have to extend your rights to the administrator level. For more information on how to get administrator rights, see the section below.

Getting Administrator Rights

To get administrator rights by switching to the elevated mode for License Use Management Runtime on Windows Vista, use one of the following methods:

1. Open **Start** menu.
2. Type the cmd command in the **Search** field.
3. Press **Ctrl+Shift+Enter**. User Account Control window appears.
4. Click **Continue**. A new title with the prefix Administrator appears in the Command Prompt window.

or:

1. Open **Start** > **All Programs** > **Accessories**.
2. Right-click on Command Prompt.
3. Select **Run as administrator**.

Your rights have been extended to the administrator level.

Note: If you want to run the Configuration Tool, type **i4cfg** and **i4cfg - services** for the Service Manager Tool in the Command Prompt window.

License Server Startup Time

On all Windows platforms, the license server might need a lot of time at startup to populate its cache. This might be caused by a high number of licenses enrolled or a high number of transactions.

An environment variable **I4TIMESERV** extends the standard timeout period during which the server is waiting for the notification that the license server is working. This allows you to determine if the license server has been able to start correctly. The default value of **I4TIMESERV** is 30 seconds. Follow the steps below to change the default setting:

1. For Windows XP and Windows Server 2003, open **Start** > **Settings** > **Control Panel** > **System** > **Advanced** > **Environment Variables**.
2. For Windows Vista, open **Control Panel** > **System** > **Advanced System Settings** > **Environment Variables**.
3. In the System Variables field, click **New**.
4. In the New System Variable dialog, type the following information:
   - I4TIMESERV in the Variable Name field.
   - The timeout period (in seconds) that you want the service to wait for the notification, in the Variable Value field. For example 45.
5. Click **OK** and restart your computer.

Troubleshooting Windows installation

If you install the FixPack on Windows by running the msi file, the following message may appear:

```
Another version of this product is already installed. Installation of this version cannot continue.
```

You can either try to install the product by running setup.exe, or force the installation with the msi file, by adding **REINSTALLMODE=vomus REINSTALL=ALL** parameters to the command line.

Checking the Version of License Use Management Runtime

The suggestions in this chapter pertain only to License Use Management Runtime Version 4.x, not to earlier versions of License Use Management Runtime. If you are in doubt about which License Use Management Runtime version is installed on your system check the file:
- /var/lum/VERSION (on UNIX)
- /var/ifor/VERSION (on AIX)
- c:\ifor\ls\conf\runver (on Windows)
- c:\ifor\ls\conf\conver (on Windows)

Checking License Details

Before you proceed, be sure you know the following details about the product that is not starting properly. Check the product enrollment certificate file for all these details.
- Product name (ProductName parameter)
- Product version (ProductVersion parameter)

- Vendor name (VendorName parameter)
- Target type (TargetType parameter)
- Whether the product implements customer-managed or vendor-managed use control (RegistrationLevel parameter; 1= customer-managed, 3= vendor-managed)
- Whether the product is enabled for a custom configuration policy (SerialNumber parameter)
- Whether the password is simple or compound (LicenseStyle parameter= compound, or LicenseStyle parameter= license type if the password is simple)
- License type (LicenseStyle parameter if the password is simple; DerivedLicenseStyle parameter if the password is compound)
- Whether the product is enabled for the hard stop/soft stop policy (SoftStop parameter)
- When the license becomes valid and when it expires (LicenseStartDate and LicenseEndDate parameters)
- Whether the password specifies a license type or a policy introduced in Version 4. The PasswordVersion parameter has one of the following values:
  - 12=new in Version 4.6.8
  - 11=new in Version 4.6.7
  - 10=new in Version 4.6.6
  - 9=new in Version 4.6.4
  - 8=new in Version 4.6
  - 7=new in Version 4.5.5
  - 6=new in Version 4.5.0
  - 5=new in Version 4.0
  - 4=not new in Version 4

This is an example of an enrollment certificate file:

```
4blt -a
-v "''IBM Corporation'' 8499f53d66dd.8d.01.51.32.4c.00.00.00 6tpcewdeyqufi"
-p "'DataWare' '2.1a' v56a7v87iriijz8yzq2ky87hisv9yuhpmmaefasaaaaa "
-S 00102293AE8

[LicenseCertificate]
CheckSum=89E9F6E20FD2395558193CF8436494BA
TimeStamp=1043163603
PasswordVersion=8
VendorName='IBM Corporation'
VendorPassword=6tpcewdeyqufi
VendorID=8499f53d66dd.8d.01.51.32.4c.00.00.00
ProductName=DataWare
ProductID=2222
ProductVersion=2.1a
ProductPassword=v56a7v87iriijz8yzq2ky87hisv9yuhpmmaefasaaaaa
ProductAnnotation=
AdditionalLicenseData=
LicenseStyle=concurrent
LicenseStartDate=01/21/2004
LicenseDuration=12764
LicenseEndDate=12/31/2037
LicenseCount=24
MultiUseRules=User/Accessor
RegistrationLevel=3
TryAndBuy=No
SoftStop=No
Bundle=No
```

```
CustomAttribute1=No
CustomAttribute2=No
CustomAttribute3=No
SubCapacityEligibleProduct=No
TargetType=ANY
TargetTypeName=Open Target
TargetID=ANY
ExtendedTargetType=
ExtendedTargetID=
SerialNumber=00102293AE8
Upgrade=No
InstallProgram=
CapacityType=Processors configured
MaxOfflinePeriod=
DerivedLicenseStyle=
DerivedLicenseStartDate=
DerivedLicenseEndDate=
DerivedLicenseAggregateDuration=
```

The i4blt command at the top of the certificate file is the command that could be used to enroll the password. In the actual enrollment certificate file it would appear on one line; here it is shown on four lines because of space constraints.

Troubleshooting Licenses (All Types)

If a user tries to start a license-enabled product and it does not start, some of the first things to check are:

- First, check the product documentation.
- Check to be sure the license for the application you are running is installed, and, if not, install it. See "Enrolling the Product" on page 103 for information on how to install a license.
- Check that the license you have installed is the correct license for the version of the software you are running.
- Check that the date and time on the workstation are set correctly. Each license has a start date and an end date built in. If the date or time is set incorrectly on the workstation where you are running the product or on a license server, the license may not be recognized as active.
- Check that the time zone and daylight saving time settings are correct.
- Check that the start date of the enrolled license is not later than the current date, and that the license has not expired.

Troubleshooting Nodelocked Licenses

If a workstation with a nodelocked license does not allow an end user to use a license-enabled product, check the product documentation to determine whether the product was enabled for License Use Management Runtime Version 4 or for an earlier version.

Products Enabled for License Use Management Runtime Version 4.x

If the product uses non-runtime-based enabling:

- Check that the enrollment certificate file is in the path specified by the vendor of the product and that its permissions are set so that all users can read it.
- Check that the license is correctly installed in the nodelock file specified by the vendor and that its permissions are set so that all users can read it.

The default location of the nodelock file is:

```
/var/lum/nodelock (on UNIX)
/var/ifor/nodelock (on AIX)
c:\'COMMON_APPDATA'\IBM\LUM\nodelock or c:\ifor\ls\conf\nodelock (on
```
Windows)

where:

COMMON_APPDATA

is the initial part of the path that is returned by the SHGetFolderPath API

An example of a typical path for the nodelock file on a Windows operating system is as follows:

```
c:\Documents and Settings\All Users\Application Data\IBM\LUM\nodelock
```

If the file is not in the default directory, check your product documentation or contact the product vendor.

If the product uses runtime-based enabling:

- Check that the nodelocked license server (i4llmd) is up and running (see "Starting Required Subsystems" on page 227).
- If the request waits for some time and then fails with error message:

```
Inter process communication failure:  check log file i4ipc.out
```

it may be that the maximum wait time for an application to receive a response from the nodelocked license server via Interprocess Communications, as specified in the configuration file, is too short. Edit the configuration file and increase the value of the ReadTimeout parameter, for example to 20:

```
ReadTimeout=20
```

Troubleshooting Network Licenses (All Types)

If a user tries to start a product with a network license and the product does not start, do the following steps. These suggestions apply to concurrent, use-once, reservable/reserved, and per-seat licenses.

- Use the i4tv command from the client workstation to verify the connection to the license server where you have the licenses installed.
  - If the i4tv output shows no active servers, check that the network license server is running on the server workstations where you have the licenses installed.
  - If i4tv does display active servers, check that they include a workstation where licenses for the product are installed. Use i4blt -lp to display the licenses installed on each server. You may need to reconfigure the client to connect to the proper servers.
  - In direct binding, verify that the client is configured to connect to the correct servers.
  - If you are using namespace binding, verify that the client is in the same cell as a server where the licenses are installed. (See "Quick Checklist" on page 225.)
  - If you are using namespace binding, verify that the location brokers are running. See "Starting Required Subsystems" on page 227.
  - If you are using namespace binding, use the lb_admin tool to verify that the network license server where you have the licenses installed is registered to the global location brokers.

– If you are using namespace binding with more than one global location broker, use the drm_admin tool to verify that the global location broker databases are synchronized.

ALL
- If you get the error message:

  `Time disparity is too large`

  check that the date and time on the servers and client are synchronized. If server and client are in different time zones, be sure that time zone and daylight saving time have been set correctly.

- If an enabled application requests more than one license to run, be sure the requested number of licenses is available on one server.

  License Use Management Runtime does not combine licenses installed either on different servers or on the same server but with multiple enrollment actions, to satisfy the same request.

  Similarly, if you received the licenses in a compound password, check that you have distributed, in one single distribution, on one server, at least the number of licenses requested. License Use Management Runtime does not combine licenses distributed either on different servers, or on the same server but with multiple distribution actions, to satisfy the same request.

  For the same reason, if the product is enabled for soft stop, you may see soft stop licenses in use even if there are still some available licenses.

Troubleshooting Reservable and Reserved Licenses

Reservable licenses are enrolled on a network license server. When reserved they are moved to the central registry license server, and when granted they are moved to the nodelocked license server on the client workstation. If a license has been reserved for a user but, when that user tries to use the product, it does not start:

- Check that the central registry license server is up and running (see "Starting Required Subsystems" on page 227).
- Check that the client workstation can reach the central registry.
- Check that the nodelocked license server is up and running at the client (see "Starting Required Subsystems" on page 227).
- Check that the date and time set on the central registry are the same as that set on the network license client. It is possible that, according to the date and time set on the central registry, the license is not yet valid or has expired.
- Double-check the name of the user, group, or node for which licenses are reserved. Be careful with leading and trailing blanks. Note that the domain is part of the node specification.

UNIX
- Check the local host name specification of the client workstation in the `/etc/hosts` file. Make sure it is in the form:

  ip_address hostname_including_domain hostname_without_domain

  For example:

  `69.100.67.70  lab67070.rome.tivoli.com  lab67070`

Troubleshooting Per-Server and Per-Seat Licenses

- If an application with per-server licenses fails to start, be sure the nodelocked license server is running on the workstation where the application server runs (see "Starting Required Subsystems" on page 227).
- If an application with per-seat licenses fails to start:

– Be sure the central registry license server is up and running, and that the nodelocked license server is running on the workstation where the application server runs (see "Starting Required Subsystems" on page 227).

– Be sure the per-seat license is enrolled and that per-seat licensing has been enabled (see "Scenario 3: Switching from Per-Server to Per-Seat Licenses" on page 112).

– Be sure the workstation where the application server runs can reach the central registry.

Troubleshooting Concurrent-Offline Licenses

• If you are unable to reserve an offline license, check with the license Administrator that you have been correctly authorized for that license, and check that your password is correct.

• If you are unable to use an offline license that you reserved, check if the reservation period has expired. If the reservation period has expired, the handback has been automatically made. If you still have problems with reservation and handback, check permissions on the nodelock file. Write permissions should be set for all users of the nodelock file.

• If you are unable to use a license you have reserved, check that the system date on your machine is consistent with the system date on the machine where the reservation was made.

• If you still are unable to work with offline licenses, check that you have enough free space:

– On UNIX platforms, check that /var and /opt have at least 10 Mb free.

– On Windows platforms, check that there is at least 10 Mb free on the disk drive where the operating system is installed.

• A reboot might be needed on Windows in order to refresh the system environment.

• If you have tried everything and still cannot resolve your problem, contact License Use Management support.

• Using a mixture of concurrent licenses and concurrent-offline licenses for the same product is not recommended, because it could lead to some inconsistent data being displayed by the basic license tool (in both the CLI and the GUI).

Troubleshooting Licenses of Customer-Managed Use Products

If you are unable to enroll, update, or distribute licenses for a customer-managed product, a customer-managed use product fails to start, if soft stop does not work, or the high-water mark does not work:

• Be sure that the central registry license server is up and running, and that you have defined only one central registry license server in the direct binding servers list or NCS cell (see "Starting Required Subsystems" on page 227).

• Be sure the workstation where you are working can reach the central registry.

• If you received the licenses in a compound password, make sure you have distributed the licenses. See "Distributing the Licenses" on page 105 for information on how to distribute licenses.

• If soft stop does not work, be sure the soft stop policy is enabled.

Troubleshooting Licenses of Vendor-Managed Use Products

If enrollment of a vendor-managed use product fails, check that the target ID and the target type in the license match the target ID and target type of the workstation where the license is installed. To get the target ID of the workstation, run the target view tool (i4target - Target View Tool) on that workstation.

If there is a mismatch, one of the following is possible:

- The vendor put the wrong target ID or target type into the license.
- You are using the license on the wrong workstation.
- The target ID of your target workstation has changed (see "Troubleshooting the Hardware" on page 229).

Note: The AIX PREFERRED LUM Target ID is only compatible with License Use Management Version 4.6.6 or later. If you need a license to be installed on an older license server, or a nodelock license for an application enabled with an older License Use Management toolkit, then request the license using the LUM Target ID shown second in the output of the **i4target -Ol** command.

Troubleshooting Custom Configuration Licenses

ALL

If, for a custom configuration, you are unable to install a network concurrent license or nodelocked license from the certificate file:

- Check the serial number.
- Check whether another license with the same serial number is already installed. For a concurrent license, use i4blt. For a nodelocked license, use the nodelock file.
  - If you are installing the initial key, no other key can already be installed.
  - If you are installing a replacement key, another key must already be installed.

Troubleshooting HAL Licenses

ALL

An application cannot have its HAL license granted even if it was successfully enrolled. In this case, check the status of the cluster using the basic license tool. The status must be **active**. If the status is not **active**, do one of the following:

- If the status is **Change pending**, wait for the state transition to finish.
- If the status is **inactive**, restart the servers that are listed as **unavailable**.
- If the status is **incomplete**, activate the server members that are listed as **not activated**.

An application cannot have its HAL license granted even if the cluster is active. In this case, using the basic license tool, go to the product details notebook, select the license and right-click on **Show Servers**. If one of the servers in the list has a non-zero "share" but has zero "shared licenses", try to start that server again. If the problem persists delete the license and enroll it again.

An application that had its license granted stops working when one of the cluster members goes down. A possible reason for a failure in the automatic re-granting of the license is that another instance of the same application has started before the first application instance was notified of the change and the total number of available licenses is not enough to satisfy this second request. In this case, install additional licenses.

Troubleshooting License Databases

If the basic license tool displays frequent non-recoverable error messages when enrolling or deleting license certificates, or when trying to clean-up stale licenses, you should refresh the license database. To do so, use the `-clean` option of the license server, as follows:

1. Stop the license services by entering the command:

   `i4cfg -stop`

2. Backup the following files:
   - On UNIX:

     `/var/lum/licdb.dat` and `/var/lum/licdb.idx`
   - On AIX:

     `/var/ifor/licdb.dat` and `/var/ifor/licdb.idx`
   - On Windows:

     `c:\ifor\ls\conf\licdb.dat` and `c:\ifor\ls\conf\licdb.idx`

3. Edit the `i4ls.ini` file and set the **CleanStart** parameter to **yes**.

4. Restart the license services by entering the command:

   `i4cfg -start`

Defining Rules for Multiple-Use Concurrent Licenses

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, concurrent-offline, and per-server licenses.

See "Multiuse Rules" on page 12 for general information about multiuse rules.

Multiuse rules are specified for individual passwords when the software vendor runs **i4lct**, rather than in calls from the product to the license server. This means that rules are applied to individual licenses, rather than to the product itself. The only exception is that the product itself can override the default meaning of a multiuse rule of type **g**, or a combination that includes **g**, to put a vendor-specific rule into effect. The vendor can therefore specify multiple use rules for each customer, without making any changes to the product itself, and without affecting other customers' licenses for the product.

Scenarios

The following scenarios describe how the multiuse rules work when:

- A license with same group rule is installed on the server (10 licenses are available on the server).
- Two clients are in the same group.

When the second request in a scenario is higher than the first, License Use Management Runtime adds the requests, ignoring the multiuse rule.

- Scenario 1
  1. Client1 requests 1 license; License Use Management Runtime shows 1 license in use
  2. Client2 requests 1 license; License Use Management Runtime still shows 1 license in use
- Scenario 2

1. Client1 requests 5 licenses; License Use Management Runtime shows 5 licenses in use
2. Client2 requests 2 licenses; License Use Management Runtime still shows 5 licenses in use
- Scenario 3
  1. Client1 requests 2 licenses; License Use Management Runtime shows 2 licenses in use
  2. Client2 requests 5 licenses; License Use Management Runtime shows 7 licenses in use

Troubleshooting Performance Problems

Read this section for assistance with optimizing performance.

Tuning the Input/Output performance of disks in a License Use Management Runtime server (AIX)

On some older AIX systems with not very fast hard disk drives IO performance problems may occur. If the disk becomes saturated, that is, if the utilization of it is more than 80% (the utilization may be measured with the iostat command, e.g. **iostat 10 10** - it will print 10 measurements in 10 second intervals) some performance tuning is recommended:

1. Disable sequential write-behind mechanism on your system by issuing the following command: **ioo -p -o j2_nPagesPerWriteBehindCluster=0** (the command makes permanent changes, so after reboot the value will be preserved, for testing purposes use the **ioo -o j2_nPagesPerWriteBehindCluster=0** command instead).

   Note: It can only be applied in case of Enhanced Journaled File System (JFS2). If you want to check what file system is used on /var use the **lsfs** command.

2. Add another disk to your system and put /var on it

Basic License Tool Performance

In a network with many users, or when you are creating large reports, if performance is consistently slow when you use the Basic License Tool graphical user interface, consider switching to the command line interface.

Performance in a Direct Binding Environment

In a direct binding environment, careful configuration can help you to optimize performance. See the performance notes under "Planning Direct Binding" on page 34.

Performance in a Namespace Binding Environment

In a namespace binding environment when an NCS cell is running two or more global location brokers (GLBs), the database at each GLB node must be kept synchronized with the others, so that any GLB in the cell can satisfy a location request by a client. Occasionally, a license server is removed or is stopped without being shut down properly, with the result that invalid entries are left in the GLB databases. The invalid entries can introduce significant delays when applications attempt to get licenses, or when running the Basic License Tool.

In such situations, you can clean up the database manually ("Manual Cleanup of GLB Databases"). To schedule automatic periodic cleanup of the databases, see "Periodic Cleanup of GLB Databases" on page 224.

Manual Cleanup of GLB Databases

To do an immediate cleanup by hand, you must remove the invalid entries by using local broker administration (lb_admin) and resynchronize the GLB databases by using GLBDs replicas administration (drm_admin). Both tools are interactive. For more information on how to use these tools, see Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Note: The glbd replica is not available on Linux operating systems.

- To remove the invalid entries, follow these steps:
  1. Start the lb_admin tool at one of the GLB servers. Enter the command:

     `lb_admin`
  2. Set the object to be worked on to be the local location broker:

     `lb_admin: use local`
  3. Enter the clean subcommand to remove any invalid entries:

     `lb_admin: clean`
  4. If prompted to remove entries, type **y**.
  5. Set the object to be worked on to be the global location broker:

     `lb_admin: use global`
  6. Use the clean subcommand to remove any invalid entries:

     `lb_admin: clean`
  7. If prompted to remove entries, type **y**.
  8. Exit lb_admin by using the **quit** subcommand:

     `lb_admin: quit`
- To synchronize the GLB databases at all nodes, follow these steps:
  1. Start the GLBD Replicas Administration tool by entering the following command:

     `drm_admin`
  2. Set the object to be worked on to **global location broker** on your workstation (replace HostName with your actual workstation host name):

     `drm_admin: set -o glb -h ip:HostName`
  3. Synchronize all the GLBs in the cell:

     `drm_admin: merge_all`
  4. If the messages inform you that a host is unreachable, remove it from the global replica list:

     `drm_admin: purgerep ip:HostName`

     where HostName is the host name of this workstation that is no longer acting as a server. If a host workstation is purged from the replica list, it should no longer be running the global location broker process (glbd). If the global location broker needs to be run on this workstation at a later date, configure it and join it to the cell.
  5. Synchronize all the GLBs in the cell:

     `drm_admin: merge_all`
  6. To exit drm_admin, type the quit subcommand:

     `drm_admin: quit`

Periodic Cleanup of GLB Databases

An automatic periodic cleanup of stale entries in the global location broker database is set up by default. If you want to change the settings of the periodic cleanup, edit the i4ls.ini configuration file and set the values of the following parameters:

SelfClean The cleanup enabling flag. Its possible values are **yes** and **no**. The default value is **no**. When a network license server is heavily loaded, its performance could be severely impacted. In such situations, the i4glbcd subsystem may clean up the network license server entry in the global location broker. To prevent this, set the **SelfClean** parameter to **no**.

Frequency The number of minutes between cleanups. The allowed values are 15 to 43 200. The default value is 180.

Timeout The type of timeout. Its possible values are **long** and **short**. The default value is **long**.

Troubleshooting Heavy Server Workloads

When a License Use Management server is overloaded, performance could deteriorate to the point that the server can no longer manage licenses. Use the tools described in "Monitoring Your Environment" on page 204 to determine if the server is being overloaded.

Troubleshooting Network Connections

If connections to license servers seem not to be working properly, use the test verification tool (i4tv) tool to verify that the license servers are up and running, or use the i4blt -ln command to get a list of active servers (network license servers and the central registry). For more information about these commands, see Chapter 6, "License Use Management Runtime Command Line Interface," on page 139.

Troubleshooting Namespace Binding

If the license server uses namespace binding, a failure in NCS can cause License Use Management Runtime to degrade in performance or fail altogether. It may be the case that a License Use Management Runtime problem is actually a problem in the state of NCS.

Under high-volume conditions, if all client workstations are unable to contact a server that runs the global location broker, it is possible that the global location broker database cleaner was unable to contact the server and therefore deregistered it.

It is not necessary to have the database cleaner running on every global location broker server. It is enough to run the database cleaner on one global location broker server in the cell. Choose one that has relatively low-volume traffic, and on the others, do the following to stop the database cleaner:

1. Stop services (i4cfg -stop command).
2. Edit the i4ls.ini file and set **SelfClean=no**.
3. Start services (i4cfg -start command).

Quick Checklist

1. Check that the llbd subsystem is running.
2. Check that the glbd subsystem is running.
3. Check that all the system clocks specify the same time. Use the **setclock** command to synchronize all systems for a short-term solution. It is recommended to implement external time providers and a distributed time service on the network.
4. Check that the /etc/lum/ncs/glb_obj.txt (on UNIX) and /etc/ncs/glb_obj.txt (on AIX) file is the same on all workstations in the NCS cell and that it has at least permission **-rw-r--r 644** (all users can read it).
5. Check that the /etc/lum/ncs/glb_site.txt (on UNIX) and /etc/ncs/glb_site.txt (on AIX) file (if any) points to one or more valid GLB hosts and that it has at least permission **-rw-r--r 644** (all users can read it).
6. Check that the GLB database files still exist. Check particularly for the existence of the files glb.e, and glb.p in the /etc/lum/ncs (on UNIX), and /etc/ncs (on AIX) directories.
7. Check whether llbd was able to create its temporary file /tmp/llbdbase.dat.
8. Check that no more than one default cell has been defined at your location.
9. Use the ncs_test.sh script to test setup and runtime parameters.
10. If you choose to use the default cell, be sure no other user of NCS at your location has created or might create a default cell. Since the default cell always has the same UUID, results would be unpredictable.

License Use Management Runtime Clients Fail to Communicate with Servers

If a client is not communicating with a server properly, it is possible that the client workstation is in a different NCS cell from the license server. To put the client in the same cell as the license server, reconfigure your client workstation. Refer to "Planning Namespace Binding" on page 34.

It is also possible that the client workstation is in a different communications subnetwork from the global location broker (GLB), and cannot contact the GLB. In this case, see "Reaching a Global Location Broker in a Different Subnetwork" on page 36.

License Use Management Runtime Servers Fail to Communicate with Global Location Broker

On License Use Management Runtime servers that run the global location broker, if the UUID stored in the file /etc/lum/ncs/glb_obj.txt (on UNIX) and /etc/ncs/glb_obj.txt (on AIX) is changed, the glbd subsystem continues to use the old UUID even after the glbd subsystem is stopped and restarted. The communication between the glbd subsystem and the i4lmd or i4gdb subsystems will fail.

For an example, observe the following scenario on the server rouse:

```
Cell UUID:
      65d6f8f6471e.02.09.03.01.45.00.00.00
Content of the glb_obj.txt
      657cab79f66f.02.81.23.1c.51.00.00.
```

The i4tv command displays the following error message:

```
i4tv Version 4.6.4  -- LUM Test and Verification Tool
(c) Copyright 1995-2001, IBM Corporation, All Rights Reserved
US Government Users Restricted Rights - Use, duplication or disclosure
```

```
             restricted by GSA ADP Schedule Contract with IBM Corp
             (c) Copyright 1991-1998, Gradient Technologies Inc, All Rights Reserved
             (c) Copyright 1991,1992,1993, Hewlett-Packard Company, All Rights Reserved
             ?(ls_tv) request_license: No servers available for this vendor
             Active License Servers:
             No servers found
             Active Central Registry License Server:
             No Central Registry License Server found
```

This failure will occur even if there is a glbd replica on another node in the cell. The UUID is a 16-byte alphanumeric string and is hard to remember; therefore it is recommended that a copy of the current valid UUID be kept in a secure place.

Note: The glbd replica is not available on Linux operating systems.

The lb_find command is still able to communicate with the glbd subsystem and displays a message similar to the following:

```
# lb_find
sent to broadcast address 9.3.1.255
waiting for replies received response from glbd subsystem at
ip:rouse.itsc.austin.ibm.com(9.3.1.69)
port 1765......
replicatable   ip:rouse.itsc.austin.ibm.com   alternate_2
65d6f8f6471e.02.09.03.01.45.00.00..00
```

If the change to the file glb_obj.txt was made without the administrator's knowledge, the administrator probably will not compare the displayed UUID with the UUID currently stored in the glb_obj.txt file.

The problem can be solved with the following manual steps:

1. Stop all running NCS and License Use Management Runtime subsystems:

   `i4cfg -stop`

2. Remove the following files from the /etc/lum/ncs directory (on UNIX), or the /etc/ncs directory (on AIX):

   ```
   glb_log
   glb.e
   glb.p
   ```

   also remove the file:

   `/tmp/llbdbase.dat`

3. To put the correct UUID into the glb_obj.txt file, reconfigure the license server to start the global location broker or to start a replica from any other existing global location broker in the cell (if any).

4. Restart the NCS and License Use Management Runtime subsystems:

   `i4cfg -start`

Troubleshooting Direct Binding

ALL

If servers and clients are not communicating correctly in a direct binding environment (the i4tv command reports No servers found), check that exactly the same server names and direct binding port numbers were configured for each client, each network license server, and the central registry license server.

For example, if you have a server called *louise* running the network license server and the central registry license server, and clients connected to *louise*, the direct binding configuration for each workstation, including *louise* itself, must contain the entries:

```
ip:louise[10999]
ip:louise[1515]
```

The port numbers must match the values of the **ipPort** and **ipGDBPort** parameters in the configuration file of louise. *louise* and its clients all must have TCP/IP installed.

Troubleshooting License Use Management Runtime Servers

This section covers problems that could arise if License Use Management Runtime and NCS subsystems are not started, or go down.

Starting Required Subsystems

When a license-enabled product fails to start, the problem may be that a required License Use Management Runtime or NCS subsystem is not running.

To get a list of the License Use Management Runtime and NCS subsystems that are running on a workstation, use the `i4cfg -list` command. The names of the subsystems are shown in Table 11. For an overview of which license servers are required for each license type, see Table 8. In a namespace binding environment, the local location broker is required on every network license server and the central registry license server. The global location broker is required on one license server, and the global location broker database cleaner is required on one license server.

Table 15. License Use Management Runtime and NCS Subsystems

Subsystem	Name
Nodelocked License Server	i4llmd
Network License Server	i4lmd
Central Registry License Server	i4gdb
Local Location Broker	llbd
Global Location Broker	glbd
Global Location Broker Database Cleaner	i4glbcd

To start the subsystems, use the `i4cfg -start` command to start all subsystems configured on a workstation.

If the subsystem fails to start when you issue the command, check the error messages in the i4ls.log, i4lmd.log, l4llmd.log, and l4gdb.log files in the `/var/lum` directory on UNIX and `/var/ifor/` directory on AIX, or the glb_log file in the `/etc/ncs` directory (on AIX), and `/etc/lum/ncs` (on UNIX).

Automatic Startup of Subsystems

If the License Use Management Runtime and NCS subsystems do not start automatically when you start your workstation, do the following:

1. Login with root authority.

2. Check if the file /etc/i4ls.rc exists. If not, create a text file called
   /etc/i4ls.rc containing the following string:

   `"i4cfg -start -nopause"`

3. Check the permission on the file /etc/i4ls.rc. If it is different from **744**,
   change it by running the following command:

   `chmod 744 /etc/i4ls.rc`

4. Check that the i4ls entry is in the inittab file, by running the following
   command:

   `lsitab -a`

   If not, add it by running the following command:

   `mkitab i4ls:2:wait:"/etc/i4ls.rc > /dev/console 2>&1 #Start i4ls"`

Starting Required Processes

Windows

When a license-enabled product fails to start, the problem may be that a required
License Use Management Runtime process is not running.

To get a list of the License Use Management Runtime processes that are running
on a workstation, use the `i4cfg -list` command or double-click the **Service
Manager Tool** icon in the License Use Runtime folder.

For an overview of which license servers are required for each license type, see
Table 16.

Table 16. License Use Management Runtime Processes

Name	Process
Nodelocked License Server	i4llmd
Network License Server	i4lmd
Central Registry License Server	i4gdb

To start the processes, select **Start** from the **Service** pulldown menu.

If the process fails to start when you issue the command, check the error messages
in the i4ls.log, i4lmd.log, i4llmd.log, and i4gdb.log files in the *x*:\ifor\ls\conf
directory, where *x* is the installation directory.

Restart and Recovery

ALL

- If a network license server, a nodelocked license server, or the central registry
  license server goes down, a record of users who currently have licenses is kept
  on disk. When the server is restarted, the record is reinstated and the licenses
  are still assigned to those users.

  If you want a cold start (that is, if you want the server to restart as if it had
  granted no licenses before going down), use the **-c** parameter on the command
  used to restart the server (i4lmd, i4llmd, or i4gdb, all described in Chapter 6,
  "License Use Management Runtime Command Line Interface," on page 139). To
  change the default permanently to cold start, edit this parameter of the i4ls.ini
  file:

  `ColdStart=yes`

  and then restart services (`i4cfg -start`).

Cold start is not possible for reserved and per-seat licenses.

- If the client workstation goes down or the network fails, the licenses it was using become *stale* (after a check period expires, if the application is enabled using concurrent access or reservable licenses, and the application is programmed to check in with the server after a specified check period). In this case, the licenses are available to be granted to other clients. Note that those licenses will still be displayed as in use until you perform the **Clean up stale licenses** function or until a license is newly requested and none is available, in which case the server does its own cleanup of stale licenses. The client behavior depends on the software product that is in use.

Troubleshooting the Communication Protocol

If you have communication problems, first verify that the TCP/IP protocol is installed on the workstation and configured properly.

Following is a brief checklist to help you make sure your TCP/IP system is working correctly:

1. Check that the TCP/IP system is up and running.
2. Check whether IP addresses or network interfaces have been changed.
3. Check whether normal TCP/IP communications are working between the nodes you want to be connected (for example, using `ping` or FTP).
4. Make sure your routing setup definition is valid.

   The netstat command shows the local definition. To see the hubs, use the `route print` command.
5. Is name resolution working?

   Name resolution is very often the reason for long startup times or many sorts of problems in large networks. Use the DNS (Domain Name System), and spend some time developing a good layout.
6. Is the MTU (Maximum Transmission Unit - Internet protocols) size equal on all hosts?
7. NCS and License Use Management Runtime are based on universal datagram protocol (UDP). In a very highly loaded network, UDP connections may receive time-outs before data is delivered. This is normal behavior; you need to reduce the total network load.
8. If your License Use Management Runtime server is on a workstation configured with the loopback facility of TCP/IP, when you use the `i4blt -a` command to enroll a product on the server, you must specify:

   `-n ip:localhost`

   in the `i4blt` command.

Troubleshooting the Hardware

The following is a brief checklist to help you make sure your hardware is working correctly.

1. If you get the error message:

   `Invalid target ID`

   - On UNIX and AIX: check that the CPU planar of your License Use Management Runtime server has not changed.

Vendor-managed licenses on the License Use Management Runtime server may be tied to the CPU ID, which changes after a CPU planar swap.

- On Windows: check that your target ID has not changed

2. Check that the cables are still where they should be.

3. Check whether you have reached the Ethernet length limitations on your LAN.

4. Check whether a security feature has been enabled on a router.

   Some routers allow enabling of security features. It is possible to block certain TCP/IP ports.

   In direct binding, the ports are predefined in the configuration file.

   If you are in a UNIX or AIX environment and are using namespace binding, the llbd program runs on port 135. The glbd, i4lmd, and i4gdb programs use runtime-assigned ports whose port numbers are greater than 1024. In direct binding, the ports are predefined in the configuration file.

5. Check whether any adapters or other network definitions have been changed. Because the NCS definition and database files are linked to network addresses, changes may lead to connection errors.

Running Nodelocked Applications and the Basic License Tool on Windows 2000 Server Terminal Services

Windows

To run nodelocked applications and the basic license tool on a Windows 2000 Terminal Services system, follow these steps:

1. Select **Start** --> **Administrative Tools** --> **Terminal Services Configuration**.

2. Check the values of the settings shown in Table 17.

Table 17. Windows 2000 Terminal Services Configuration for Nodelocked Applications

Setting	Value
Terminal Server Mode	Application Server
Permission Compatibility	Terminal Server 4.0 User

3. Double-click any incorrect setting, to open the setting for editing.

4. Change the value of the setting. Ignore any warning message about License Use Management Runtime working.

5. Save and close the file.

6. When running the basic license tool, add the system host name to the License Use Management direct binding list of nodelocked license servers.

7. Run the **reglum.cmd** script (located in the `Windows/Application Compatibility Scripts/Install` directory) to register DLLs.

Running Nodelocked Applications and the Basic License Tool on Windows XP Terminal Services

On a Windows XP system, when the Nodelocked License Server is configured from a fast-switched user session, you cannot run nodelocked applications or the Basic License Tool.

Running Nodelocked Applications and the Basic License Tool on Windows 2003 Terminal Services

To run nodelocked applications and the basic license tool from a Remote Desktop connection to a Windows 2003 Server you must install the Terminal Server component on the Windows 2003 Server machine. The Remote Desktop for Administration mode, that is installed by default on any Windows 2003 Server, is not suitable to run LUM nodelocked applications and the basic license tool. You have also to run the **reglum.cmd** script (located in the `Windows/Application Compatibility Scripts/Install` directory) to register DLLs.

Troubleshooting Installation of License Use Management Runtime on Windows XP

If, after installing License Use Management Runtime on a Windows XP workstation, you have problems communicating with the server, reboot your workstation, even if it is not required by the installer.

Troubleshooting Installation of License Use Management Runtime on Solaris

Troubleshooting Installation on Solaris

If there is an install user on the system, this user's UID must be changed from its current value to 0. To do it, first backup the `/etc/passwd` file, then edit that file, change the third column from the current value to 0. For example, there may be an entry such as:

```
install:x:1001:1001:Install user:/:
```

Change it to:

```
install:x:0:1001:Install user:/:
```

Save the changes to `/etc/passwd`. If an install user does NOT exist in the system, you must create one, and give it the UID of 0. To do this, first backup `/etc/passwd`, then edit that file, and add the following:

```
install:x:0:1:Install user:/:
```

Save the changes to `/etc/passwd`. Once the install user has been either created or modified, re-run all installations. After License Use Management is installed, revert the changes to `/etc/passwd`.

Know issue with memory leaks on Solaris

The problem may occur on License Use Management Runtime Server running for a longer period of time (e.g. a few hours). The situation may occur on Solaris 8, 9 or 10 early releases. In particular, the problem has been identified on Solaris 10 OS released in 2004 with the `/usr/lib/libiostream.so.1` library version seen as RELEASE VERSION Sun C++ 5.7 s10_74L1 2004/12/22. Please run the following command to check the version of /usr/lib/libiostream.so.1:

```
/usr/ccs/bin/what /usr/lib/libiostream.so.1 | grep RELEASE
```

If a huge memory consumption is observed (up to the whole OS memory) for License Use Management Runtime **i4lmd** server process, the latest **Shared library**

patch for C++ has to be installed. The patches for each Solaris OS 8, 9, 10 releases for both x86 and sparc architectures for both 32 and 64bit can be downloaded directly from the following website:

http://developers.sun.com/sunstudio/downloads/patches/ss11_patches.html

You might also do it in a different way:
1. Go to: `http://developers.sun.com/sunstudio`.
2. Choose the **Downloads** tab.
3. In the second navigation bar below choose the **Patches** tab.
4. From the list choose **Sun Studio 11**.
5. On the new web page that opens, choose the **Shared library patch for C++** for a given Solaris OS version 8, 9 or 10, for a given x86 or SPARC architecture, for 32 or 64-bit.

Collecting Error Log Data

In order to help IBM assist you in problem determination, you should gather additional information to send to your IBM representative when you request support. License Use Management Runtime subsystems and tools can be run in trace mode as explained in the following sections.

Running Servers in Trace Mode

To run License Use Management Runtime Servers in trace mode, see the section "Using the Server Tracing Service" on page 206.

Viewing the trace data on UNIX and AIX:

The trace files produced by the Server Tracing Service are as follows:
- For the network license server, the files are named `i4lmd.trc*`, where * is a number from 1 to 99 with a wrap around mechanism. The current trace file has an underscore appended (for example, `i4lmd.trc44_`).
- For the nodelocked license server, the files are named `i4llmd.err` and `i4llmd.out`.
- For the central registry server, the files are named `i4gdb.err` and `i4gdb.out`.

By default, all of the trace files are located in the `/var/ifor` directory on AIX, or the `/var/lum` directory on UNIX. You can specify a different path using the **DebugToPath** tag of the `i4ls.ini` file.

Viewing the trace data on Windows using the GUI:
1. In the **Service Manager Tool** window, select **Stop** from the **Service** pulldown menu.
2. Select the line corresponding to a specific process in the list.
3. Select **View** trace from the **Selected** pulldown menu. The **View trace** window is displayed. It shows the trace messages.

The output trace information is collected in the files with a *.trc* extension.

See "Server Log File" on page 202 for information about managing the trace files produced by the Server Tracing Service.

Running Enabled Applications in Trace Mode

To run enabled applications in trace mode, set the following environmental variables:

- On UNIX and AIX:

  ```
  export I4LIB_VERB=Yes
  export IFOR_LT_DEBUG=Yes
  export I4_IPC_DEBUG=4
  export RPC_DEBUG=Y
  ```
- On Windows:

  ```
  set I4LIB_VERB=Yes
  set IFOR_LT_DEBUG=Yes
  set RPC_DEBUG=Yes
  set I4_IPC_DEBUG=4
  ```

  Then start the enabled application from the same window.

Trace messages will be displayed in the same window where you have set the variable and from which you run the application. You can redirect the output of the trace in a log file by setting the environment variable as follows:

- `export I4LOG_PATH=name_file` on AIX and UNIX
- `set I4LOG_PATH=name_file` on Windows

Running Tools in Trace Mode

To run tools in trace mode use the flag **-B** as the first option when you invoke i4blt and i4cfg. The trace records will be displayed in the window where you run the tools.

Set Windows Services Trace Mode

The environment variable I4_TRACE_SERV enables the tracing of the start and stop phases of Windows services on License Use Management servers. To set the I4_TRACE_SERV variable, follow these steps:

1. For Windows Server 2003 and Windows XP, open **Start** > **Settings** > **Control Panel** > **System** > **Advanced** > **Environment Variables**.
2. For Windows Vista, open **Control Panel** > **System** > **Advanced system settings** > **Environment Variables**.

   The Environment Variables dialog is displayed.
3. In the System Variables box, click **New**. In the New System Variable dialog, add the following information:
   a. In the **Variable Name** field, type:

      ```
      I4_TRACE_SERV
      ```
   b. In the **Variable Value** field, type **Y**.
   c. Click **OK**.

      The Environment Variables dialog is redisplayed.
4. Click **OK**.
5. Reboot the computer.

The output I4_TRACE_SERV logfile is stored in: *I4_INSTALL_DRIVE*:\IFOR\LS\CONF\ i4logservices.txt, where *I4_INSTALL_DRIVE* is the drive where License Use Management Runtime was installed.

Collecting Other Data on UNIX

UNIX

Other information concerning License Use Management Runtime servers is automatically collected by the license server subsystems (i4llmd, i4lmd, and i4gdb) and by the global location broker (glbd). This data is stored in the following files:

```
/etc/lum/ncs/glb_log
/var/lum/i4ls.log
```

Most of the messages you find in these files and the related return codes are not documented.

Also provide IBM with:

- /var/lum/i4ls.ini(the configuration file)
- /var/lum/user_file(the user file)
- /etc/lum/ncs/glb_obj.txt (must always be present when the workstation is part of a non-default NCS cell. Its content must be the uuid of the cell this workstation belongs to, the same as the NCSCell keyword in the i4ls.ini file.)
- /etc/lum/ncs/glb_site.txt (if any; a list of servers running the global location broker that this server can reach.)
- License Use Management Runtime Databases
  - License Databases

    ```
    /var/lum/licdb.dat
    /var/lum/licdb.idx
    /var/lum/llmdb.dat
    /var/lum/llmdb.idx
    /var/lum/crpdb.dat
    /var/lum/crpdb.idx
    /var/lum/usrdb.dat
    /var/lum/usrdb.idx
    /var/lum/adddb.dat
    /var/lum/adddb.idx
    ```

  - Log Databases

    ```
    /var/lum/logdbnn_.dat
    /var/lum/logdbnn_.idx
    /var/lum/llmlgnn_.dat
    /var/lum/llmlgnn_.idx
    /var/lum/crlognn_.dat
    /var/lum/crlognn_.idx
    ```

Collecting Other Data on AIX

AIX

Other information concerning License Use Management Runtime servers is automatically collected by the global location broker (glbd) and by the license server subsystems (i4llmd, i4lmd, and i4gdb). This data is stored in the following files:

```
/etc/ncs/glb_log
/var/ifor/i4ls.log
```

Most of the messages you find in these files and the related return codes are not documented.

Other files you need to provide are:

- /var/ifor/i4ls.ini (the configuration file)
- /var/ifor/user_file (the user file)
- /etc/ncs/glb_obj.txt (must be always present when the workstation is part of a non-default NCS cell. Its content must be the uuid of the cell this workstation belongs to, the same as the NCSCell keyword in the i4ls.ini file.)
- /etc/ncs/glb_site.txt (if any; a list of servers running the global location broker that this server can reach.)
- License Use Management Runtime Databases
  - License Databases
    ```
    /var/ifor/licdb.dat
    /var/ifor/licdb.idx
    /var/ifor/llmdb.dat
    /var/ifor/llmdb.idx
    /var/ifor/crpdb.dat
    /var/ifor/crpdb.idx
    /var/ifor/usrdb.dat
    /var/ifor/usrdb.idx
    /var/ifor/adddb.dat
    /var/ifor/adddb.idx
    ```
  - Log Databases
    ```
    /var/ifor/logdbnn_.dat
    /var/ifor/logdbnn_.idx
    /var/ifor/llmlgnn_.dat
    /var/ifor/llmlgnn_.idx
    /var/ifor/crlognn_.dat
    /var/ifor/crlognn_.idx
    ```

Collecting Other Data on Windows

Windows

Other information concerning License Use Management Runtime servers is automatically collected by the license server processes (i4llmd, i4lmd, and i4gdb). This data is stored in the following files:

```
c:\ifor\ls\conf\i4ls.log
c:\ifor\ls\conf\i4lmd.log
c:\ifor\ls\conf\i4llmd.log
c:\ifor\ls\conf\i4gdb.log
```

Most of the messages you find in these files and the related return codes are not documented.

Also provide IBM with:

- The default location of the configuration file, i4ls.ini, is as follows:

  `c:\'COMMON_APPDATA'\IBM\LUM\i4ls.ini`

  where:

  COMMON_APPDATA
  is the initial part of the path that is returned by the SHGetFolderPath API

  An example of a typical path for the `i4ls.ini` file on a Windows operating system is as follows:

  `c:\Documents and Settings\All Users\Application Data\IBM\LUM\i4ls.ini`

  If the file is not in the default directory, use the **SHGetFolderPath** API to retrieve it.

- `c:\ifor\ls\conf\userfile` (the user file)
- License Use Management Runtime databases
  - License databases

    ```
    c:\ifor\ls\conf\licdb.dat
    c:\ifor\ls\conf\licdb.idx
    c:\ifor\ls\conf\llmdb.dat
    c:\ifor\ls\conf\llmdb.idx
    c:\ifor\ls\conf\crpdb.dat
    c:\ifor\ls\conf\crpdb.idx
    c:\ifor\ls\conf\usrdb.dat
    c:\ifor\ls\conf\usrdb.idx
    c:\ifor\ls\conf\adddb.dat
    c:\ifor\ls\conf\adddb.idx
    ```

  - Log Databases

    ```
    c:\ifor\ls\conf\logdbnn_.dat
    c:\ifor\ls\conf\logdbnn_.idx
    c:\ifor\ls\conf\llmlgnn_.dat
    c:\ifor\ls\conf\llmlgnn_.idx
    c:\ifor\ls\conf\crlognn_.dat
    c:\ifor\ls\conf\crlognn_.idx
    ```

Troubleshooting the Java GUI

Refer to the following sections or to the readme.jgui file if you are having trouble with the License Use Management Runtime Java GUI.

AIX

If you have trouble with the LUM Runtime Java GUI:

- Run i4blt with the -B option to collect debugging information.
- Make sure Java Runtime Environment 1.3.0 or later is installed on your workstation (applicable only to AIX 5.*x*).
- Make sure that the PATH variable is properly set to find the Java interpreter. (To identify the Java interpreter in use, use the **java -version** command.)
- Make sure the DISPLAY environment variable is properly set in the format `'hostname':0`

Solaris

If you have trouble with the LUM Runtime Java GUI:

- Run i4blt with the -B option to collect debugging information.
- Make sure the DISPLAY environment variable is properly set in the format `'hostname':0`

HP-UX

If you have trouble with the LUM Runtime Java GUI:

- Run i4blt with the -B option to collect debugging information.
- Make sure Java Runtime Environment 1.3.0 or later is installed on your workstation.
- Make sure that the PATH variable is properly set to find the Java interpreter. (To identify the Java interpreter in use, use the **java -version** command.)
- Make sure the DISPLAY environment variable is properly set in the format `'hostname':0`

UNIX

If, after migrating from an older Version of License Use Management Runtime 4.6.x to License Use Management Runtime Version 4.6.8.12, you have problems, make sure that you have installed Version 4.6.8.12 of the Java GUI package. Different versions of License Use Management Runtime and the Java GUI cannot coexist on the same machine.

Binary incompatibility on HP-UX

UNIX

If you encounter one of the following problems:

- The ARK Version 4.6.8.12 tools do not start and you get an error message stating that there are unresolved symbols
- The applications enabled with ADK version 4.6.0 or earlier, running in shared mode stop working after installing ARK Version 4.6.8.12

you are using the wrong version of the HP aC++ compiler. To solve this problem download the patches from the following website:

http://www.hp.com/products1/softwareproducts/

Select C++ from the **development software** drop-down menu.

Troubleshooting the i4ls.ini Configuration File

If the i4ls.ini configuration file becomes corrupted, destroyed or lost, or the file is the wrong file, the customer can try to recover the file using the **i4cnvini** command. The i4cnvini.exe file is an executable located in one of the following directories:

AIX   `/usr/opt/ifor/ls/os/aix/bin`
Windows
    `C:\ifor\win\bin`
HP-UX
    `/usr/opt/ifor/ls/os/hp/bin`
Solaris
    `/usr/opt/ifor/ls/os/solaris/bin`
Linux `/usr/opt/ifor/ls/os/linux/bin`

The **i4cnvini** command performs the following tasks:
- Creates a new `i4ls.ini` file if it is missing
- Updates an old `i4ls.ini` file (for example it is called during migration)
- Repairs a corrupted `i4ls.ini` file.

Syntax:
- On AIX and all other UNIX operating systems there are no parameters, the syntax is:

    i4cnvini
- On Windows, the syntax is:

    i4cnvini c: {All | Communication | Runtime}

    where c: is the drive where License Use Management was installed (I4_INSTALL_DRIVE). It is followed by the component that was installed:
    - **All** if both the Communication and the Runtime components were installed
    - **Communication** if only the Communication component was installed
    - **Runtime** if only the Runtime component was installed

Note:

If the **i4cnvini** command fails to recover the `i4ls.ini` file, the suggested procedure is to manually delete the `i4ls.ini` file and then run the **i4cnvini** command again. This recreates a brand new configuration file. The machine must then be configured.

Troubleshooting Windows Server 2003

If you are installing License Use Management Runtime 4.6.8.12 on Windows Server 2003, and an older version of License Use Management Runtime is already installed, you should remove the older version before proceeding. Database migration should be performed manually as described in "Installing after an Uninstallation" on page 66.

Appendix A. License Use Management Runtime Configuration File

This appendix describes the License Use Management Runtime configuration file (i4ls.ini).

The default location of the configuration file, i4ls.ini, is as follows:

HP-UX, Solaris
> `/opt/lum/ls/conf`

AIX   `/var/ifor`

Windows
> `c:\'COMMON_APPDATA'\IBM\LUM\i4ls.ini`

> where:

> **COMMON_APPDATA**
>> is the initial part of the path that is returned by the SHGetFolderPath API

> An example of a typical path for the `i4ls.ini` file on a Windows operating system is as follows:

> `c:\Documents and Settings\All Users\Application Data\IBM\LUM\`
> `i4ls.ini`

If your current License Use Management installation is an upgrade of a previous *License Use Management 4.5.x* installation the default location of the `i4ls.ini` file is `I4_INSTALL_DRIVE`: `\ifor\ls\conf` on Windows 2000, Windows XP, and Windows Server 2003.

You should normally use the configuration tool to configure License Use Management Runtime. In case you have no access to the configuration tool or you just want to change a few parameters in your configuration, the information in this appendix will enable you to modify the parameters by editing the file. Also, some parameters (designated in this appendix) can only be changed by editing the configuration file. The default `i4ls.ini` can be overidden by setting the environment variable *IFOR_CONFIG*.

If a parameter has a default value, it is shown with the parameter name (for example, **BackupMode=daily**).

[iFor/LS Machine-Configuration]

ConfigureAs=client   All platforms
Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Specifies whether the workstation is a server or a client only. Possible values are **server** and **client**.

Transport=tcpip	AIX, HP-UX, Solaris

Specifies the transport protocol used in License Use Management Runtime client server communications. The only possible value is **tcpip**.

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

MachineName=	AIX, HP-UX, Solaris

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

NCSCell=333b91c50000.0d.00.00.87.84.00.00.00	AIX, HP-UX, Solaris

The NCS uuid of the cell this workstation belongs to. If you are configuring the workstation as a network license client only, just specify the NCS cell you want to join. If you are configuring a network license server as a GLB replica, specify the NCS cell you want your server to join. The keyword **CreateFrom** must be set to the ip:servername of any of the replicable GLB replicas of the cell. If you are configuring a network license server as a first GLB, specify the NCS cell uuid of the cell you are creating (the keyword **Create** must be set to **new**). The uuid specified here must be the same as that specified in the `glb_obj.txt` file, if the file exists. In the case of the default cell, there must not be a `glb_obj.txt` file.

UserName=	Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

The name of the user of this workstation.

GroupName=DefaultGroupName	Windows

The name of the group to which the user of this workstation belongs.

DebugProc=no	All platforms

Specifies whether or not the License Use Management Runtime subsystems (UNIX) or processes (Windows) must be started in debug traced mode. Possible values are **yes** and **no**.

DebugNCS=no	All platforms

Specifies whether or not the License Use Management Runtime subsystems must be started in debug traced mode and additional communication-related information collected. For UNIX operating systems it also specifies whether the NCS subsystems must be started in debug traced mode. Possible values are **yes** and **no**.

DebugToFile=no	Windows

Specifies whether or not the License Use Management Runtime processes are started in debug trace mode and the output is redirected to a file. Possible values are **yes** and **no**.

ConcurrentNodelock=No	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Specifies whether or not the License Use Management Runtime concurrent nodelock subsystem (UNIX) or process (Windows) has to be started. Possible values are **yes** and **no**. This keyword is set by the system to Yes when the first concurrent nodelocked password is installed and is set to No when the last concurrent nodelocked password is deleted from the nodelock file.

LogLevel=1	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Level of logging of the concurrent nodelock subsystem:

0   No logging

1   Logging of license add, delete, change events

2   Logging of license not granted events

3   Logging of all level 1 and 2 events plus initializations, requests and releases of licenses. Error conditions are logged too.

LogMsgsMaxNum=1000	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Maximum number of messages logged by the concurrent nodelock subsystem.

LogFile= /var/lum/i4conmgr.log	HP-UX, Solaris

LogFile=/var/ifor/i4conmgr.log	AIX

LogFile=	Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Log file path and name of the concurrent nodelock subsystem log.

CommunVersion=V4R5	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Version of License Use Management Runtime communication subcomponent.

RunMeteringAgent	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

RuntimeVersion=V4R5	All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Version of License Use Management Runtime runtime subcomponent.

NCSSupportVersion=V4R5	AIX, HP-UX, Solaris

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Version of License Use Management Runtime namespace binding support subcomponent.

Communication=yes	All platforms

The workstation is configured to communicate in a network.

NamespaceBindingSupport=yes	AIX, HP-UX, Solaris

Namespace binding support is configured on the workstation.

AdvancedConfiguration=no	All platforms

The user selected **Advanced Configuration** when configuring the workstation.

[iFor/LS GLBD-Configuration]

Create=new	AIX, HP-UX, Solaris

Whether the started GLB is the first one in the cell or one of the subsequent GLB replicas. Possible values are **new** and **replicate**.

CreateFrom=	AIX, HP-UX, Solaris

If you are configuring as a GLB replica, this must be the ip:servername of any of the replicable GLBs of the cell. Also specify the associated NCS cell UUID in the NCS Cell keyword.

Family=ip	AIX, HP-UX, Solaris

Transport protocol used between GLB replicas of the same cell. The only possible value is **ip**.

DefaultCell=yes	AIX, HP-UX, Solaris

Whether or not you are starting the new GLB in a default NCS cell. If you do, make sure you also specified the default UUID in the NCSCell keyword and the new value in the Create keyword, and do not create the /etc/ncs/glb_obj.txt file.

SelfClean=no	AIX, HP-UX, Solaris

Whether you want an automatic periodic cleaning of the location broker's database. This parameter can be changed only by editing the configuration file.

Frequency=180	AIX, HP-UX, Solaris

The frequency in minutes of the automatic periodic cleaning of the location broker's database. This parameter can be changed only by editing the configuration file.

Timeout=long	AIX, HP-UX, Solaris

The timeout used to make sure the license server is alive in the automatic periodic cleaning of the location broker's database. Possible values are **long** and **short**. This parameter can be changed only by editing the configuration file.

[iFor/LS LMD]

BackupMode=daily	All platforms

The mode of the License Use Management Runtime database backup procedure. Possible values are:

daily	The backup is started at the time specified in the **BackupParm** parameter.
weekly	The backup is started at approximately midnight (00:00) of the day specified in the **BackupParm** parameter.
changes	The backup is made each time the database is changed, such as when an object is added or deleted. For operations like **authorisation change** and **license request change** the state of the database will be changed but backup procedure will not be triggered.
none	No backup takes place.

This parameter value must be the same on all servers within your licensing environment. This parameter can be changed only by editing the configuration file.

logAutoRepair=yes	All platforms

The flag which determines whether Log database is to be repaired automatically (after its corruption has been discovered). The default is "yes".

licAutoRepair=no	All platforms

The flag which determines whether License database is to be repaired automatically (after its corruption has been discovered). The default is "no", which means that the server will not start when the database is damaged.

usrAutoRepair=no	All platforms

The flag which determines whether User database is to be repaired automatically (after its corruption has been discovered). The default is "no", which means that the servers will not start when the database is damaged.

badDbBackupDir=C:\ifor\ls\conf\corrupted	All platforms

The path to the backup directory where corrupted databases will be copied just before the repairing process. This example shows the path in Windows operating system.

BackupParm=0	All platforms

If **BackupMode** is **daily**, the hour when the backup occurs (midnight=0). If **BackupMode** is **weekly**, the day of the week when the backup occurs (Sunday=0).

This parameter value must be the same on all servers within your licensing environment. It can be changed only by editing the configuration file.

BackupPath=/tmp BackupPath=*drive*:\ifor\ls\conf\backup	All platforms

The path where the server files and databases are copied during the automatic backup procedure. This parameter can be changed only by editing the configuration file.

NumberOfLogFile=2	All platforms

The number of log files License Use Management Runtime writes. For example, if `logdb` is the log file name, and **NumberOfLogFile** is set to **2**, License Use Management Runtime changes the name to `logdb00_`. When it is full, it starts logging events on `logdb01_`. When this is full, it restarts writing on `logdb00_`. This parameter can be changed only by editing the configuration file.

MaxLogFileSize=10	All platforms

The maximum length of the log files, in tens of kilobytes. After that size is reached, License Use Management Runtime starts writing on another log file. This parameter can be changed only by editing the configuration file.

Note: This functionality is managed by the operating system and it may happen that the real size of the file is bigger, especially when the server is busy.

ValidityPeriod=15	All platforms

	Internal period, in days, to validate per-seat licenses stored on the nodelocked license server against the central registry. This parameter can be changed only by editing the configuration file.
HALFrequency=30	All platforms
	The length, in seconds, of the interval at which servers in a cluster synchronize data among themselves. You can increase this number if you have performance problems, but doing so delays synchronization between members of a cluster.

[iFor/LS NCS-Server]

llbd=no	AIX, HP-UX, Solaris
	Whether you want to start the local location broker subsystem on this server and have the License Use Management Runtime subsystem use it. Possible values are **yes** and **no**. The **llbd** and **glbd** parameters must always be set to the same value.
glbd=no	AIX, HP-UX, Solaris
	Whether you want the network and central registry license servers running on this workstation to register themselves into the global location broker database. Possible values are **yes** and **no**. By specifying no, you disable namespace binding support on this server; it will support only clients locating the server in direct binding mode. The **llbd** and **glbd** parameters must always be set to the same value.
ipPort=1515	All platforms
	The TCP/IP port number that the license server listens to when supporting its clients.
ipGDBPort=10999	All platforms
	The TCP/IP port number that the central registry license server listens to when supporting its clients.
ipHALPort=11999	All platforms
	The TCP/IP port number used for internal communication between servers in a cluster. Change this number only if 11999 is already used for some other purpose. If you change this value, change it on cluster members too.
ipNDLPort=12999	All platforms
	The TCP/IP port number that the nodelocked license server listens to for remote administration.
CleanStart=no	All platforms
	By default, this is set to no. If set to yes, the server behaves as if it has been started with the **-clean** option, that is, it performs a database refresh before starting to listen.
ipAuthPort=	All platforms
	By default, the port number is randomly chosen, but if a specific value is set then the authentication server port is fixed to that value. This parameter can be changed only by editing the configuration file. Setting this parameter to a specific value will disable remote administration of the server.
netbiosPort=115	Windows
	Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The NetBIOS port number the license server listens to when supporting its clients.
netbiosGDBPort=109	Windows
	Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The NetBIOS port number the central registry license server listens to when supporting its clients.
netbiosNDLPort=12999	Windows
	Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The NetBIOS port number the nodelocked license server listens to for remote administration.
ipxPort=1515	Windows
	Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The IPX port number the license server listens to when supporting its clients.
ipxGDBPort=10999	Windows
	Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The IPX port number the central registry license server listens to when supporting its clients.

ipxNDLPort=12999	Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. The IPX port number the nodelocked license server listens to for remote administration.

RunGLBD=no	AIX, HP-UX, Solaris

Whether the global location broker subsystem is to be started on this workstation. Possible values are **yes** and **no**.

RunGDB=no	All platforms

Whether the central registry license server is to be started on this workstation. Possible values are **yes** and **no**.

DisableRemoteAdmin=no	All platforms

Whether the administration of this network license server is to be disabled when using the administration tool started on a different server. Possible values are **yes** and **no**.

DisableRemoteNdlAdmin=yes	All platforms

This determines whether the administration of this nodelocked license server is to be disabled when using the administration tool started on a different server. Possible values are **yes** and **no**. However, it is recommended to set the flag DisableRemoteNdlAdmin to no to optimize the performance of graphical Basic License Tool.

LogAllEvents=no	All platforms

Whether or not all the events are to be logged on the license servers. Possible values are **yes** and **no**.

LogFile=/var/lum/logdb	HP-UX, Solaris

LogFile=/var/ifor/logdb	AIX

LogFile=*drive*:\ifor\ls\conf\log_file	Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Log file path and name of the license server subsystem log (UNIX) or process log (Windows).

LogPath=/var/lum/	HP-UX, Solaris

LogPath=/var/ifor/	AIX

LogPath=*drive*:\ifor\ls\conf	Windows

Log file path of the license server subsystem log (UNIX) or process log (Windows).

DCEDWAITTIME=20	AIX, HP-UX, Solaris

The maximum number of seconds to wait for the dce daemon to start in place of the llbd subsystem. During `i4cfg -start`, if the dce is installed but not running after this number of seconds, the llbd subsystem is started. This parameter can be changed only by editing the configuration file.

RunNDL=yes	All platforms

Whether the nodelocked license server is to be started on this workstation. Possible values are **yes** and **no**.

RunLMD=no	All platforms

Whether the network license server is to be started on this workstation. Possible values are **yes** and **no**.

UseHostTable=no	AIX, HP-UX, Solaris

Change this parameter to **yes** on a workstation with multiple network interfaces if you want to control on which network interface (such as token ring or Ethernet) the network license server and the central registry license server will start. The first entry in the `/etc/hosts` file on the local workstation will be used.

PassiveTime=300	AIX, HP-UX, Solaris

Specifies the length of time in seconds that an activity is to be kept in the server activity pool after the activity has been completed. You can change this parameter only by editing the configuration file.

TraceActivities=no	AIX, HP-UX, Solaris

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime. Specifies whether the server is to write activity-related messages to stdout. You can change this parameter only by editing the configuration file.

MaxActivities=4096   All platforms

Specifies the maximum number of activities that any server can manage. Activity represents some information related to a RPC communication between client and server. For each client, server creates only one activity. You can change this parameter only by editing the configuration file.

MaxActivitiesThreshold=90   All platforms

Specifies the percentage of activities beyond which license requests are rejected. Licenses can still be checked and released beyond this threshold.

This parameter is updated every time the **i4blt -S** command is run.

[iFor/LS Server Logging]

LogGrant=no   All platforms

Log when a license was granted or released. Possible values are **yes** and **no**.

LogCheckin=no   All platforms

Log when a licensed product has sent a check-in call to the server to notify it that the product is running. Possible values are **yes** and **no**.

LogWait=no   All platforms

Log when a license request cannot be satisfied because no licenses are available, and the user is added to a queue. Possible values are **yes** and **no**.

LogVendor=yes   All platforms

Log when a new vendor was added or deleted. Possible values are **yes** and **no**.

LogProduct=yes   All platforms

Log when a new product was registered or deleted. Possible values are **yes** and **no**.

LogTimeout=no   All platforms

Log when the server has canceled the request for a license because the check period has expired. Possible values are **yes** and **no**.

LogErrors=yes   All platforms

Log server errors that do not stop the server, but return a status code and a message. Possible values are **yes** and **no**.

LogVendorMsg=yes   All platforms

Log error messages the vendor inserted in the product. Possible values are **yes** and **no**.

LogSvrStartStopThr=no   All platforms

Log the successful start or stop of the license server and workload threshold-related events. Possible values are **yes** and **no**.

LogSvrStartStop=no   All platforms

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

[iFor/LS NetBIOS -Configuration]

LanAdaptor=0   Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

NCBS=

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

HasOS2Clients=no   Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

[iFor/LS Client]

Threshold_Automatic=0   All platforms

Whether the threshold is automatically checked periodically. Possible values are **0** (no) or **1** (yes).

| Threshold_Frequency=60 | All platforms |

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

| Threshold_Level=80 | All platforms |

The default value of the threshold level used in the Basic License Tool GUI. Used for vendor-managed use products, and for customer-managed use products when no specific threshold value is specified.

| Refresh_Automatic=0 | All platforms |

This determines whether the Basic License Tool is to do a periodic automatic refresh. Possible values are **0** (no) or **1** (yes). However, enabling this option is not recommended.

| Refresh_Frequency=60 | All platforms |

The frequency, in minutes, of the automatic refresh.

| ReadTimeout=4 | All platforms |

The maximum wait time, in seconds, for an application to receive a response from the nodelocked license server via Interprocess Communications. The minimum is 1 and the maximum is 60. You may need to increase this value if performance on your workstation is poor. This parameter can be changed only by editing the configuration file.

| Cleanup_Automatic=No | All platforms |

Whether the Basic License Tool automatically cleans stale licenses. Possible values are **yes** and **no**. You can change this parameter only by editing the configuration file.

[iFor/LS NCS-Client]

| UseDirectBindingOnly=yes | All platforms |

Whether the client licensed applications running on this workstation are to locate the license servers using direct binding only. The administration tool is considered a client application. **yes** is the only possible value on Windows and UNIX.

| GDBServer= | All platforms |

On UNIX, the format is *ip:servername[port]* or not found.

On Windows, the format is *family:servername[port]*, where *family* can be:

ip   Any platform

netbios *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

ipx   *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

Note: This value allows client to bind to Central Registry Server, its format is *ip: [servername] [port]* where:

- *servername* - the hostname of the Central Registry Server

- *port* - the port number of the Central Registry Server

| FilterNDL=no | All platforms |

Whether nodelocked licenses are to be excluded from the set of licenses administered by the Basic License Tool.

| FilterNet=no | All platforms |

Whether network licenses are to be excluded from the set of licenses administered by the Basic License Tool.

| NumDirectBindServers=0 | All platforms |

The number of direct binding servers the client applications are configured to point to directly, using just transport family protocol, name, and port number. Specify the **DirectBindServer** keyword for each server the client points to. If you need to contact the central registry license server, there must also be an entry for it. The default ip port numbers are 1515 for the license server and 10999 for the administration server. Make sure you insert the correct ones if you are not using the defaults.

| DirectBindServer*X*= | All platforms |

Where *X* is a value between 1 and NumDirectBindServer. On UNIX, the format is *ip:servername[port]*.

On Windows, the format is *family:servername[port]*, where *family* can be:

ip   Any platform

netbios *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

ipx   *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

OS2NumServers=0   Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

OS2NetbiosServerX=no   Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

NumDirectBindNDLServers=   All platforms

The number of nodelocked license servers whose licenses can be administered remotely from this workstation.

DirectBindNDLServer*X*=   All platforms

On UNIX, the format is *ip:servername[port]*.

On Windows, the format is *family:servername[port]*, where *family* can be:

ip   Any platform

netbios *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

ipx   *Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.*

OS2NetbiosNDLServerX=no   Windows

Obsolete; supported only for backward compatibility with earlier versions of License Use Management Runtime.

Appendix B. Using the Nodelock File

This appendix explains how to prepare the nodelock file manually and how to use it. You might need to modify the nodelock file for configurations without a nodelocked license server or for backward compatibility.

Nodelock file access rights

Permissions to the nodelock file are set to 0666 on UNIX systems, which means that all users have **read** and **write** permissions to the nodelock file. If a user changes the permissions of the nodelock file manually, they will be preserved.

The situation on Windows machines is different: the nodelock file inherits access rights from the folder in which it is located. Additionally, the file has full permissions for group name, which is set in **i4ls.ini** file.

```
[iFOR/LS Machine-Configuration] GroupName=Users
```

If GroupName is set to the group which does not exist in the operating system, the file will get full access rights characteristic of the Administrators account.

Preparing and using the nodelock file

To prepare and use the nodelock file:

1. On a Windows workstation, go directly to step 2

   On an AIX, HP-UX, or Solaris workstation, log in as root, or use the **su** command.

2. Create or edit the nodelock file:

   - On a Windows workstation:

     ```
     c:\'COMMON_APPDATA'\IBM\LUM\nodelock
     ```

     where:

     COMMON_APPDATA
       is the initial part of the path that is returned by the SHGetFolderPath API

     An example of a typical path for the nodelock file on a Windows operating system is as follows:

     ```
     c:\Documents and Settings\All Users\Application Data\IBM\LUM
     ```

     Note: The files in these directories will override the nodelock file in the ifor\ls\conf directory.

   - On an HP-UX or Solaris workstation:

     ```
     /var/lum/nodelock
     ```

   - On an AIX workstation:

     ```
     /var/ifor/nodelock
     ```

   The format of the nodelock file is:

   ```
   # comment
   vendorID  productPassword  Annotation  version  [serialNumber]
   ```

   where:

   - The first line starts with a comment character, #, and is included for information only. It indicates the product name and license expiry date.

249

- The second line is the product license. Its fields and their content are as follows:

vendorID
> The vendor ID.

productPassword
> The long alphanumeric password that enables the nodelock license.

Annotation
> The annotation field, which is used by the application developer to provide any unique enablement options of the license. This optional field, which is set to null ("") in the example, can contain up to 255 alphanumeric characters.

version
> The version number of the product.

serialNumber
> The serial number of a custom configuration license. This field can contain up to 31 alphanumeric characters.

Initially, this file could have entries similar to the following lines:

```
# nodelock example for the licensed product expires 12/25/2004
543b0f87c093.02.81.87.92.34.00.00.00 gganccupqb5dauxabdw "" "2.0" "85AB2215691"
```

3. To help yourself and others identify the license in the future, because there may be other nodelocked software on the same computer, you should enter a comment above the license. That comment should include the full product name, version, and any expiration date.

4. Double-check the information to ensure that it is the same as that supplied by the vendor.

5. Test the product.

Note: License programs enrolled only in the nodelocked file by an application show a third line starting with #[admin_comment]. The line contains reference information, for example, the name of the vendor, the name of the product, and the expiration timestamp. The information in the #[admin_comment] line may be used for display purposes, and must not be altered.

Appendix C. Operating Systems and License Use Management Support

This appendix identifies the operating systems supported by the various versions of License Use Management:

- Version 4.6.8 (Table 18)
- Version 4.6.8 fix pack 12 (Table 19)

The tables also indicate which versions have been tested and the level of support provided.

Table 18. Operating Systems and License Use Management Version 4.6.8 Support

Operating System	License Use Management Version 4.6.8				
	Application Developer's Toolkit	Runtime	Application Developer's Toolkit 64–Bit	Java GUI	Native Java APIs
AIX 5.1 and 5.2	Yes	Yes	Static link only	Yes	Yes
AIX 5.3	Yes	Yes	Static link only	Yes	Yes
HP-UX 11.0 and 11.1	Yes	Yes	Static link only	Yes	Yes
IRIX 6.5	Yes	Yes	Static link only	Yes	Yes
Red Hat Enterprise Linux 3, and SUSE LINUX Enterprise Server 9, all for Intel x86 platforms only	Yes	Yes		Yes	Yes
Solaris 7, 8, 9 and 10	Yes	Yes	Static link only	Yes	Yes
Windows 2000 and XP	Yes	Yes		Yes	Yes
Windows Server 2003 32–bit	Yes	Yes		Yes	Yes

Table 19. Operating Systems and License Use Management Version 4.6.8 fix pack 12 Support

Operating System	License Use Management Version 4.6.8 fix pack 12				
	Application Developer's Toolkit	Runtime	Application Developer's Toolkit 64–Bit	Java GUI	Native Java APIs
AIX 6.1	Yes	Yes	Static link only	Yes	Yes
AIX 5.3	Yes	Yes	Static link only	Yes	Yes
AIX 5.1, 5.2		Yes		Yes	
HP-UX 11.i v.1 (B.11.11), 11.i v.1.5 (B11.20), 11.i v.1.6 (B.11.22)	Yes	Yes	Static link only	Yes	Yes
HP-UX 11i v2 (PA-RISC)	Yes			Static link only	Yes
Solaris 8 and 9 (SPARC)		Yes		Yes	
Solaris 10 (SPARC)	Yes	Yes	Static link only	Yes	Yes
Solaris 10 (AMD)	Yes		Static link only		Yes
Windows 2000 32-bit		Yes		Yes	
Windows XP and Windows Server 2003 32 and 64–bit	Yes	Yes	Static link only	Yes	Yes
Windows Vista 32 and 64-bit	Yes	Yes	Static link only	No	No

Appendix D. Accessibility

Accessibility is a product feature that helps anyone who has a physical disability, such as restricted mobility or limited vision, to use the software product successfully.

The major accessibility features in License Use Management Runtime include:
- Keyboard accessibility features
- Object information
- Sound and multimedia
- Display
- Documentation

Using License Use Management Runtime, you can:
- Use accessibility technologies such as screen-reader or screen-magnifier software.
- Perform tasks or actions by using only the keyboard.
- Customize display attributes, such as color and font size.

In addition, the License Use Management Runtime product documentation aids accessibility by providing:
- Accessible documentation formats, such as Hypertext Markup Language (HTML) or convertible Portable Document Format (PDF). You can access HTML documents using screen-reader software.
- Alternative text for all graphics so that any documentation user with vision impairments understands the contents of the images.

Keyboard Accessibility Features

License Use Management Runtime supports these keyboard accessibility features:
- Keyboard equivalents for all mouse actions.

  For each software platform, there are standard keys and key combinations for doing common actions. See "Keys Help" on page 255 for a list of the keyboard equivalents so that, if you are unable to use the mouse, you can access all product functions using the keyboard.
- Access to the accessibility features of the operating system.

  License Use Management Runtime does not provide keyboard sequences for the shortcut keys that are reserved for the accessibility features of the operating system. The Windows operating system, UNIX systems (such as AIX, HP-UX, and Solaris), and Linux have a set of accessibility options. These accessibility options allow anyone with a disability to customize system-wide keyboard settings to improve accessibility. For example, you can customize Windows systems using the Windows Accessibility Options on the Control Panel. On UNIX systems, accessibility options are part of the AccessX package.

Java Interface and Object Information

License Use Management Runtime uses a Java-based graphical user interface, which provides object-oriented accessibility features, such as:

- Implementing the Java Accessibility programming interfaces, or adhering to guideline for building custom components when extending the Java Foundation Classes.
- Setting the focus. In an application, access devices, software, and users need to know where to direct the focus. For License Use Management Runtime, use the Tab key to change the focus.
- Setting the name and description on components, icons, and graphics such as using an image to identify programmatic elements. All License Use Management Runtime icons have alternate text for icons and graphics.
- Associating labels with controls, objects, and icons, such as associating the text label with an entry field control. License Use Management Runtime labels all components that you can access with speech-reader software.

Sound and Multimedia

License Use Management Runtime uses the following sound and multimedia accessibility functions:

- An option to display a visual cue for all audio alerts. Typical alerts might be the receipt of new mail, beeps to indicate system errors, or sounds to indicate a change in status. License Use Management Runtime allows you to change your user preference to turn on visual cues for audio alerts.
- Use of the operating system volume setting to adjust the volume for the application.

For the Windows operating system, you control the volume by selecting **Start ▶ Programs ▶ Accessories ▶ Entertainment ▶ Volume Control**.

Display

License Use Management Runtime supports the operating system display settings for color scheme, font size, and high contrast display. License Use Management Runtime supports a variety of color selections.

- On Windows systems, you can customize colors through the Microsoft® Windows Accessibility Settings for High Contrast in the Control Panel. Or you can customize colors through the License Use Management Runtime view options.
- When you customize colors, you can produce a range of contrast levels.
- Color is not the only way that License Use Management Runtime conveys information or indicates an action. License Use Management Runtime uses other attributes, such as bold or italic, in addition to color, to distinguish the text.
- To view a document or an application work area in an enlarged font, you can use various magnifications of the Zoom option in Windows. Using magnifications does not affect the actual document or work.

Documentation

License Use Management Runtime provides documentation in these formats:

- Accessible HTML format. Accessibility tools, such as screen readers, can read directly this format.
- Accessible text. All License Use Management Runtime documentation with graphics has alternate text that describes each illustration and graphic used in the documentation.

Even though License Use Management Runtime provides an accessible HTML format, you can also create accessible Portable Documentation Format (PDF) format. You can convert Adobe® V4.0 PDF files to the accessible HTML format. Use the Adobe PDF-to-HTML conversion program to create accessible PDF formats. For information about converting PDF documents to HTML, refer to the Adobe book *Optimizing Adobe PDF Files for Accessibility*

In addition, License Use Management Runtime provides documentation of all keyboard access methods, such as accelerator keys.

Keys Help

License Use Management Runtime provides documentation on all accessibility features, including keyboard access. In additional to this appendix, you can select **Help ▸ Help Keys** for online help.

Following are lists of keys that are available to you when using License Use Management Runtime. These keys are grouped according to their function.

Note: When you see two keys that are joined by a plus sign (+), hold down the first key while pressing the second key at the same time.

General Keys

Use the following keys to act on a window:

F1	Provide help for the panel.
F3	Exit the panel.
F5	Refresh the panel.
F10 or ALT	Go to or from the action bar.
Arrow keys	Move among choices on the action bar in menus, in dialog windows, or in list boxes.
Underlined letter	Move to a choice on the menu bar or menus.
Alt+F4	Close the window.
Alt+F5	Restore the window (only for supported Windows platforms).
Alt+F7	Move the window (only for supported Windows platforms).
Alt+F8	Size the window (only for supported Windows platforms).
Shift+Esc or Alt+Spacebar	Go to or from the system menu.
PgUp	Scroll the contents of the window up one page.
PgDn	Scroll the contents of the window down one page.
Ctrl+PgUp	Scroll the contents of the window left one page.
Ctrl+PgDn	Scroll the contents of the window right one page.
Esc	Cancel the application menu or the system menu.

Keys for Dialog Window

Use the following keys to act on a dialog window.

Enter	Perform the action indicated on the selected push button.
Spacebar	Set a check box on or off or perform the action indicated on the selected push button.
Tab	Move to the next group of entry fields or check box.
Backspace (left arrow delete)	Delete the character to the left of the cursor in input fields.

End or Ctrl+Right Arrow Key	Move to the end of a field.
Home or Ctrl+Left Arrow Key	Move to the beginning of a field.

Keys for Help

Use the following function keys to get help from the application or to get different types of help within a help window.

F1	Help

Accelerator Keys for the Configuration Tool

You can use accelerator and command keys to navigate through License Use Management Runtime. All accelerator keys have an underlined letter. For example, F is the accelerator for the File menu. In Windows, press the ALT key, then the accelerator key to trigger an action. For example, Alt+F shows the File menu. You can show additional accelerators for bookmarks and task buttons by pressing and holding down the ALT key.

Command keys directly trigger an action and usually make use of the CTRL (Windows) keys. For example, to print, press CTRL+P.

License Use Management Runtime uses accelerator keys. The following sections list the accelerator keys used for the configuration tool. Although the accelerator keys listed are shown as they appear on the menus (both in uppercase and in lowercase), the accelerator keys are not case sensitive.

Configuration Tabs

The following tabs are in the configuration tool:
- Configure As
- User
- Accessibility
- Direct binding
- Start up
- Direct binding ports
- Log

Each tab has its own accelerators.

Configure As Tab: Use the following accelerator keys in the **Configure as** tab:

ALT+C	To select network license client.
ALT+N	To select NodeLocked License Server (NodLS).
ALT+D	To disable remote administration for NodeLocked LS.
ALT+S	To select Network License Server (NetworkLS).
ALT+L	To disable remote administration for Network LS.
ALT+R	To select Central Registry License Server (CrLS).

User Tab: Use the following accelerator key in the **User** tab:

ALT+G   To select Group name.

Accessibility Tab: Use the following accelerator keys in the **Accessibility** tab.

ALT+F   To open the Font Selection dialog.
ALT+L   To open the Color Selection dialog.

Direct Binding Tab: Use the following accelerators in the **Direct binding** tab:

ALT+S   To select the Servers list.
ALT+N   To select the Name field.
ALT+K   To select Network License Server.
ALT+C   To select Central Registry License Server.
ALT+D   To select Nodelocked license server.
ALT+L   To select the Network License Server port field.
ALT+P   To select the Central Registry License Server port field.
ALT+O   To select the Nodelocked license server port field.
ALT+A   To add a server.
ALT+R   To remove a server.

Start Up Tab: Use the following accelerator in the **Start up** tab:

ALT+U   To start the Basic License Tool.

Direct Binding Ports Tab: Use the following accelerators in the **Direct Binding Ports** tab:

ALT+N   To select the Network License Server port.
ALT+C   To select the Central Registry Server port.
ALT+D   To select the NodeLocked License Server port.

Log Tab: Use the following accelerators for the **Log** tab:

ALT+A   To log all events.
ALT+C   To log license check-in events.
ALT+W   To log license wait events.
ALT+T   To log license timeout events.
ALT+E   To log errors.
ALT+G   To log license grant or release events.
ALT+V   To log vendor added or deleted events.
ALT+M   To log vendor message events.
ALT+P   To log product added or deleted events.
ALT+S   To log the successful start or stop of the license server, and when the
               threshold on the workload is exceeded or regained.
ALT+L   To select the log file directory field.

Font Selection Dialog Accelerators

Use the following non-standard accelerators for the Font Selection dialog:

ALT+D   To reset the default font and font size.

Color Selection Dialog Accelerators

Use the following non-standard accelerators for the Color Selection dialog:

ALT+F	To change the foreground color.
ALT+B	To change the background color.
ALT+D	To reset the default background and foreground colors.

Accelerator Keys for the Basic License Tool

You can use accelerator and command keys to navigate through License Use Management Runtime. All accelerator keys have an underlined letter. For example, F is the accelerator for the File menu. In Windows, press the ALT key, then the accelerator key to trigger an action. For example, Alt+F shows the File menu. You can show additional accelerators for bookmarks and task buttons by pressing and holding down the ALT key.

Command keys directly trigger an action and usually make use of the CTRL (Windows) keys. For example, to print, press CTRL+P.

License Use Management Runtime uses accelerator keys. The following sections list the accelerator keys used for the basic license tool. Although the accelerator keys listed are shown as they appear on the menus (both in uppercase and in lowercase), the accelerator keys are not case sensitive.

Main Window Accelerators

Use the following accelerator keys in the main window:

ALT+P	To open the Products menu.
ALT+S	To open the Selected menu.
ALT+E	To open the Edit window.
ALT+R	To open the Servers menu.
ALT+V	To open the View menu.
ALT+O	To open the Options menu.
ALT+H	To open the Help menu.

Products Menu: The Products menu has the following accelerator keys:

N	To open the Enroll Products submenu.
i	To open enrollment for single products.
M	To open enrollment for multiple certificates.
x	To exit from the Basic License Tool.

Selected Menu: The Selected menu has the following accelerator keys:

O	To open the details window.
A	To add a password to a license.
p	To create a report.
D	To distribute licenses.
U	To update licenses.
C	To clean up stale licenses.
R	To remove licenses.
E	To expand a bundle.
b	To collapse a bundle.
v	To remove a bundle.

Edit Menu: The Edit menu has the following accelerator keys:

l	To select all licenses.
D	To deselect all licenses.

Servers Menu: The Servers menu has the following accelerator keys:

A	To view active license servers.
C	To view clusters.

View Menu: The View menu has the following accelerator keys:

o	To sort the licenses.
n	To include servers, vendors, or products.
F	To refresh the view.

Sort Menu: The Sort menu has the following accelerator keys:

d	To sort licenses by product.
y	To sort licenses by vendor.

Options Menu: The Options menu has the following accelerator keys:

i	To open the system settings dialog.
T	To set the toolbar viewing mode submenu.
N	To set the toolbar viewing mode to normal.
F	To set the toolbar viewing mode to floating.
i	To set the toolbar viewing mode to hidden.
F	To open the Font Selection dialog.
C	To open the Color Selection dialog.

Help Menu: The Help menu has the following accelerator keys:

G	To open general help on the Basic License Tool.
K	To open this keys help.
r	To open information about the product version.

Details Window Accelerators

The Details window has the following tabs:

- Product
- Compound
- Concurrent
- Concurrent Users

Product Tab: The **Product** tab has the following accelerators:

N	Product Name
V	Product Version
I	Product ID
a	Vendor Name
D	Vendor ID

Compound Tab: The **Compound** tab has the following accelerators:

L   To open the licenses menu
S   To open the selected menu
H   To open the help menu

Concurrent Tab: The **Concurrent** tab has the following accelerators:

L   To open the Licenses menu.
S   To open the Selected menu.
H   To open the Help menu.

Concurrent Users Tab: The **Concurrent users** tab has the following accelerators:

L   To open the Licenses menu.
S   To open the Selected menu.
H   To open the Help menu.

Font Selection Dialog Accelerators

Use the following non-standard accelerators for the Font Selection dialog:

ALT+D   To reset the default font and font size.
ALT+F   To select the font list.
ALT+S   To select the sizes list
ALT+T   To select the styles list.

Color Selection Dialog Accelerators

Use the following non-standard accelerators for the Color Selection dialog:

ALT+F   To change the foreground color.
ALT+B   To change the background color.
ALT+D   To reset the default background and foreground colors.

Update Licenses Dialog Accelerators

The Update Licenses dialog has the following accelerators:

E   To enable per-seat licensing.
u   To activate customer-managed use.
D   To disable soft stop.
g   To set High Water Mark licenses.
R   To reset High Water Mark licenses.
T   To set the threshold.
l   To select enrolled licenses.

Reserved Licenses Details Window Accelerators

The Reserved Licenses Details window has the following accelerators:

L   To select the licenses.
N   To set the name.
G   To set the group.
d   To set the node ID.
Y   To set the year of license expiration.
M   To set the month of license expiration.

a	To set the day of license expiration.
u	To set the hour of license expiration.
i	To set the minute of license expiration.

Include Dialog Accelerators

The Include dialog has the following accelerators:

ALT+A	To select the **All servers** radio button.
ALT+S	To select the **Select servers** radio button.
ALT+V	To select the **Select servers** button.
ALT+L	To select the **All vendors** radio button.
ALT+T	To select the **Select vendors** radio button.
ALT+N	To select the **Select vendors** button.
ALT+P	To select the **All products** radio button.
ALT+Y	To select the **Currently in use** radio button.
ALT+U	To select the **All users** radio button.
ALT+E	To select the **Select users** radio button.
ALT+R	To select the **Select users** button.

Select Users Dialog Accelerators

The Select Users dialog has the following accelerators:

l	To move to the available users box.
S	To move to the selected users box.
A	To add a user to the selected users box.
d	To add all users to the selected users box.
R	To remove a user from the selected users box.
e	To remove all user from the selected users box.

Define Cluster Members Dialog Accelerators

The Define Cluster Members dialog has the following accelerators:

| A | To add a member to a cluster. |
| R | To remove a member from a cluster. |

Active License Servers Window Accelerators

The Active License Servers window has the following accelerators:

ALT+S	To open the Selected submenu.
ALT+C	To change the threshold value.
ALT+T	To open the Tracing submenu.
ALT+N	To select the None tracing option.
ALT+I	To select the High tracing option.
ALT+L	To select the Low tracing option.
ALT+G	To select the Low&High tracing option.
ALT+H	To open the general help for the Basic License Tool.

Clusters Window Accelerators

The Clusters window has the following accelerators:

ALT+C	To open the Create menu.
ALT+S	To open the Selected menu.
ALT+H	To open the Help menu.

Create Menu: The Create menu has the following accelerators:

ALT+R	To open the Create Clusters dialog.

Selected Menu: The Selected menu has the following accelerators:

ALT+O	To show the selected servers.
ALT+U	To update the selected licenses.
ALT+R	To remove the selected licenses.

Help Menu: The Help menu has the following accelerators:

ALT+G	To show the Basic License Tool general help.

Server Details Window Accelerators

The Server Details window has the following accelerators:

ALT+A	To activate the selected server.
ALT+D	To deactivate the selected server.

Enroll Product Window Accelerators

The Enroll Products window has the following accelerators:

I	To open an import dialog for selecting the product to import.
a	To set the product name.
V	To set the product version.
w	To set the license password.
l	To set the license serial number.
t	To set the license annotation.
N	To set the vendor name.
D	To set the vendor ID.
P	To set the vendor password.

Report Window Accelerators

The Report window has the following accelerators:

A	To select all events.
L	To select license related events.
V	To select vendor message events.
D	To select LDM modification events.
E	To select error events.
p	To select server start and stop events.
F	To select fatal error events.
t	To set the date range for the report.
S	To select a standard report format.
r	To select a license request by product report format.

q	To select a license request by user report format.
y	To select a license use by product report format.
u	To select a license use by user report format.
m	To select a customer-managed user audit report format.

Settings Dialog Accelerators

The Settings dialog has the following accelerators:

D	To set the default threshold default level.
A	To set the automatic refresh option.
F	To set the automatic refresh time in minutes.
R	To select the Refresh now button.
E	To enable threshold logging.
N	To enable nodelocked licenses.
w	To enable network licenses.

Update Offline Period Dialog Accelerators

The Update Offline Period dialog has the following accelerators:

ALT+M	To set the maximum offline period.

Authorizations Dialog Accelerators

The Authorizations dialog has the following accelerators:

ALT+L	To modify the default authorization to All allowed.
ALT+N	To modify the default authorization to All not allowed.
ALT+R	To modify the default authorization to All concurrent allowed.
ALT+E	To modify the password of the default authorization.
ALT+A	To add a new authorization.
ALT+M	To modify an authorization.
ALT+D	To delete an authorization.

Add/Modify Authorization Dialog Accelerators

The Add/Modify Authorization dialog has the following accelerators:

ALT+U	To set the user name.
ALT+G	To set the user group name.
ALT+P	To set the password for the authorized user.
ALT+T	To set the target ID of the machine.
ALT+A	To authorize the use of the license as concurrent-offline.
ALT+N	To deny the use of the license.
ALT+R	To authorize the use of the license as concurrent.

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 that are expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications by 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.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Glossary

A

accessor ID. In License Use Management, a generic string identifying the requestor of a license. Vendors can define their own multiuse rule based on the accessor ID.

ACID. See *application client identifier*.

administrator. In License Use Management, the person who is responsible for setting up the License Use Management Runtime environment. The tasks of the administrator include:

- Installing and configuring nodelocked license servers, network license servers, network license clients, and the central registry.
- Installing the software product licenses on the servers.
- Monitoring the software products use through the Basic License Tool.
- Configuring the network.

annotation. See *license annotation*.

application client. A computer that runs a software product and plays the role of the client in the traditional client-server model.

application client identifier. In License Use Management, the unique identifier of the application client.

When a license is granted to a client, the ACID of the client is recorded in the central registry, which is checked at any new license request. This avoids granting a license twice to the same application client.

application server. A computer where an enabled product is installed, which provides shared access to the product to workstations (the application clients) over the network.

In License Use Management, the application server is the License Use Management Runtime client. It requests the licenses for all its application clients.

authorization. Defines the mode for which each user, group, or target ID can use a concurrent or a concurrent-offline license. The mode can be one of:

- For a concurrent license:
  allowed or not allowed
- For a concurrent-offline license:
  allowed, not allowed, or concurrent allowed

B

Basic License Tool. In License Use Management, the administration tool included in License Use Management Runtime, which enables the administrator to add or delete licenses from the server database, display the licenses installed, distribute the licenses among the servers available on the network, and generate reports on license usage and server events.

binding. In License Use Management, one of two methods by which a network license client can locate a server in order to request a license. See *direct binding* and *namespace binding*.

bundle. In License Use Management, a set of product licenses with a single serial number called the bundle ID. The set of licenses can include any combination of license types. All products that are part of a bundle are enrolled together when the bundle is enrolled. A product that is part of a bundle cannot be removed separately, it can be removed only by removing the bundle.

C

C language. A language used to develop software applications in compact, efficient code that can be run on different types of computers with minimal change.

C++ language. A programming language based on the C language and developed by Bjarne Stroustrup. C++ defines data types called classes. Classes provide data abstraction and are composed of data (data members) and operations that operate on that data (member functions). C++ also implements the object-oriented programming concepts of inheritance through class derivation and polymorphism through virtual functions and overloading.

capacity type. In License Use Management, a number that specifies the type of unit that a license applies to. The type of unit can be defined by IBM or chosen by the vendor. If no capacity type is specified by the vendor, the license applies to users.

cell. See *NCS cell*.

central registry. In License Use Management, a database that contains information about:

- The enrollment and distribution of customer-managed use control products.
- Which application clients already have a per-seat license.
- Reservation of reservable licenses.

check period. In License Use Management, a time period during which a product holding a concurrent or unreserved reservable license must check in with the network license server. If the product does not check in during this period, the network license server assumes that the product is not running, and may release a granted license to another user.

cluster. In License Use Management, a group of network license servers that jointly serve vendor-managed concurrent licenses that are tied to the cluster rather than to an individual server. 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.

compound password. In License Use Management, a password from which it is possible to extract multiple simple passwords, each representing one or more licenses.

Enabled applications cannot use the compound password directly.

concurrent license. In License Use Management, a type of license, administered by the network license server, that can be used by different users from any node that is connected to a network license server. Concurrent licenses enable as many users to use a particular software product concurrently as there are licenses.

concurrent nodelocked license. In License Use Management, a nodelocked license that allows a limited number of concurrent uses of the licensed product on the node where the license is installed. Concurrent nodelocked licenses enable as many concurrent uses of a particular software product as there are licenses.

concurrent offline license. In License Use Management, a concurrent-offline license is identical to a normal concurrent license, but the concurrent-offline license allows users to reserve a concurrent license for a certain number of days and to use it on a portable computer disconnected from the network. Only authorized users can exploit this feature.

custom configuration. A selected combination of products, tailored by a vendor to the needs of one or more users. Each custom configuration is identified by unique serial number, which is incorporated into the custom configuration license.

custom configuration license. A special case of either a concurrent network license or a simple nodelocked license that contains a unique serial number identifying a custom configuration. See also *license*.

customer-managed use control. In License Use Management, a level of password use control in which the customer manages compliance with the terms of the software product acquisition. It is the customer's responsibility to set the upper limit on the number of

licenses that can be extracted and distributed, based on the terms of the software product acquisition.

D

default NCS cell. A cell that is identified by the default GLB object UUID. Workstations in the default cell do not have the glb_obj.txt file.

direct binding. In License Use Management, a type of binding between network license servers and clients in which client applications locate license servers by means of a local text file that contains network addresses of the license servers.

direct binding servers list. In License Use Management, a set of network license servers and a central registry license server that collectively serve a set of network license clients.

dynamic nodelocking. In License Use Management, a way of using licensing APIs in which a compound password installed on a network license server carries simple nodelocked licenses. Upon first invocation of the product at a client, a simple nodelocked license is extracted from the compound password and installed on the client workstation.

E

end user. In License Use Management, a user of license-enabled software products. The tasks of the end users may include:

- Installing License Use Management Runtime with the help of the administrator.
- Configuring License Use Management Runtime as a network license client.

enrollment certificate. In License Use Management, a mechanism for the distribution of licenses to end users. It is usually in the form of an electronic file, and contains all the information that is related to the licenses acquired for a license-enabled product.

G

gdb server. See *central registry*.

GLB. See *global location broker*.

glbd replica. In License Use Management, a copy, on a newly configured network license server, of a global location broker database that already exists on another server.

glb_obj.txt. A file that specifies the object UUID of the global location broker. The `glb_obj.txt` file makes it possible to override the default value by specifying a different GLB object UUID for a particular workstation. The `glb_obj.txt` file is used only in special

configurations that require several disjoint GLB databases (each of which is possibly replicated). In most networks and internets, there is only one GLB database (possibly replicated), and workstations do not need to have a `glb_obj.txt` file. If a workstation has a `glb_obj.txt` file, the UUID in the file identifies the GLB object to which that workstation directs lookups and updates.

global location broker. Part of the Network Computing System (NCS) that enables clients to locate servers in a network or internet. It is a process that manages a database that stores the locations (network addresses and port numbers) where server processes are running. The global location broker process maintains this database and provides access to it.

H

hard stop. In License Use Management, a policy according to which, if the end user starts the product and there are no licenses available, the product does not start.

high-availability licensing. In License Use Management, an option that makes it possible for a cluster of network license servers to jointly serve concurrent licenses, with one or more servers in reserve in case a server goes down. The software vendor must create passwords to be enrolled on the cluster rather than on an individual server.

high-water mark. In License Use Management, the maximum number of soft stop licenses that have been granted for a given product, over the number of licenses enrolled for that product. It is updated when the soft stop policy is set. In hard stop policy no updating of the high-water mark occurs, since it is assumed that the product stops its execution if no licenses are available.

I

internet. A set of two or more connected networks. The networks in an internet do not necessarily use the same communications protocol.

License Use Management Runtime supports the TCP/IP protocol.

initial key. A license key for a custom configuration license generated without using the Upgrade flag. It is an encrypted character string that specifies some terms of the acquisition of the selected combination of software products in a customer's initial custom configuration. Contrast with replacement key.

J

Java. An object-oriented programming language for portable interpretive code that supports interaction

among remote objects. Java was developed and specified by Sun Microsystems, Incorporated.

Java Development Kit (JDK). Software that Sun Microsystems distributes for Java developers. This software includes the Java interpreter, Java classes, and Java development tools. The development tools include a compiler, debugger, dissassembler, stub file generator, and documentation generator.

K

key. See *password*.

L

license. 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. Sometimes, a user needs more than one license to make full use of a particular product features.

The term *license* as used in the context of License Use Management does not refer to the license agreement that governs use of and rights to a product.

license annotation. A string that the vendor can use to modify the use of a license.

license database. In License Use Management, the database of licenses that a license server maintains.

license-enabled product. A product that is enabled for license use management.

A vendor provides a license-enabled product together with a password that authorizes use of the product. The password contains an encryption of certain terms of the acquisition of the product (such as how many licenses the customer can use, the expiration date of the licenses, and the type of license).

license information. In License Use Management, the information that describes licenses. This information consists of product name, product version, number of licenses, license type, start and end dates for the licenses, and a time stamp.

license key. See *password*.

license password. See *password*.

license server. A program that provides the license services, administering licenses for software products. It may be a network license server or a nodelocked license server.

licensed product. See *license-enabled product*.

local location broker. Part of the network computing system (NCS). It manages the local location broker (LLB) database, which stores information about NCS-based server programs that run on the local host.

location broker. See *local location broker* and *global location broker*.

log file. A database that records messages and errors from the license server, and sometimes also from licensed products.

M

multiuse rules. In License Use Management, rules that define the conditions under which multiple invocations of a product require only a single license. These rules are applicable only to concurrent access, concurrent nodelocked, and per-server licenses. The vendor of the product defines multiuse rules.

N

namespace binding. In License Use Management, a binding mechanism in which the network license servers register themselves with the global location broker, which locates an appropriate license server when a client requests a license. Namespace binding is not available on Windows platforms.

NCS. A set of software components, developed by Apollo Computer Inc., that conform to the Network Computing Architecture. These components include the Remote Procedure Call (RPC) runtime library and the Location Brokers.

NCS cell. A logical grouping of clients and servers; a subset of a network. Workstations in one cell cannot communicate with workstations in other cells. Workstations cannot be in more than one cell at a time. Workstations in the same cell are identified by the same *global location broker* (GLB) object *universal unique identifier* (UUID).

network. A group of nodes and the links that interconnect them.

network license. In License Use Management, a license that is maintained on a network license server for use upon request by a License Use Management Runtime client.

network license client. In License Use Management, a node configured to make use of licenses by requesting them from a network license server.

network licensed product. In License Use Management, a licensed product that is enabled such that the licenses are maintained on a server for use upon request by a License Use Management Runtime client.

network license server. In License Use Management, a node in the network on which network licenses are stored for use by License Use Management Runtime clients.

node. A workstation in the network. In License Use Management, it can be configured as a nodelocked license server, a network license client, a network license server, the central registry license server, or a combination.

nodelocked license. In License Use Management, a type of license locked to a specific node, so that the product can be used only at that node. The nodelocked license is installed on the workstation for which it was created.

nodelocked license server. In License Use Management, a server on a node that manages nodelocked licenses on that node.

non-runtime-based enablement. In License Use Management, a type of license enablement for a product with simple nodelocked licenses that does not make use of License Use Management Runtime on the end user's workstation. The password is stored in a special file when the enabled product is installed. When the enabled product is started, it checks the file to ensure that there is a valid license.

O

object. In the Network Computing System, an entity that is manipulated by well-defined operations. Databases, files, directories, devices, processes, and processors are all objects.

offline-nodelocked license. In License Use Management, an offline-nodelocked license is the copy of the concurrent-offline license that has been installed on a portable computer that is disconnected from the network. There cannot be more than one offline-nodelocked license for the same product, same product version, same user, same node (same machine), same group, and same nodelock file (there can be several nodelock files on the same machine). The offline-nodelocked license is local to the portable computer where the application has been installed.

offline period. The maximum number of days that a user can reserve a concurrent-offline license.

P

password. An encrypted character string that specifies some terms of the acquisition of a software product. See also *simple password* and *compound password*.

password use control level. In License Use Management, a level of control of compliance with the terms of the acquisition of a license-enabled product. The password use control levels are:
- customer-managed use control
- vendor-managed use control

per-seat license. In License Use Management, a license used to enable client/server applications that are constructed for multiple-server solutions. Assignment of a per-seat license to an application client is permanent. 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 that have an assigned license. If an application client connects to multiple application servers, only one license is assigned to it.

per-server license. In License Use Management, a license used to enable client/server applications that are constructed for multiple-server solutions. Each server license is associated with a specific number of clients. This number represents the maximum number of clients that may concurrently request that server application services at any given time. Assignment of a per-seat license to an application client is temporary. If an application client connects to multiple application servers at the same time, it is assigned more than one license.

product ID. In License Use Management, a number that identifies a vendor licensed software product. By means of product IDs, the license server can distinguish between products from the same vendor.

Q

queue. In License Use Management, a sequence of users who are waiting for a concurrent license to become available so they can run a product. The administrator can monitor the number of users in queue through the Basic License Tool.

R

remote procedure call. remote procedure call (RPC) A client request to a service provider located in another node.

replacement key. A license key for a custom configuration license generated using the Upgrade flag. It is an encrypted character string that specifies some terms of the acquisition of the selected combination of software products in a customer's upgraded custom configuration. Contrast with *initial key*.

replica. See *glbd replica*.

report. In License Use Management, a summary of the events related to the licenses that are installed on the selected servers, filtered as the administrator specified. Examples of events are:

- Requests for licenses for a product in a given interval of time.
- Server startup.

reservable license. In License Use Management, a network license that the administrator can reserve for

the exclusive use of a user, a group, or a node. The reservation is for a specified time period.

reservation. In License Use Management, a concurrent-offline license that is reserved (for a defined offline period) for use as an offline-nodelocked license.

reserved license. In License Use Management, a license that the administrator has reserved for the exclusive use of a user, a group, or a node.

RPC. See *remote procedure call*.

runtime-based enablement. In License Use Management, a type of license enablement for a product with nodelocked licenses that use License Use Management Runtime on the end user's workstation to manage the licenses.

S

selected servers. In License Use Management, the servers that the administrator is working with through the Basic License Tool. All the products whose licenses are installed on the selected servers are displayed in the Basic License Tool main window.

serial number. In License Use Management, a unique number that identifies a specific custom configuration.

simple nodelocked license. In License Use Management, a nodelocked license that allows an unlimited number of simultaneous uses of the licensed application on the local workstation.

simple password. In License Use Management, a password that, once enrolled on a license server, represents one or more licenses.

Enabled applications can use the simple password directly.

soft stop. A policy according to which, if the end user starts the product and there are no licenses available, the product starts.

T

target. In License Use Management, the node at which a password is to be installed. If the password specifies a nodelocked license, the target is the node where the licensed product is run. If the password specifies multiple nodelocked license (that is, a compound password for nodelocked licenses) or network licenses, the target is a node at which the network license server (i4lmd) is running.

target ID. In License Use Management, a unique identifier of a node. A vendor can generate a password that can be installed only on a node that has a specific target ID. The target ID can be based on hardware or generated by License Use Management Runtime.

TCP/IP. Transmission Control Protocol/Internet Protocol. A set of communication protocols that support peer-to-peer connectivity functions for both local and wide area networks.

threshold. In License Use Management, a percentage of licenses; if more than this percentage of licenses for a product are in use, messages about the level of use are logged.

time stamp. In License Use Management, a number that identifies the date and time at which a set of licenses was created.

trace. A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed.

try-and-buy license. In License Use Management, a nodelocked license that has a fixed duration and a start date equal to the date when the license is enrolled. A try-and-buy license is made available for purposes of evaluating the application, and can be replaced by a production license after evaluation.

U

UDP. See "User Datagram Protocol."

universal unique identifier. An identifier that is used by NCS to identify interfaces, objects, and types.

use-once license. In License Use Management, a type of license, administered by the license server, that is effective for only a single instance of starting a product or using a service. The license server decrements the number of available use-once licenses each time the product or service is used.

use-once nodelocked license. In License Use Management, a license that is valid for only a single instance of starting a product or using a service, on the node where the license is installed.

User Datagram Protocol (UDP). In the Internet suite of protocols, a protocol that provides unreliable, connectionless datagram service. It enables an application program on one machine or process to send a datagram to an application program on another machine or process. UDP uses the Internet Protocol (IP) to deliver datagrams.

user file. In License Use Management, an ASCII file created by an administrator using a text editor. The file lists users who specifically are or are not allowed to use specified products.

UUID. See *universal unique identifier*.

V

value units policy. A policy vendors use to price their products by introducing a common unit of measurement known as a *value unit*. Vendors can combine usage count, different system capacity measurements, and other product-specific indicators into the same license certificate. Vendors can also define the relative weights and rules for the translation of the basic metric units into value units.

vendor ID. In License Use Management, the identifier of a vendor of licensed products. Vendor IDs are a License Use Management Runtime specific usage of NCS Universal Unique Identifiers (UUIDs).

vendor-managed use control. In License Use Management, a level of password use control in which the vendor manages compliance with the terms of the software product acquisition.

The customer of a vendor-managed use product supplies a unique identifier (target ID) of each workstation where product licenses are to be installed. The vendor uses this information to create the password, which is tied to the target workstation and cannot be used on another workstation.

Index

A

access restriction
  user   15
accessibility   253
accessor ID   12
ACID
  definition   267
activating
  tracing   163
addresses, list of glb addresses   36
administering licenses using the basic license tool (BLT)   18
administrator
  definition   267
Admintool on Solaris   52
AIX
  installing on   45
  version
    compatible versions of LUM Runtime   3
AIX 5
  uninstalling on   48
alternate cells, for test environments   35
annotation
  definition   267
  license
    definition   269
annotation, license
  definition   269
ANY target id, use with VMU   5
API calls
  embedded   1
API code
  embedded   1
application client
  definition   267
application client identifier
  definition   267
application server
  definition   267
authorization
  definition   267
authorization policy, user
  ensuring consistency   116

B

backup
  and recovery, using built in procedure   199
  make before installing LUM Runtime   47
  procedure
    automatic   199
balance, workload   33
  troubleshooting heavy workloads   224
  tuning   204
basic concepts
  License Use Management   2
    license   2
basic license tool
  administering licenses   18
  description   18
  generating reports   18

basic license tool *(continued)*
  overview   3
  running on Windows 2000 Terminal Services   230
  running on Windows 2003 Terminal Services   231
Basic License Tool
  definition   267
basic license tool (BLT)
  CLI   140
    activate tracing   163
    clean up stale licenses   146
    deactivate tracing   163
    deleting bundles   147
    deleting licenses   147
    deleting reserved licenses   160
    deleting server log entries   166
    displaying a list of bundles   151
    displaying a list of licenses   151
    displaying a list of products   151
    displaying a list of servers   151
    displaying a list of vendors   151
    displaying a product license status   162
    distributing licenses   148
    enroll a product   142
    extracting licenses   148
    general rules for   141
    generating a report   158
    setting the Stress threshold   163
    update a product   164
    updating reserved licenses   160
  GUI
    clean up stale licenses   102
    filtering licenses and servers   102
    starting   101
    using   101
  help for   149
  performance
    troubleshooting   222
  troubleshooting
    performance   222
binding
  definition   267
  direct
    configuring   75
    description   31
    example   43
    HAL cluster considerations   40
    need to identify all servers   34
    no problems with multiple network interfaces   95
    planning for   34
    servers list   31
    troubleshooting   226
  how to select the correct type   31
  namespace
    configuring   75
    description   32
    HAL cluster considerations   40
    managing namespace binding with DSS or DCE   203
    planning for   34
    problems with multiple network interfaces   95
    troubleshooting   224
  performance considerations when choosing   33

S

V

W

IBM