

# ***MATRIXx***<sup>™</sup>

---

## **System Administrator's Guide (UNIX)**

## **Worldwide Technical Support and Product Information**

ni.com

### **National Instruments Corporate Headquarters**

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

### **Worldwide Offices**

Australia 1800 300 800, Austria 43 0 662 45 79 90 0, Belgium 32 0 2 757 00 20, Brazil 55 11 3262 3599,  
Canada (Calgary) 403 274 9391, Canada (Ottawa) 613 233 5949, Canada (Québec) 450 510 3055,  
Canada (Toronto) 905 785 0085, Canada (Vancouver) 514 685 7530, China 86 21 6555 7838,  
Czech Republic 420 224 235 774, Denmark 45 45 76 26 00, Finland 385 0 9 725 725 11,  
France 33 0 1 48 14 24 24, Germany 49 0 89 741 31 30, Greece 30 2 10 42 96 427, India 91 80 51190000,  
Israel 972 0 3 6393737, Italy 39 02 413091, Japan 81 3 5472 2970, Korea 82 02 3451 3400,  
Malaysia 603 9131 0918, Mexico 001 800 010 0793, Netherlands 31 0 348 433 466,  
New Zealand 0800 553 322, Norway 47 0 66 90 76 60, Poland 48 22 3390150, Portugal 351 210 311 210,  
Russia 7 095 783 68 51, Singapore 65 6226 5886, Slovenia 386 3 425 4200, South Africa 27 0 11 805 8197,  
Spain 34 91 640 0085, Sweden 46 0 8 587 895 00, Switzerland 41 56 200 51 51, Taiwan 886 2 2528 7227,  
Thailand 662 992 7519, United Kingdom 44 0 1635 523545

For further support information, refer to the *Technical Support and Professional Services* appendix. To comment on the documentation, send email to [techpubs@ni.com](mailto:techpubs@ni.com).

# Important Information

---

## Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THEREOF PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

## Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

## Trademarks

AutoCode™, DocumentIt™, MATRIXx™, National Instruments™, NI™, ni.com™, SystemBuild™, and Xmath™ are trademarks of National Instruments Corporation.

Product and company names mentioned herein are trademarks or trade names of their respective companies.

## Patents

For patents covering National Instruments products, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your CD, or `ni.com/patents`.

## WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

# Conventions

---

The following conventions are used in this manual:

» The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.



This icon denotes a note, which alerts you to important information.



This icon denotes a caution, which advises you of precautions to take to avoid injury, data loss, or a system crash.

**bold**

Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names.

*italic*

Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept. This font also denotes text that is a placeholder for a word or value that you must supply.

monospace

Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames, and extensions.

**monospace bold**

Bold text in this font denotes the messages and responses that the computer automatically prints to the screen. This font also emphasizes lines of code that are different from the other examples.

*monospace italic*

Italic text in this font denotes text that is a placeholder for a word or value that you must supply.

# Contents

---

## Chapter 1

### Installation Overview and Platform Requirements

Manual Organization .....	1-1
Installation Overview .....	1-1
Platform Requirements .....	1-2
Hardware Requirements .....	1-2
Software Requirements .....	1-2
Compatible Compilers and Operating System Versions .....	1-3
Minimum Disk Space Requirements .....	1-3
Requirements for Configuration Management Operation .....	1-4
Related Publications .....	1-4
Installation Configurations .....	1-4
Licensing .....	1-5
Troubleshooting Resources .....	1-5

## Chapter 2

### Software Structure

Planning Options .....	2-1
Directory Structure .....	2-1

## Chapter 3

### Licensing and Terminology

Licensing Overview .....	3-1
Floating and Node-Locked Licenses .....	3-1
Redundant License Servers .....	3-1
Flexible License Checkout .....	3-2
Xmath Licensing Commands .....	3-3
LICENSECHECKOUT .....	3-3
LICENSEUSER .....	3-3
LICENSEINFO .....	3-4
LICENSEFILE .....	3-5
NIREFNUM .....	3-5
Using Licensing .....	3-5
License Files .....	3-6
Proxy License Files .....	3-7
Compatibility .....	3-8

Licensing Requirements .....	3-8
Finding the Host ID.....	3-8
Finding the Computer Name.....	3-9
Contacting NI License Administration .....	3-9

## Chapter 4

### Installation and Related Tasks

MATRIXx 7.x Installation.....	4-1
Requirements .....	4-1
Privileges.....	4-1
Preparing for Installation .....	4-2
Mounting Your CD-ROM Drive .....	4-2
Stopping MATRIXx License Daemons (for Existing MATRIXx Users) .....	4-3
Installation Procedure .....	4-3
Permissions.....	4-5
Installing Other Components.....	4-6
Installing Altia Design .....	4-6
Installing Altia FacePlate .....	4-7
Installing the Standalone License Manager .....	4-7
Installing the Redundant License Servers.....	4-9
Unmounting Your CD-ROM Drive.....	4-10
Updating a License File.....	4-10
Using the MATRIXx Help .....	4-10
Using Netscape with MATRIXx.....	4-10
Environment Variables .....	4-10
Printing Files.....	4-11
Additional Netscape Information .....	4-11
Installation-Related Tasks .....	4-11
Setting Up a User Environment (.cshrc).....	4-11
Running Multiple MATRIXx Versions .....	4-12
Recording Your NI Reference Number .....	4-12
Moving the MTXHOME Directory Tree to Another Directory .....	4-12
Configuration Management Installation Tasks .....	4-12
Setting Up ClearCase.....	4-13
Setting Up PVCS .....	4-13
Troubleshooting Your Installation .....	4-14

## Chapter 5

### FLEXlm Licensing Tools

FLEXlm Utilities .....	5-1
lmgrd.....	5-1
lmstat .....	5-2
lmreread.....	5-3
lmhostid.....	5-3
lmdiag.....	5-4
lmver.....	5-4
lmborrow .....	5-4
Initiating Borrowing.....	5-5
Clearing the Borrowed License Setting .....	5-5
Determining Borrowed License Status .....	5-6
Returning a Borrowed License Early.....	5-6
lmutil.....	5-7
Resolving Problems .....	5-7
General Debugging Tips.....	5-8
Support Issues.....	5-8
Troubleshooting License Manager Problems .....	5-9
Host ID Problems .....	5-9
Connection Problems.....	5-10
Other Client Problems .....	5-11
Other Server Problems .....	5-11
Daemon Log File .....	5-13
Informational Messages.....	5-13
Configuration Problem Messages.....	5-15
Daemon Software Error Messages .....	5-16

## Appendix A

### Technical Support and Professional Services

### Index

---

# Installation Overview and Platform Requirements

This manual tells how to install MATRIXx Product Family release 7.x.

## Manual Organization

---

This manual includes the following chapters:

- Chapter 1 (this chapter) defines the MATRIXx release 7.x installation and platform requirements.
- Chapter 2, *Software Structure*, describes the MATRIXx directory structure and planning options.
- Chapter 3, *Licensing and Terminology*, describes licensing-related terminology and tasks.
- Chapter 4, *Installation and Related Tasks*, gives a detailed explanation of the MATRIXx 7.x installation procedure and related tasks.
- Chapter 5, *FLEXlm Licensing Tools*, discusses the MATRIXx release 7.x license manager utilities, license manager debugging, and daemon log file.

This guide also has an *Index*.

## Installation Overview

---

The MATRIXx 7.x software includes an installation program that performs a complete installation of the product. The installation instructions provided are for the C Shell. If you want to install 7.x using a different shell, use the equivalent commands for the shell you prefer. The installation software includes the utilities described in Table 1-1.

**Table 1-1.** Installation Utilities

Utility	Description	Path
INSTALLMX INSTALLLM INSTALLALTTIA INSTALLALTTIAFP	Automated installation programs for installing the MATRIXx software, standalone license manager, and Altia animation software.	\$MTXHOME
lmhostid	Command used to report the host ID of a system.	\$MTXHOME/NILM
lm_startup	Script used to start the license manager.	\$MTXHOME/NILM

MATRIXx software also includes the Flexible License Manager (FLEXlm), which resides in the NILM directory.



**Note** MATRIXx can be installed on a file server using a shared license manager.

## Platform Requirements

National Instruments supports the MATRIXx 7.x software in the hardware and software environments described in the following sections.

### Hardware Requirements

The following requirements apply to UNIX systems running MATRIXx software.

- Minimum 128 MB RAM per user, 256 MB or greater recommended.
- Minimum 256 MB swap space per user, 384 MB or greater recommended (typical ratio of RAM to swap space is 1:4).
- Approximately 340 MB of disk space to complete a full MATRIXx installation. Refer to Table 1-3 for details.
- CD-ROM drive for software installation.
- 3-button mouse or equivalent preferred.

### Software Requirements

The following requirements apply to UNIX systems running MATRIXx software.

- The MATRIXx software requires a supported version of UNIX. Supported versions are described in Table 1-2.
- Floating licenses require properly functioning TCP/IP networking.

- The *MATRIXx Help* requires Netscape Navigator 3.0 or later. A current version of Navigator Stand-Alone International is included on the MATRIXx product CD.
- MATRIXx 7.x has been tested with OpenWindows 3.x and Common Desktop Environment (CDE) on Sun Solaris. National Instruments software might work under other X Windows-based window managers, however, their use is not supported. The applicable X11 version is R5 or later for all supported UNIX platforms.

## Compatible Compilers and Operating System Versions

The MATRIXx 7.x software was developed and tested with the operating system version and compiler version listed in Table 1-2. These configurations are recommended for optimal compatibility.

Other operating system versions and compiler versions may be compatible with MATRIXx 7.x. For additional information, visit the support page of the National Instruments Web site at [ni.com/support](http://ni.com/support) or contact Technical Support at [support@ni.com](mailto:support@ni.com) or (877) 493-2404.

**Table 1-2.** Verified Compiler and Operating System Version

OS Version/Compiler	C	C++	FORTRAN
Sun Solaris 2.7	SC4.2	SC4.2	SC4.0



**Caution** Other compilers may work in some cases but are not supported. National Instruments recommends against using unsupported compilers.

## Minimum Disk Space Requirements

The minimum space requirements for MATRIXx 7.x on a UNIX host are shown in Table 1-3.

**Table 1-3.** Minimum Disk Space Requirements

Product	Space Required
MATRIXx	180 MB
License Manager	10 MB
Altia Design Animation (optional)	50 MB
Altia FacePlate (optional)	40 MB
Full installation	~280 MB

## Requirements for Configuration Management Operation

The MATRIXx configuration management (CM) feature provides integration between SystemBuild and various configuration management tools. A user can store catalog items in files, and perform common version management operations from within the SystemBuild environment. NI currently supports the following CM tools:

- ClearCase using both dynamic and snapshot views on the Solaris operating system
- Merant PVCS on the Solaris operating system

Each of these tools has its own native working environment that must be installed and functioning before the tool can be used with SystemBuild.

## Related Publications

NI provides a library of publications to support its products. The following documentation might be of interest.

- *MATRIXx Getting Started Guide*
- *FLEXlm End User Manual* (from Macrovision Corporation)

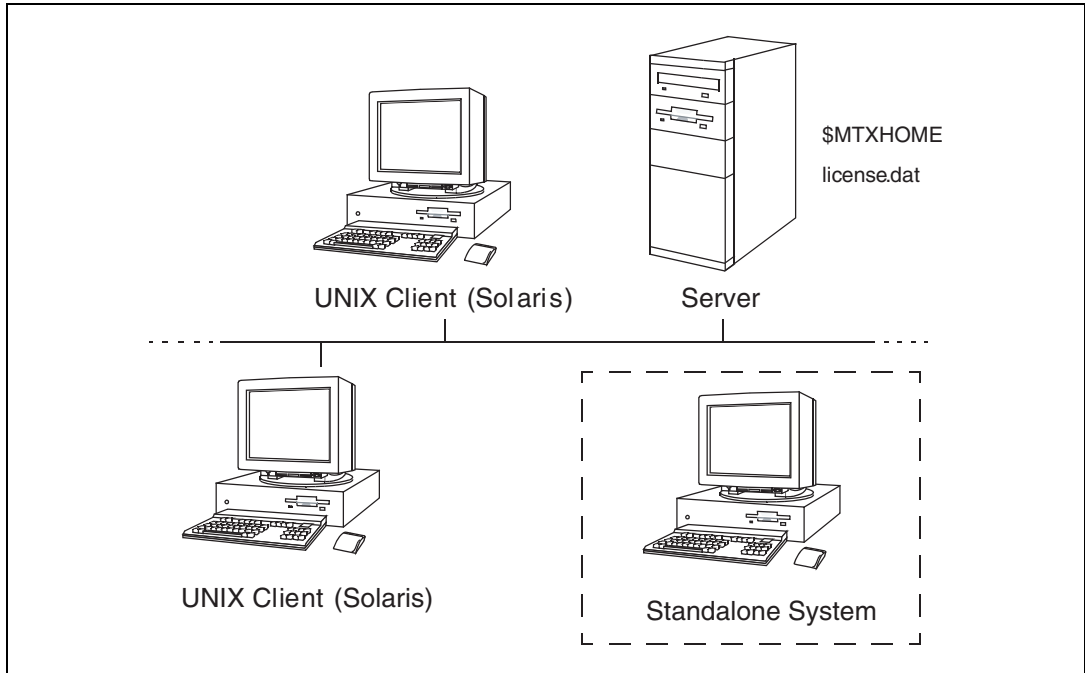
These and other MATRIXx documents are on the online documentation CD. For additional documentation, refer to the *MATRIXx Help* or the National Instruments Web site at [ni.com](http://ni.com).

## Installation Configurations

---

MATRIXx supports standalone and server installation on UNIX systems. Multiple different versions of MATRIXx software can be installed at one time on the same server, including for heterogeneous platforms. All of these versions can access the same `license.dat` file using a common license server.

All machines that use MATRIXx software must be able to access the `$MTXHOME` root directory. Figure 1-1 shows a typical client-server configuration.



**Figure 1-1.** Typical Client-Server Configuration

## Licensing

MATRIXx uses FLEXlm licensing utilities. For licensing information, refer to Chapter 3, *Licensing and Terminology*; Chapter 5, *FLEXlm Licensing Tools*; or the Macrovision *FLEXlm End User Manual* on the *MATRIXx Bookshelf*.

## Troubleshooting Resources

For troubleshooting tips, refer to the *Troubleshooting Your Installation* section of Chapter 4, *Installation and Related Tasks*. For troubleshooting assistance, refer to the National Instruments Web site at [ni.com/support](http://ni.com/support).

To participate in a moderated online forum with other MATRIXx customers and Product Support engineers select *Discussion Forums* on the MATRIXx support Web site at [ni.com/support/matrixsupp](http://ni.com/support/matrixsupp).

---

# Software Structure

This chapter describes installation planning options and MATRIXx directory structure.

---

## Planning Options

---

Before starting the installation, you should make decisions about how and where you want to install MATRIXx tools. These issues must be addressed *before* you start the installation procedure detailed in Chapter 4, [Installation and Related Tasks](#).

MATRIXx 7.x can be installed in the same root directory (`$MTXHOME`) as MATRIXx 6.x. This enables you to run two versions simultaneously with the same launching scripts and also to share the same license daemon. The MATRIXx 7.x installation will not affect an earlier release of MATRIXx software as long as you do not use the same parent directory.



**Note** MATRIXx 7.x cannot be installed in the same root directory as MATRIXx 5.x or earlier.

You can install a copy of MATRIXx on a local node using either floating or node-locked licensing, or you can use a file server (a shareable disk) and a shared license manager to install the MATRIXx software on a network server. The directory structure detailed here applies to both options, as it handles multiple releases and multiple platforms.

---

## Directory Structure

---

The directory structure shown in Figure 2-1 shows the MATRIXx product structure, which enables you to manage the software in a central location, even in a heterogeneous network environment. This structure reduces the system administration issues of multiple installations for a given platform, because installation can be completed solely on a file server using a common license manager. You can safely install multiple versions of MATRIXx 7.x into a common root installation directory (`$MTXHOME`).



**Note** All machines that will use MATRIXx software must be able to access the `$MTXHOME` root directory.

The commands used to invoke the MATRIXx suite of products are platform independent. These commands are found in the directory `$MTXHOME/bin`. Consequently, it is important that all users include `$MTXHOME/bin` in their path statements.



**Note** `$MTXHOME` and `install-dir` are effectively equivalent.

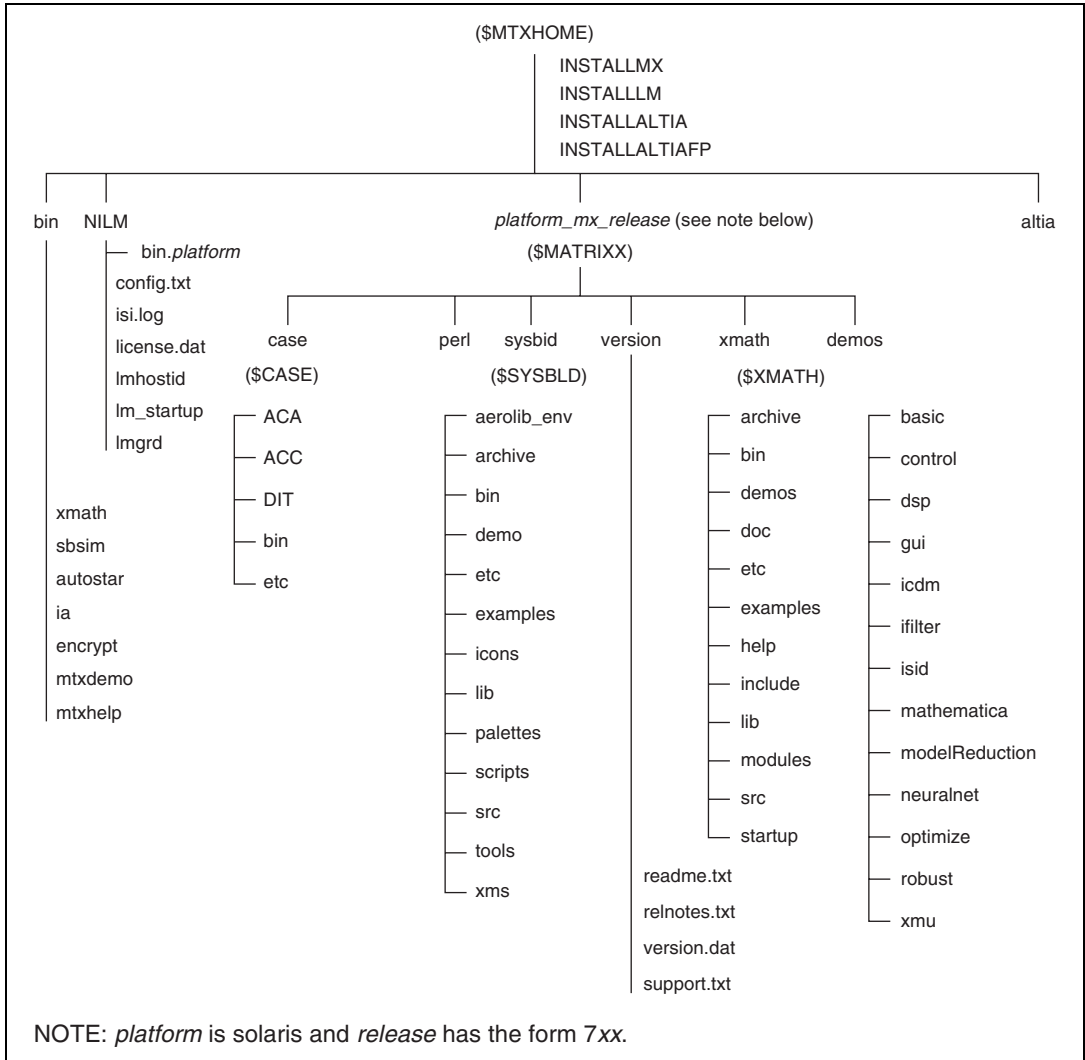


Figure 2-1. MATRIXx Product Family Sample File Structure

---

# Licensing and Terminology

This chapter provides a licensing overview and describes licensing-related terminology and tasks. MATRIXx 7.x includes a license daemon, `lmgrd`, which controls all NI product licenses. The `lmgrd` daemon resides in the `NILM/bin.platform` directory and must be running on your license server (or on all three if you have a redundant license server configuration). It does not have to be on the node where you install MATRIXx 7.x.

---

## Licensing Overview

MATRIXx licensing includes the following features:

- Floating and node-locked licenses
- Redundant license servers (optional)
- Flexible license checkout

### Floating and Node-Locked Licenses

License configurations for counted floating licenses, node-locked licenses, and evaluation licenses are available on UNIX systems. To check out a floating license or counted node-locked licenses, a client system must be connected to the network and have properly functioning TCP/IP software. Evaluation and uncounted node-locked licenses do not require you to have a running license server or network support.

Windows 2000/NT/XP and UNIX systems can check out floating licenses from a UNIX license server by referencing a copy of the same `license.dat` file used by the UNIX server, or by using a proxy license file. For more information, refer to the [Proxy License Files](#) section.

### Redundant License Servers

Redundant license servers are a set of three nodes designated to serve the same license file. If any node fails, the other two nodes will still be available to serve the licenses for MATRIXx.

To support the redundant license server scheme, NI License Administration requires information for three servers. For UNIX servers, this includes the `hostname` and `hostid` (or equivalent parameter) for each server.

NI issues redundant license server keys once you provide the `hostid` of the three servers. At a minimum, you need to run the standalone license server installation program (`INSTALLLM`) and start the license server on each machine. Complete the steps in the *Installing the Redundant License Servers* section of Chapter 4, *Installation and Related Tasks*.

Figure 3-1 shows a typical redundant license-server configuration.

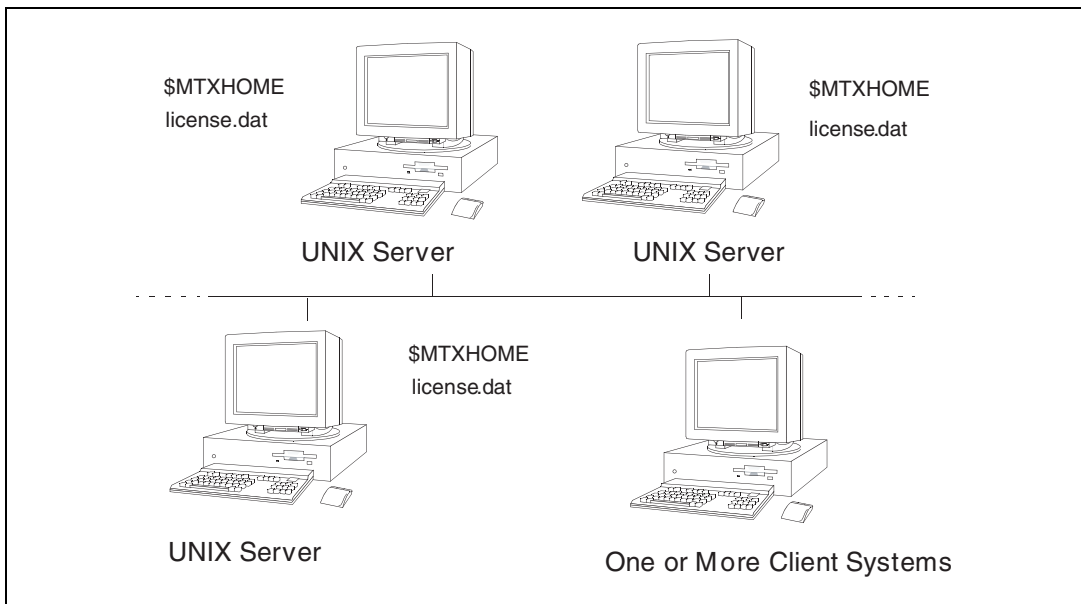


Figure 3-1. Typical Redundant License-Server Configuration

## Flexible License Checkout

With MATRIXx 7.x licensing, a feature (product component) is checked out as you use it. A feature stays checked out until you exit that product component. For example, Xmath and SystemBuild modules are checked out when first used and stay checked out until you exit Xmath or SystemBuild. SystemBuild is checked out when you launch the **Catalog Browser** (by starting SystemBuild) and stays checked out until you exit SystemBuild. The flexible license checkout feature is also known as *on-demand checkout*.

# Xmath Licensing Commands

---

Xmath has the following license-related commands:

- `LICENSECHECKOUT` for checking out features for future use
- `LICENSEUSER` for displaying assigned licenses for any feature names specified
- `LICENSEINFO` for displaying a license report for your site
- `LICENSEFILE` for displaying the current license file search path
- `NIREFNUM` for identifying your NI reference number

All of these commands must be run from the **Xmath Commands** window.

## LICENSECHECKOUT

`LICENSECHECKOUT` is an intrinsic command that checks out a license for the listed feature or features. Run `LICENSEINFO` to see a list of feature names. Features are specified as strings, and the name must be exactly as displayed by `LICENSEINFO`.

Although features are normally checked out on-demand, you can use `LICENSECHECKOUT` to reserve features for future use—for example, for a scheduled demo or presentation. They are checked in when you exit Xmath.

This command can be placed in a global or local `startup.ms` file to check out required features at startup. The syntax is as follows:

```
LICENSECHECKOUT feature1,feature2,...featureN
```

For example:

```
LICENSECHECKOUT
"Xmath", "Control", "Sysid", "Sysid2", "sysbld", "case"
```

## LICENSEUSER

The `LICENSEUSER` utility displays assigned licenses for any feature names specified. The default feature is Xmath if no argument is specified.

`LICENSEUSER` accepts feature names in string form. To see a list of the feature names for products you have purchased—for example, `aca` indicates AutoCode Ada—issue the `LICENSEINFO` command. The syntax for `LICENSEUSER` is as follows:

```
LICENSEUSER feature1,feature2,...featureN
```

Typical output is shown in Example 3-1.

**Example 3-1 LICENSEUSER Output**

```
licenseuser "xmath", "aca"
```

Users of Xmath:

```
Number of licenses purchased: 10
Number of unused licenses   : 4
```

User	Node Name	Starting Time
muemura	sampras	Thu May 29 19:33:44 2003
lynch	tribble	Thu May 29 19:34:36 2003
sanjay	castaway	Sat May 31 19:01:04 2003
rpizzi	castor	Mon Jun 2 15:21:54 2003
dawn	venus	Wed Jun 4 11:05:06 2003
uma	sampras	Mon Jun 9 13:45:47 2003

Users of aca:

```
Number of licenses purchased: 4
Number of unused licenses   : 2
```

User	Node Name	Starting Time
muemura	sampras	Thu May 29 19:56:44 2003
uma	sampras	Mon Jun 9 13:51:47 2003

## LICENSEINFO

LICENSEINFO displays a license report for your site, as shown in Example 3-2.

**Example 3-2 LICENSEINFO License Report**

```
licenseinfo
```

```
License File: $MTXHOME/NILM/license.dat
```

```
License Server Name: hardrock
```

```
License Server ID: 38234001
```

Module Name	Feature Name	Expiration Date	Type	Seats	Licensed Node
Xmath	Xmath	31-jan-2004	Floating	10	
Control Design Module	Control	31-jan-2004	Floating	10	
SystemBuild	sysbld	31-jan-2004	Floating	10	
HyperBuild Module	hyper	15-jan-2004	Floating	10	
AutoCode C Single Processor	acc	31-jan-2004	Node-Locked	4	80379b46
Model Reduction Module	Modred	31-jan-2004	Floating	10	
Interactive Animation Module	Icdm	31-jan-2004	Floating	10	
Robust Control Module	Robust	31-jan-2004	Floating	10	
RT/Fuzzy Module	blk_fuz	31-jan-2004	Floating	10	
AutoCode Ada Single Processor	aca	31-jan-2004	Floating	10	
DocumentIt	docit	31-jan-2004	Node-Locked	4	80379b46

## LICENSEFILE

The `LICENSEFILE` command displays the current license file search path. The syntax and example output are as follows:

```
LICENSEFILE
License files searched by MATRIXx are as follows:
/homes/ni/nilm/license.dat
/homes/ni/nilm/tplic.dat
```

## NIREFNUM

The `NIREFNUM` command displays the NI reference number, also known as your customer identification number. You should have this number ready if you plan to call NI Technical Support.

The syntax and example output are as follows:

```
NIREFNUM
Your NI Reference Number is as follows:
CUSTOMID99
```

## Using Licensing

---

With concurrent licensing, you can have a different number of licenses for each module or application. An application or module is either available on a first-come, first served basis, or node-locked to a given CPU. Some common uses of licensing are as follows:

- To see all the features your site has purchased or to view the license expiration date, use the `LICENSEINFO` command.
- To see who is using a given feature, or to see if a feature is available, use the `LICENSEUSER` command and specify the desired feature.
- To see the current license file search path, use the `LICENSEFILE` command.
- To reserve features for future use—for example, for a scheduled demo or presentation—use the `LICENSECHECKOUT` feature.

The initial distribution of licenses is determined at the time of the installation. To change licensing, your system administrator must get a new key from NI License Administration. To generate a new license file from this key, run `INSTALLLM` as described in Chapter 4, [Licensing and Terminology](#).

National Instruments uses the FLEXlm license manager, a product of Macrovision Corporation. If you have license manager questions beyond the scope of this document, refer to the *FLEXlm End User Manual* on the *MATRIXx Bookshelf*. For additional information about FLEXlm, refer to the Macrovision Web site at [www.macrovision.com](http://www.macrovision.com).

## License Files

---

The set of licensed features available for checkout by a MATRIXx user are contained in one or more license files, each containing encrypted feature lines. The following default license files are used by the MATRIXx products.

- `$MTXHOME/NILM/license.dat`—This primary license file contains feature lines for built-in MATRIXx features. The `license.dat` file is generated from NI-provided authorization keys during the MATRIXx or standalone license manager installation procedures.
- `$MTXHOME/NILM/tplic.dat`—This optional file contains license feature lines for third-party MATRIXx components. The encrypted feature lines in this file and the instructions for using the file will typically be provided by your third-party supplier. Multiple features from different third-party suppliers can be placed in this file. Typically, this file also will contain the same `SERVER` and `DAEMON` lines as `license.dat`. This file is not modified during a MATRIXx or license manager installation, nor is it deleted by uninstall procedures.

The search paths used to locate these license files are set in the environment variable `$NILM_LICENSE_FILE` by the MATRIXx product launching scripts. These NI license file paths are prepended to any existing definition of `$NILM_LICENSE_FILE`, thus allowing access to other MATRIXx or non-MATRIXx products also licensed by a FLEXlm license manager.

Normally, the only things you can change in a license file are as follows:

- The `hostname` (not the `hostid`).
- The port number on the `SERVER` line. If this port number is already in use, the license daemon (`lmgrd`) will report **Address In Use**. The default port number used in MATRIXx 7.x license files is 27000 (for earlier releases, it was 5200). A valid number is any unused port number between 1025 and 64000. The port number can be removed and the FLEXlm license manager will serve on any available FLEXlm reserved port in the range 27000–27009. For redundant license servers, however, you must use a fixed port number.

- The path to the vendor daemon executable on the `DAEMON` line. The vendor daemon is named `nilm`.
- `name=value` pairs on a `FEATURE` line can be changed if `name` is lowercase.

Redundant license server files contain three `SERVER` lines. The first `SERVER` line defines the primary license server. The second and third lines define the secondary license servers.

## Proxy License Files

---

If you are using a floating-license server, each licensing client needs a copy of the same license file used by the server. Alternatively, to avoid copying the license server file to each installation of `MATRIXx`, you can create a proxy license file in the `NILM` directory of each `MATRIXx` installation. This approach can be used for both single and redundant license server configurations. The proxy license file `license.dat` can be constructed by taking the `SERVER` line or lines from the license server file and then adding a line containing `USE_SERVER`.

For example, a proxy license file for a single server configuration would be as follows:

```
SERVER host1 17003456 27000
USE_SERVER
```

For example, a proxy license file for a redundant server configuration would be as follows:

```
SERVER host1 17003456 27002
SERVER host2 17004355 27002
SERVER host3 17007ea8 27002
USE_SERVER
```

Unless your server nodes are changed, you do not have to update the license file for each individual `MATRIXx` installation when the server license file is updated.

Although a proxy license file is the preferred way to reference the actual license file used by a remote license server, you also can use a `port@host` definition for either of the environment variables `LM_LICENSE_FILE` or `NILM_LICENSE_FILE`. `port` is the license file port number and `host` is

the hostname, and both are taken from the `SERVER` line of the actual license file. `NILM_LICENSE_FILE` is similar in use to `LM_LICENSE_FILE`, except that it is specific to the `MATRIXx` license manager daemon and overrides any concurrent definition of `LM_LICENSE_FILE`.

## Compatibility

---

`MATRIXx` releases 7.1 and later use the vendor daemon `nilm` (versus `mtx1md` for 6.x/7.0 and `isilm` in releases prior to 6.x). If you are installing `MATRIXx` 7.1 or later in a common root directory with `MATRIXx` 6.x/7.0 you do *not* have to kill the `mtx1md` license manager daemon in the `ISILM` directory because the license manager directory has been changed to `NILM` for `MATRIXx` 7.1 and later.



**Note** Because of the addition of new features and new license manager functionality, license keys issued prior to `MATRIXx` 6.x cannot be used to regenerate a license file for `MATRIXx` 7.x.

## Licensing Requirements

---

In order to obtain a standard license, you must provide the `computername` and `hostid` for each license server machine.

### Finding the Host ID

`FLEXlm` uses different machine identifications for different machine architectures. For example, all Sun machines have a unique host machine identification. An Ethernet address is used on some machine architectures as the *host ID*. An Ethernet address is a 6-byte quantity, with each byte specified as two hexadecimal characters. Specify all 12 hex characters when using an Ethernet address as a host ID. For example, if the Ethernet address is `8:0:20:0:5:ac`, specify `0800200005AC` as the host ID. To find an Ethernet address for a machine, use the command `ifconfig interface` from the root account, where a valid *interface* can be found by typing `netstat -i`.

For 32-bit host IDs, which are displayed as less than eight hex characters, prepend the ID with zeros (0's) until it has exactly eight+ hex characters. The same requirement applies to an Ethernet ID, which should be padded to exactly 12 hex characters.

The `$MTXHOME/NILM/lmhostid` program prints the exact host ID that FLEXlm expects to use on any given machine. Table 3-1 lists alternate methods to obtain the required host ID for each machine architecture.

**Table 3-1.** Obtaining the Host ID on Solaris

Platform	Host ID	OS Command	Example
Solaris	32-bit ID	hostid	73201bba

## Finding the Computer Name

Use either the `hostname` or the `uname -n` command at the UNIX prompt to display the Computer Name.

## Contacting NI License Administration

Contact NI License Administration in one of the following ways:

- Complete a Key Request Form (included with your software) and fax it to (512) 683-9007.
- Send email with the information asked for in the Key Request Form to `matrixx@ni.com`.
- Call (877) 493-2404 between 7 a.m. and 7 p.m. Central Time, Monday through Friday to speak with a MATRIXx representative to assist you with obtaining your license key.

NI supplies the authorization key(s) and checksum(s) used to generate the license file, which enables you to use the packages and modules you purchased. An authorization key or checksum is an alphanumeric string that does *not* contain the numbers 0, 1, or 5 to avoid confusion with uppercase letters O, I, and S.

---

# Installation and Related Tasks

This chapter describes how to install MATRIXx software, including any optional components such as Altia Design animation, and perform related tasks. It also describes how to view online books.

## MATRIXx 7.x Installation

---

This section describes how to use the installation program `INSTALLMX` to install MATRIXx 7.x software on a single platform using a CD-ROM drive.

### Requirements

Your system must meet the hardware and software requirements for your platform specified in Chapter 1, [Installation Overview and Platform Requirements](#).

### Privileges

No special privilege is required to install the MATRIXx 7.x software. NI recommends that you perform the installation from an account that allows write permission to the installer only. If you plan a heterogeneous installation (installing the MATRIXx software for more than one platform) do *not* use root privilege. However, you may need root privilege to perform the following installation-related tasks:

- Mount the CD-ROM.
- Kill an existing license daemon before installing, if it was started from another account.
- Modify the boot file to start the license daemon automatically after rebooting the license server.

## Preparing for Installation

Preparing for installation includes the following tasks, as applicable:

- Mount your CD-ROM.
- If you are an existing MATRIXx user, stopping any MATRIXx 6.x license daemon (`mtx1md`) on the installation machine is *not* necessary because MATRIXx 7.1 and later installs a new daemon (`nilm`).

### Mounting Your CD-ROM Drive

You can use either a local or remote CD-ROM drive to access the MATRIXx 7.x installation file sets. If you do not already have a mounted CD-ROM, mount one by completing the steps in the following sections.

#### Mounting a Local CD-ROM Drive

1. Log in to the root account and make a CD-ROM mount point. For example,
 

```
% mkdir /cdrom
```
2. Mount the directory with the appropriate command for your platform:
 

```
Solaris % mount -r -F hsfs /dev/dsk/c0t6d0s0 /cdrom
```



**Note** Device names—for example, `/dev/sr0`—vary depending on configuration.

3. Log out of the root account.

#### Mounting a Remote CD-ROM Drive

1. Log in as `root` on the machine that has the CD-ROM drive attached to it.
2. Create a directory to be the CD-ROM drive mount point (for example, `mkdir /cdrom`) and mount the CD-ROM drive according to the command specified in step 2 of the *Mounting a Local CD-ROM Drive* section.
3. Export the mount point of the CD-ROM drive to the NFS client. This allows your installation machine to mount the directory.
 

```
% exportfs -i -o ro /cdrom
```
4. Log out from the machine that has the CD-ROM drive attached.
5. Log in to your installation machine as `root`.

6. Create a directory on the installation machine to be the mount point for the remote CD-ROM drive machine, and then mount it.

```
% mkdir /cdrom
% mount cdrom_drive_machine_name:/cdrom /
cdrom_target
```

7. Log out from the installation machine, unless you need to stop a MATRIXx license daemon.

## Stopping MATRIXx License Daemons (for Existing MATRIXx Users)

If you have an existing MATRIXx release 7.0 or 6.x software installed and are installing MATRIXx 7.1 or later in the same \$MTXHOME directory, you can optionally stop the existing MATRIXx license daemon (mtxld) because the MATRIXx 7.1 or later uses a new daemon (nilm) installed in the new NILM directory.

1. Log in to the installation machine and check to see if a release 6.x or 7.x is running by entering:

```
ps -ef | grep ISILM/bin | grep -v grep
```

2. (Optional) If a 6.x or 7.0 lmgrd daemon (mtxld) is present (as determined by the displayed installation path), note the process ID and kill it (if the lmgrd daemon was started by root, you need root privilege to kill the process).

For example:

```
% kill process-id
```

## Installation Procedure

Before running the installation program, you will need the NI reference number, authorization key, and checksum. You also will need the hostname of your MATRIXx license server. If you have an existing license (for example, MATRIXx 7.x) you will not need this information.

1. Make sure that you are logged on to the correct operating system type (for example, Solaris) under the desired account. The installation procedure detects your operating system type, so it must match the target installation operating system.
2. Run the `setup.sh` script that resides in the `matrixx` directory on the CD. The `setup.sh` script prompts you for your installation directory.



**Note** Do not install MATRIXx 7.x into an existing MATRIXx 5.x directory.

The `setup.sh` script extracts your product files and runs the installation program. The installation program prompts you for your reference number, authorization key, and checksum if you choose to generate a new license file. If you have redundant server keys, the installation program also prompts for the hostnames of two other machines.

When you are prompted to generate a new license file, do one of the following:

- If you have an existing license file, exit by pressing the <Ctrl-C> keys when prompted to avoid generating a new license file.
  - If you need a new license file, enter the license key data when prompted.
  - If you intend to access a floating license provided by another host:
    - a. Exit by pressing the <Ctrl-C> keys when prompted to avoid generating a new license.
    - b. With a text editor, create a proxy license file named `license.dat` in the `install-dir/NILM` directory as described in the [Proxy License Files](#) section of Chapter 3, [Licensing and Terminology](#).
3. For evaluation licenses, or if you already have a MATRIXx license server running on another node, you can skip this step. Otherwise, start the 7.x license manager with the following command:

```
% install-dir/NILM/lm_startup
```



**Note** `install-dir` and `$MTXHOME` are effectively equivalent.

To shut down the license manager, enter:

```
% install-dir/NILM/lmdown -c license.dat
```

To check the status to see if startup worked, enter:

```
% install-dir/NILM/lmstat -c license.dat
```

This lets you know if both the license server and vendor daemon are *up*.

You also can check the `install-dir/NILM/ni.log` file to see if the license server is running correctly.

4. To start Xmath 7.x, and test the installation, type:
 

```
% install-dir/bin/xmath &
```

 or
 

```
% install-dir/bin/xmath -v release &
```

 where *release* is the software release that you just installed, or one of the releases in *install-dir/bin*.
5. If you are installing redundant license servers, complete the steps in the [Installing the Redundant License Servers](#) section.
6. (Optional) After you have verified that Xmath started successfully, refer to the [Installing Altia Design](#) section, if you plan to use Altia animation.
7. To ensure that the MATRIXx 7.x license daemon is available after a reboot, add the following startup command to the platform boot file as described in Table 4-1:

```
/bin/su username -c "umask 022; /path-to-lmgrd/lmgrd
-c /path-to-nilm/license.dat -l
/path-to-nilm/isi.log"
```

where

*/path-to-lmgrd* is equivalent to *install-dir/NILM/bin.platform*,

*/path-to-nilm* is equivalent to *install-dir/NILM*,

*platform* is *solaris*,

and *username* is a nonprivileged user.

**Table 4-1.** License Daemon Startup Command Instructions

Platform	Boot File Instructions
Solaris 2.7/2.8	<ul style="list-style-type: none"> <li>• Create <i>/etc/rc2.d/matrixx1md</i> containing the startup command</li> <li>• Enter the following:               <pre>ln /etc/rc3.d/S26matrixx1md /etc/init.d/matrixx1md</pre> </li> </ul>

## Permissions

License administration does not require root permission. This includes FLEXlm, *lmgrd*, and the vendor daemon. NI recommends that you do not run the license server (*lmgrd*) as root, because root processes can introduce security risks. If *lmgrd* is started from root, as in a system boot script,

NI recommends that you use the `/bin/su` command to run `lmgrd` as a nonprivileged user. You must ensure that the vendor daemons listed in the `license.dat` file have execute permissions for `username`. The paths to the vendor daemons are listed on the corresponding DAEMON lines.

## Installing Other Components

---

The following sections describe the installation procedures for installing optional MATRIXx components.

- [Installing Altia Design](#)
- [Installing Altia FacePlate](#)
- [Installing the Standalone License Manager](#)
- [Installing the Redundant License Servers](#)

### Installing Altia Design

Since the Altia Design software uses the NI license manager, you must have either MATRIXx or the standalone license manager installed. Before beginning installation, follow the instructions in the [Mounting Your CD-ROM Drive](#) section.

To install Altia Design, complete the following steps:

1. Make sure that you are logged on to the correct operating system type (for example, Solaris) under the desired account. The installation procedure detects your operating system type, so it must match the target installation operating system.
2. Run the `setup.sh` script located in the `altia/design` directory on the CD. The `setup.sh` script prompts you for the installation directory, which should be the same as your MATRIXx installation directory. The `setup.sh` script extracts your product files and runs the installation program.
3. After the installation program completes, you can run the Altia Design editor with the following command:
 

```
% install-dir/bin/altia
```
4. (Optional) You can execute the automated demo with the `install-dir/bin/altia_demo` command. You also can run any of the five tutorials with the `install-dir/bin/altia_tutorial1` with the `altia_tutorial15` command.



**Note** The Altia software uses the NI license manager. It does this by setting up the Altia code words file with the location of the MATRIXx license file, `license.dat`, during the execution of the installation program.

## Installing Altia FacePlate

Since the Altia FacePlate software uses the NI license manager, you must have either MATRIXx or the standalone license manager installed. Before beginning installation, follow the instructions in the [Mounting Your CD-ROM Drive](#) section.

To install Altia FacePlate, complete the following steps:

1. Make sure that you are logged on to the correct operating system type (for example, Solaris) under the desired account. The installation procedure detects your operating system type, so it must match the target installation operating system.
2. Run the `setup.sh` script located in the `altia/face` directory on the CD. The `setup.sh` script prompts you for the installation directory, which should be the same as your MATRIXx installation directory. The `setup.sh` script extracts your product files and runs the installation program.
3. After the installation program completes, you can run the Altia FacePlate with the following command:

```
% install-dir/bin/altiafp
```



**Note** The Altia software uses the NI license manager by setting up the Altia code words file with the location of the MATRIXx license file, `license.dat`, during the execution of the installation program.

## Installing the Standalone License Manager

The standalone license manager installation, which populates only the `NILM` subdirectory tree, is a subset of the full MATRIXx installation. It can be used to provide support for other standalone product installations such as Altia animation, which requires MATRIXx licensing but may not need the full MATRIXx product set to function. In addition, it can provide the required licensing installation on single or redundant license servers.

The sequence of steps to install the standalone license manager is essentially the same as installing MATRIXx. Before beginning installation, follow the instructions in the [Mounting Your CD-ROM Drive](#) section.

Before running the installation program, you will need the reference number, authorization key, and checksum. You also need the hostname of any MATRIXx license server.

1. Make sure that you are logged on to the correct operating system type (for example, Solaris) under the desired account. The installation procedure detects your operating system type, so it must match the target installation operating system.
2. Run the `setup.sh` script located in the `nilm` directory on the CD. The `setup.sh` script prompts you for your installation directory.



**Note** Do not install MATRIXx 7.x into an existing MATRIXx 5.x directory.

The `setup.sh` script extracts your product files and runs the installation program. The installation program prompts you for your NI reference number, authorization key, and checksum if you choose to generate a new license file. If you have redundant server keys, the installation program also prompts for the hostnames of two other machines.



**Note** Self-extracting versions of this same installation program are provided under the `nilm` directory on the CD for each of the supported license server platforms. Run the desired self-extracting shell script rather than `setup.sh` once you are logged on to the correct operating system type, and then follow the instructions in this section.

3. Start the MATRIXx 7.x license manager with the following command (skip this step for an evaluation license).

```
% install-dir/NILM/lm_startup
```

You can check the `install-dir/NILM/ni.log` file to verify that the license server is running correctly.

4. To ensure that the license daemon is available after a reboot, follow the instructions in step 7 of the [Installation Procedure](#) section (skip this step for an evaluation license).
5. Each separate installation of MATRIXx can access the license server with a proxy license file as described in the [Proxy License Files](#) section of Chapter 3, [Licensing and Terminology](#).

## Installing the Redundant License Servers

At a minimum, installing redundant license servers requires the installation of license manager software and starting the license manager daemon on two other nodes in addition to the primary license server. Special license keys are required for a redundant license server configuration.

To install a typical redundant license server configuration, complete the following steps:

1. Perform a standalone license manager installation on the primary license server, as described in the [Installing the Standalone License Manager](#) section, or a complete MATRIXx installation if you want this machine to also serve the MATRIXx application files, as described in the [Installation Procedure](#) section. Either procedure requires generating a license file and starting the license daemon.
2. On each of the two secondary license servers, run either the standalone license manager installation, as described in the [Installing the Standalone License Manager](#) section, or a complete MATRIXx installation if you want to replicate all the MATRIXx files, and then start the license daemon on each machine.



**Note** Rather than generating the license file on each secondary license server, copy `license.dat` from the NILM directory of the primary server to the NILM directory of each secondary server before starting the license daemons. Edit the DAEMON line of each secondary copy of `license.dat` to point to the daemon path that contains `nilmd` for that server.

3. To ensure that the license daemons are available after a reboot, follow the instructions in step 7 of the [Installation Procedure](#) section.
4. Each separate installation of MATRIXx can access the license server with a proxy license file as described in the [Proxy License Files](#) section of Chapter 3, [Licensing and Terminology](#).

## Unmounting Your CD-ROM Drive

---

After the installation is complete, you need to unmount the CD-ROM directories mounted as described in the [Mounting Your CD-ROM Drive](#) section.

1. Log in as `root` to the machine with a mounted CD-ROM directory.
2. Unmount any mounted CD-ROM directories. For example:  

```
% umount /cdrom
```
3. Log out.

## Updating a License File

---

Run `INSTALLLM` to perform license maintenance. This utility runs the license file generator. To update a license file, complete the following steps:

1. Change the directory to `install-dir`.
2. Run `INSTALLLM` with the following command:  

```
% ./INSTALLLM
```

## Using the MATRIXx Help

---

MATRIXx 7.x provides a hypertext markup language (HTML) help system. The *MATRIXx Help* is a self-contained system with multiple hypertext links from one component to another. This help system, augmented by online and printed manuals, covers most MATRIXx topics except for installation.

The *MATRIXx Help* runs with Netscape. An OEM version of Netscape is automatically included in the MATRIXx installation.

## Using Netscape with MATRIXx

### Environment Variables

If you plan to access the *MATRIXx Help* using `$XMATH/bin/netscape`, make sure that the following Netscape environment variables are *not* set before starting Xmath:

```
MOZILLA_HOME  
CLASSPATH
```

You can determine if these variables are set by looking at the output from the UNIX command `env`. If these values are set for a different version of Netscape (such as 2.0+), the **Exit** button in the *Netscape: MATRIXx Help* incorrectly interprets the associated JavaScript code. This causes all Netscape-related windows to close and produces a core file.

## Printing Files

To print a *MATRIXx Help* topic, go to the Topics Hierarchy, select a topic (for example, **Dynamic Systems»Control Design**), and then right-click and hold. A Netscape window displays with one of the options saying **Open Frame in New Window**. When the help topic is in the new window, select **Print**, or any other standard Netscape capability.

## Additional Netscape Information

For more information on Netscape products, refer to the Netscape Web site at `home.netscape.com`.

# Installation-Related Tasks

---

The following sections discuss installation-related tasks.

- [Setting Up a User Environment \(.cshrc\)](#)
- [Running Multiple MATRIXx Versions](#)
- [Recording Your NI Reference Number](#)
- [Moving the MTXHOME Directory Tree to Another Directory](#)
- [Configuration Management Installation Tasks](#)
- [Troubleshooting Your Installation](#)

## Setting Up a User Environment (.cshrc)

To allow access to MATRIXx 7.x each time you log in, place the following command in your `.cshrc` file.

```
set path=(install-dir/bin compiler_path $path)
```

where `install-dir` is your installation directory and `compiler_path` is the path by which the user can access a compiler for linking external files such as LNXs and UCBs. Specify all compilers the user will require.



**Note** In previous MATRIXx versions, makefiles on some platforms contained literal compiler paths. All makefiles now contain only `cc`, `CC`, and `f77` statements. All users must now include explicit compiler paths in their path definitions or linking will not work.

## Running Multiple MATRIXx Versions

You can install multiple versions of MATRIXx 7.x or later under a common installation directory. If you have multiple versions installed in a common directory, the following command lets you run a selected version:

```
% install-dir/bin/command_name -v version
```

where *install-dir* is your installation directory, *command\_name* is *xmath*, *sbsim*, *autostar*, *ia*, *encrypt*, *mtxdemo*, or *mtxhelp*, and *version* is one of the versions in the *install-dir/bin* directory. The version string appears as a suffix on the launching scripts.

## Recording Your NI Reference Number

To help us provide rapid support when you call National Instruments, the installation procedure prompts you for the 10-character reference number while installing MATRIXx 7.x or the license manager. Your reference number is stored in the file `$MTXHOME\NILM\config.txt`. If you do not have a reference number, you can obtain one from National Instruments when you request your Authorization Key and enter it into the first line of the `config.txt` file as follows:

```
REFERENCE NUMBER: 10_character_string
```

You can use the Xmath `NIREFNUM` command to display the reference number.

## Moving the MTXHOME Directory Tree to Another Directory

To move the `MTXHOME` directory tree to another directory after installing MATRIXx 7.x, complete the following steps:

1. Save the `license.dat` file (if required).
2. Delete the MATRIXx software after stopping the license daemon.
3. Reinstall MATRIXx in the new directory location.
4. Copy the `license.dat` file to the new directory location.
5. Restart the license daemon.

This is the method NI recommends to users.

## Configuration Management Installation Tasks

The MATRIXx configuration management (CM) utility provides integration between SystemBuild and ClearCase and Merant PVCS CM tools.

Before using this feature, you must install the CM license key. For information about obtaining and installing license features, refer to Chapter 3, *Licensing and Terminology*.

Each of these third-party CM tools has its own native working environment that must be installed and functioning before the tool can be used with SystemBuild.

SystemBuild decides which tool to use by looking at the `CMToolConfig` property setting specified in the `$/SYSBLD/etc/sysbld.ini` file or the user's `user.ini` file.



**Note** Since the CM tool selection is generally a system-wide setting, an administrator usually sets this value in the `sysbld.ini` file.

## Setting Up ClearCase

To perform ClearCase operations, SystemBuild uses a Perl script to relay calls into your ClearCase system. The script is located in `$/SYSBLD/cm/ClearCase.pl`. To use ClearCase, make sure the following line appears in the `[OPTIONS]` section of the `sysbld.ini` or `user.ini` file.

```
CMToolConfig = ClearCase.pl
```

The script accesses ClearCase functionality through the `cleartool`, `clearprompt`, and `xclearcase` commands, which are usually located in `/usr/atria/bin`. ClearCase operations will not be available unless this directory is in the user's path.

To test the user's environment, run `cleartool man -g checkin` from the user's UNIX shell. A graphical help viewer for the ClearCase checkin operation should appear.

## Setting Up PVCS

SystemBuild uses the PVCS project command-line interface (PCLI) and command-line interface (CLI) to perform PVCS operations. To use PVCS, make sure the line `CMToolConfig = PVCSPCLI` appears in the `[OPTIONS]` section of the `sysbld.ini` file or the `user.ini` file. This implementation uses a PCLI script that relays the CM operations from SystemBuild to PVCS. It uses commands such as `pcli`, `get`, and `put`, and expects these commands to be in the user's search path. These commands are normally located in the `/usr/pvcs/platform/vm/bin` directory. Also, PVCS requires that the `pvcs_install_dir/vm/solaris/lib`

/sparc/native\_threads library path be included in the user's library search path. For example, in a Solaris environment with PVCS installed in /usr/pvcs, the following lines (or equivalent) must appear in each user's .cshrc file:

```
if ($?LD_LIBRARY_PATH) then
    setenv LD_LIBRARY_PATH \
        ${LD_LIBRARY_PATH}:/usr/pvcs/vm/solaris/lib/sparc/native_threads
else
    setenv LD_LIBRARY_PATH/usr/pvcs/vm/solaris/lib/sparc/native_threads
endif
```

To test the user's environment, run the following commands from a UNIX prompt:

```
pcli run -h
get -h
```

These commands present information about the run and get commands from PVCS.

## Troubleshooting Your Installation

If you try to perform one of the following tasks and receive an error message, you may be able to correct it by identifying the cause as described in Table 4-2.

**Table 4-2.** Error Message Definitions

Command	Error Message or Problem	Cause or Remedy
lmdiag	lmdiag reports that the hostid is correct, but the license daemon has not been started.	Kill all lmgrd processes started for the license.dat file (they are trying to use the same TCP/IP port) and start a new lmgrd process.
lmreread -c license_file	You cannot use lmreread to change server node names or port numbers.	You have changed the server name or port number. Kill the old daemon, then restart the license daemon using <i>install-dir/NILM/lm_startup</i> .
INSTALLMX INSTALLLM	Message: <b>Invalid keys.</b>	You have entered the wrong authorization keys or checksum or, if you have an Evaluation License, the keys have expired.

---

# FLEXlm Licensing Tools

This chapter describes the FLEXlm utilities. This includes explaining how to troubleshoot common license manager problems and describes messages from the daemon log file. For additional licensing documentation, refer to the *FLEXlm End User Manual* from Macrovision Corporation on the MATRIXx documentation CD.

## FLEXlm Utilities

---

MATRIXx Product Family version 7.x uses FLEXlm for its license server. FLEXlm provides the following utilities to help manage the licensing activities on the network:

<code>lmstat</code>	Helps you monitor the status of all network licensing activities.
<code>lmdiag</code>	Lets you diagnose problems when you cannot check out a license.
<code>lmgrd</code>	Starts the license daemon.
<code>lmhostid</code>	Reports the host ID of a system.
<code>lmreread</code>	Causes the license daemon to reread the license file and start any new vendor daemons.
<code>lmver</code>	Reports the FLEXlm version of a library or a binary.
<code>lmborrow</code>	Supports license borrowing.
<code>lmutil</code>	FLEXlm license server management and administration.

### lmgrd

`lmgrd` is the main daemon program for FLEXlm. When you invoke `lmgrd`, it looks for the license file. The license file contains information about vendors and features.

Syntax is:

```
lmgrd [ -c license_file ] [ -t timeout ] [ -s interval ]
[ -b ] [ -l logfile ]
```

where:

<code>-c <i>license_file</i></code>	Uses the specified license file.
<code>-t <i>timeout</i></code>	Sets a timeout interval, in seconds, during which redundant daemons must complete their connections to each other. The default value is 10 seconds. A larger value may be desirable if the daemons are being run on busy systems or a heavily loaded network.
<code>-s <i>interval</i></code>	Specifies the <code>logfile</code> timestamp interval, in minutes. The default is 360 minutes.
<code>-l <i>logfile</i></code>	Specifies the pathname to the log file. If this option is not specified, the log will be sent to standard output.
<code>-v</code>	Prints the <code>lmgrd</code> version number.

## lmstat

The `lmstat` utility helps you monitor the status of all network licensing activities, including:

- Which daemons are running
- Users of individual features
- Users of features served by a specific daemon

Syntax is:

```
lmstat [-a] [-S [DAEMON]] [-f [feature]]
[-s [server_name]] [-t value] [-c license_file] [-A]
[-l [regular expression]]
```

where:

<code>-a</code>	Displays everything.
<code>-A</code>	Lists all active licenses.
<code>-c <i>license_file</i></code>	Uses <code>license_file</code> .

<code>-S [DAEMON]</code>	Lists all users of the specified daemon's features.
<code>-f [feature_name]</code>	Lists users of the specified feature(s).
<code>-l [regular expression]</code>	Lists users of matching license(s).
<code>-s [server_name]</code>	Displays status of server node(s).
<code>-t value</code>	Sets <code>lmstat</code> time-out to <code>value</code> .

## lmreread

The `lmreread` utility causes the license daemon to reread the license file and start any new vendor daemons that have been added. Additionally, all pre-existing daemons will be signaled to reread the license file for changes in feature licensing information. Syntax is:

```
lmreread [-c license_file]
```



**Note** If you use the `-c` option, the license file specified will be read by `lmreread`, not by `lmgrd`; `lmgrd` rereads the file it read originally. Also, `lmreread` cannot be used to change server node names or port numbers. Vendor daemons will not reread their option files as a result of `lmreread`.

## lmhostid

The `lmhostid` utility reports the host ID of a system. Syntax is:

```
lmhostid [ether | long]
```

The output of this command appears as follows:

```
lmhostid - Copyright (c) 1989, 199x Highland Software,
Inc. The FLEXlm host ID of this machine is "69021c89"
```

The `ether` option causes `lmhostid` to print the Ethernet address on Hewlett-Packard (HP) systems. On HP systems, the `long` option prints the value of the HP ID module. These options are only available in FLEXlm v2.21 or later. The default is `long` for FLEXlm v2.21 or earlier, and `ether` for FLEXlm v2.4 or later.

## lmdiag

`lmdiag` allows you to diagnose problems when you cannot check out a license.

Usage is:

```
lmdiag [-c license_file] [-n] [feature]
```

where `-c license_file path` to file to diagnose. `-n` run in noninteractive mode; `lmdiag` will not prompt for any input in this mode. In this mode, extended connection diagnostics are not available. Feature diagnose this feature only.

If no feature is specified, `lmdiag` will operate on all features in the license file(s) in your path. `lmdiag` will first print information about the license, then attempt to check out each license. If the checkout succeeds, `lmdiag` will indicate this. If the checkout fails, `lmdiag` will give you the reason for the failure. If the checkout fails because `lmdiag` cannot connect to the license server, then you have the option of running “extended connection diagnostics.”

These extended diagnostics attempt to connect to each port on the license server node, and can detect if the port number in the license file is incorrect. `lmdiag` will indicate each port number that is listening, and if it is an `lmdiag` process, `lmdiag` will indicate this as well. If `lmdiag` finds the vendor daemon for the feature being tested, then it will indicate the correct port number for the license file to correct the problem.

## lmver

`lmver` reports the FLEXlm version of a library in binary. Usage is:

```
lmver [filename]
```

where *filename* can be `nilm.exe` or `lmgrd.exe`.

## lmborrow

`lmborrow` supports borrowing of licenses that contain the **BORROW** attribute. It must be run on the machine where licenses are borrowed. It is used to perform the following:

- Initiating borrowing by setting the borrow period
- Clearing the borrow period

- Determining borrow status
- Returning a borrowed license early

## Initiating Borrowing

To initiate borrowing, the user sets the borrow period by running `lmborrow` from the command line:

```
lmborrow {vendor | all} enddate [time]
```

where:

`vendor`

The vendor daemon name that serves the licenses to be borrowed, or `all` specifies all vendor daemons in that license server.

`enddate [time]`

Date the license is to be returned in `dd-mm-yyyy` format. `time` is optional and is specified in 24-hour format (`hh:mm`) in the FLEXlm-licensed application's local time. If `time` is unspecified, the checkout lasts until the end of the given end date.

For example:

```
lmborrow sampled 20-aug-2003 13:00
```

To borrow licenses for the desired vendor, *on the same day and the same machine* that the user runs `lmborrow`, run the application(s) to check out the license(s). If you run the application(s) more than once that day, no duplicate licenses are borrowed. No licenses are borrowed if the application is run on a day different than the date borrowing is initiated.

In addition to the `lmborrow` utility, there are other ways to initiate borrowing:

- Using the borrowing interface in application, if provided in the application.
- Setting the `LM_BORROW` environment variable directly.

## Clearing the Borrowed License Setting

To clear the `LM_BORROW` setting in the registry, issue the command:

```
lmborrow -clear
```

Clearing the `LM_BORROW` setting stops licenses from being borrowed until borrowing is initiated again. A user might run `lmborrow -clear` after he has borrowed licenses for features that are used offline if—before disconnecting from the network—he wants to run an application that checks out additional features, served by `vendor`, that are not meant to be borrowed. Clearing `LM_BORROW` does *not* change the status for already-borrowed licenses.

## Determining Borrowed License Status

To print information about borrowed features, issue the following command on the machine from which they are borrowed:

```
lmborrow -status
```

The borrowing system does not have to be connected to the network to determine the status.

## Returning a Borrowed License Early

To return a borrowed license early, first reconnect the borrowing system back to the network and then, from the same machine that initiated the borrowing, issue the command:

```
lmborrow -return [-c license_file_list]
                [-c display] feature
```

where:

`-c licesne_file_list` Use the specified license file(s). In some configurations, the license file needs to be specified in order to return the license file early.

`-d display` Used to specify the diaply from which the borrow was initiated. Required if you current display is different than what was used to initiate the borrow. On Windows, it is the system name or, in the case of a terminal server environmnet, the termial server client name.

`feature` The name of the borrowed feature to be returned early. Use `lmborrow -status` to get a list of borrowed feature names.

Returning the license early has the effect of clearing the `LM_BORROW` setting for the vendor daemon that serves the returned license.

If the borrowing system is not placed back on the network before attempting the early return, the license is not returned and LM\_BORROW is kept intact. Additionally, an error message is issued to the end user with notification that the system needs to be connected to the network.

## lmutil

The following are valid arguments for most `lmutil` utilities:

<code>-c license_file_path</code>	Most <code>lmutil</code> utilities need to know the path to the license file. This is specified with a <code>-c license_file_path</code> argument, or by setting the <code>LM_LICENSE_FILE</code> environment variable. Otherwise, the default location is used. The utilities also honor all <code>VENDOR_LICENSE_FILE</code> environment variables. Some utilities take more than one license file path in a license-file list separated by semi-colons. Pathnames which include spaces have to be enclosed in double quotes.
<code>-v</code>	Prints the FLEXlm version of the utility.
<code>-verbose</code>	Prints the description for all errors found. The output from the utilities may be harder to read with this option, but is useful for diagnostics. (v6.0+ only)

## Resolving Problems

---

This section offers some general debugging tips, and discusses information you should gather before contacting support. The [Troubleshooting License Manager Problems](#) section lists common license manager problems users have encountered before.

## General Debugging Tips

The following are tips for debugging:

- Examine the `$MTXHOME\NILM\debug.log` file.
- If you cannot check out a feature, run:  

```
$MTXHOME\NILM\lmdiag -c $MTXHOME\NILM\license.dat
```
- If the license daemon appears to have started correctly (which you should be able to determine from the `ni.log` file), try running `lmstat` to see if that program has the same problem as your application.  

```
lmstat -a -c $MTXHOME\NILM\license.dat
```

## Support Issues

When you make a support call, please be prepared to answer the following questions:

- What kind of machine is your license daemon running on? What version of the operating system is the application running on?
- What version of FLEXlm does the program use? Use the following command on your `nilm` vendor daemon and application:  

```
$MTXHOME\NILM\lmver $MTXHOME\NILM\nilm.exe
```
- What error or warning messages appear in the log file? Did the daemon start correctly? Look for a message such as:  

```
server xyz started for: feature1 feature2.
```
- What is the output from running `lmstat -a`?
- Are you running other products which are also licensed by FLEXlm? National Instruments does not support combined license files.

# Troubleshooting License Manager Problems

---

This section lists areas of FLEXlm administration that have given customers difficulty in the past. Categories are *Host ID Problems*, *Connection Problems*, and *Other Client Problems*.

## Host ID Problems

**symptom** When I run the license manager on my machine, it tells me it is the wrong host ID.

**cause** The vendor daemon checks the host ID listed on the `server` line in the license file; if it does not match the host ID of the machine it is running on, this message will be printed.

Possible causes include:

- You are trying to run the license daemon on a different machine from the machine the file was made for.
- The host ID of the machine you are running on changed (for example, the HP ID module was moved, or the CPU board was replaced).
- The host ID in the license file was modified.

**solution** Verify that the host ID of the machine where the vendor daemon (or node-locked client program) is being run matches the host ID specified in the license file (on the `server` line for the vendor, or on the `feature` line for a node-locked client). You can run the `lmhostid` program to see what FLEXlm thinks the host ID is. You cannot modify the host ID in the license file. If the host ID of your server machine changes, you will have to get a new license file from your software vendor.

## Connection Problems

- symptom** The application program (or `lmstat`) cannot connect to the server to check out a license.
- cause** The FLEXlm routines in the application are unable to make a TCP connection to the server and port specified in the license file. Possible reasons for this are:
- The wrong license file is being referenced by the application program.
  - The server machine specified in the license file is down.
  - The vendor daemon specified in the license file is not running.
  - The `hostname` in the license file is not recognized by the system.
  - The network between the client machine and the server machine is down.
  - TCP is not running on your machine.
- solution** Verify that the application is using the proper license file. Verify that the specified server machine is up and reachable by executing another command that uses TCP, such as `rsh` or `rlogin`, from the client to the server. Verify that the vendor daemon is running (you can use the `ps` command on the server to look for it). Examine the license log file to see if any problems are reported, particularly messages indicating that the vendor daemon has quit. Run `lmstat -a` from the server machine to verify that the vendor daemon is alive. Run `lmstat -a` from the client machine to verify the connection from client to vendor daemon across the network. Try using `telnet hostname portnum` where `hostname` and `portnum` are the same as on the `server` line in your license file.

## Other Client Problems

- symptom** When I run my application program (or vendor daemon), I get the error **bad code**.
- cause** Possible causes for this are:
- The license file was modified (either the host ID on a `server` line or anything on the `feature` line was changed).
  - The vendor used the wrong version of his license creation program to generate your license file (or there is a bug in that process).
- solution** You cannot modify the license file. If you need to change something in your license file, you must get a new license from National Instruments.
- symptom** When the second user tries to check out a license, the vendor daemon prints an error concerning `Parameter mismatch` in the log file and refuses the license.
- cause** The most likely cause of this problem is that you are simultaneously trying to run two different versions of the application program, and the software vendor has not specifically set up the new version for this kind of compatibility. Check the license server log file for a **comm version mismatch** warning message; this indicates that someone is running a V1.5 client with a V2.1 or later license server.
- solution** Run only the new version of the application (or only the old version).

## Other Server Problems

- symptom** When I start `lmgrd`, it says `Retrying socket bind (address in use: port xxxx)`.
- cause** The license server listens on the port `xxxx` that has already been used by another server program. 99.44% of the time, if it is in use, it is because `lmgrd` from NI or another vendor is already running on the port—or was recently killed, and the port is not freed yet.

**solution** In the `license.dat` file NI has put 27000 at the end of the `SERVER` line as the port number. You can remove it to let `lmgrd` scan for a free port from 27000 to 27009. You also can specify a port number selected by yourself for other concerns like farewell.



**Caution** The portscan feature of `lmgrd` is only available for FLEXlm v6+ licenses. You cannot drop the port numbers when configuring redundant servers.

**symptom** When I start up `lmgrd`, it says **exec1 failed** on my vendor daemon.

**cause** `lmgrd` uses `exec1` to start each vendor daemon running. If there is a problem starting the vendor daemon, this message is output to the log file. This error is typically caused by one of the following:

- There is no executable at the location referred to by the license file (and printed out in the log file).
- The executable does not have the proper protection to be run (the file does not have the `x` bit set, or one of the directories in the path is not readable).
- There was an error building the executable, and it cannot be run.
- The executable is for a different machine architecture.

**solution** Verify that the path to the vendor daemon is absolute, and that it points to the executable program itself, not the containing directory (for FLEXlm v1.5). Ensure that the file exists by doing an `ls -l` of the vendor daemon `filename(s)` specified in the log file. Make sure you do this as the same user that started `lmgrd`. Verify that the file is executable. Run the vendor daemon directly from the command line. If the vendor daemon is properly linked, it will tell you that it must be run from `lmgrd`; if it crashes or fails to execute, then it is not properly linked.

**symptom** The license server keeps reporting `lost lock` errors in the log file and exiting.

**cause** The lockfile is being removed by someone else. There could be another daemon running, or the system administrator (or a script) could have deleted the file.

**solution** Check to see if there is more than one copy of the daemon running. Check for more than one `lmgrd` running as well, since it will restart your vendor daemon when it is killed. If more than one `lmgrd` is running, kill them all, then kill any remaining vendor daemons and start one fresh copy of `lmgrd`. Check to see whether a shell script is running that cleans out `tmp`. If so, try modifying it so that it does not delete zero length files.

## Daemon Log File

Daemons generate log files with `DAEMON NAME` messages where:

`DAEMON NAME` Either license daemon or the string from the `DAEMON` line that describes your daemon. In the case where a single copy of the daemon cannot handle all of the requested licenses, an optional underscore followed by a number indicates that this message comes from a forked daemon.

`message` The text of the message.

The log file, available as `$MTXHOME\NILM\debug.log`, can be used to:

- Inform users when they need to purchase additional application software licenses.
- Diagnose configuration problems.
- Diagnose daemon software errors.

## Informational Messages

These messages are only informational and do not necessarily indicate an error or a problem.

<code>Connected to <i>node</i>.</code>	This daemon is connected to its peer on node <i>node</i> .
<code>CONNECTED, master is <i>name</i>.</code>	The license daemons log this message when a quorum is up and everyone has selected a master.
<code>DENIED: <i>N feature to user</i> (<i>N</i> licenses)</code>	<i>user</i> was denied access to <i>N</i> licenses of <i>feature</i> .
<code>EXITING DUE TO SIGNAL <i>nnn</i>. EXITING with code <i>nnn</i>.</code>	All daemons list the reason that the daemon has exited.
<code>EXPIRED: <i>feature</i>.</code>	<i>feature</i> has passed its expiration date.

IN: feature by <i>user</i> ( <i>N</i> licenses).	<i>user</i> has checked back in <i>N</i> licenses of <i>feature</i> .
License Manager server started.	The license daemon was started.
Lost connection to <i>host</i> .	A daemon can no longer communicate with its peer on node <i>host</i> , which can cause the clients to have to reconnect, or cause the number of daemons to go below the minimum number, in which case clients may start exiting. If the license daemons lose the connection to the master, they will kill all the vendor daemons; vendor daemons will shut themselves down.
Lost quorum.	The daemon lost quorum, so it will only process connection requests from other daemons.
MULTIPLE <i>xxx</i> servers running. Please kill, and restart license daemon.	The license daemon has detected that multiple copies of vendor daemon <i>xxx</i> are running. The user should kill all <i>xxx</i> daemon processes and restart the license daemon.
OUT: feature by <i>user</i> ( <i>N</i> licenses).	<i>user</i> has checked out <i>N</i> licenses of <i>feature</i>
RESERVE <i>feature</i> for HOST <i>name</i> . RESERVE <i>feature</i> for USER <i>name</i> .	A license of <i>feature</i> is reserved for either user <i>name</i> or host <i>name</i> .
REStarted <i>xxx</i> (internet port <i>nnn</i> )	Vendor daemon <i>xxx</i> was restarted at internet port <i>nnn</i> .
Retrying socket bind (address in use).	The license servers try to bind their sockets for approximately 6 minutes if they detect “address in use” errors.
Selected (EXISTING) master node.	This license daemon has selected an existing master (node) as the master.
SERVER shutdown requested.	A daemon was requested to shut down through a user-generated kill command.
[NEW] Server started for: <i>feature-list</i>	A (possibly new) server was started for the features listed.
Shutting down <i>xxx</i> .	The license daemon is shutting down the vendor daemon <i>xxx</i> .
SIGCHLD received. Killing child servers.	A vendor daemon logs this message when a shutdown was requested by the license daemon.

Started name.	The license daemon logs this message whenever it starts a new vendor daemon.
Trying connection to <i>node</i> .	The daemon is attempting a connection to <i>node</i> .

## Configuration Problem Messages

hostname: Not a valid server host, exiting	This daemon was run on an invalid hostname.
hostname: Wrong hostid, exiting	The host ID is wrong for <i>hostname</i> .
BAD CODE for <i>feature-name</i>	The specified feature name has a bad encryption code.
CANNOT OPEN options file <i>file</i>	The options file specified in the license file could not be opened.
license daemon: lost all connections	This message is logged when all the connections to a daemon are lost, which often indicates a network problem.
lost lock, exiting	Error closing lock file.
Unable to re-open lock <i>file</i>	The vendor daemon has a problem with its lock file, usually because of an attempt to run more than one copy of the daemon on a single node. Locate the other daemon that is running via a <code>ps</code> command, and kill it with <code>kill -9</code> .
NO DAEMON line for <i>daemon</i>	The license file does not contain a <i>DAEMON</i> line for <i>daemon</i> .
No <i>license</i> service found	The TCP <i>license</i> service did not exist.
No license data for <i>feat</i> , feature unsupported	There is no feature line for <i>feat</i> in the license file.
No features to serve!	A vendor daemon found no features to serve. This could be caused by bad data in the license file.
UNSUPPORTED FEATURE request: <i>feature</i> by user	The <i>user</i> has requested a feature that this vendor daemon does not support. This can happen for a number of reasons: the license file is bad, the feature has expired, or the daemon is accessing the wrong license file.
Unknown host: <i>hostname</i>	The hostname specified on a <i>SERVER</i> line in the license file does not exist in the network database.

NO DAEMON lines, exiting	The license daemon logs this message if there are no DAEMON lines in the license file. Since there are no vendor daemons to start, there is nothing to do.
NO DAEMON line for name	A vendor daemon logs this error if it cannot find its own DAEMON name in the license file.

## Daemon Software Error Messages

select: message	An error in a select system call was detected.
Server exiting	The server is exiting. This is normally due to an error.



---

# Technical Support and Professional Services

Visit the following sections of the National Instruments Web site at [ni.com](http://ni.com) for technical support and professional services:

- **Support**—Online technical support resources at [ni.com/support](http://ni.com/support) include the following:
  - **Self-Help Resources**—For immediate answers and solutions, visit the award-winning National Instruments Web site for software drivers and updates, a searchable KnowledgeBase, product manuals, step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on.
  - **Free Technical Support**—All registered users receive free Basic Service, which includes access to hundreds of Application Engineers worldwide in the NI Developer Exchange at [ni.com/exchange](http://ni.com/exchange). National Instruments Application Engineers make sure every question receives an answer.
- **Training and Certification**—Visit [ni.com/training](http://ni.com/training) for self-paced training, eLearning virtual classrooms, interactive CDs, and Certification program information. You also can register for instructor-led, hands-on courses at locations around the world.
- **System Integration**—If you have time constraints, limited in-house technical resources, or other project challenges, NI Alliance Program members can help. To learn more, call your local NI office or visit [ni.com/alliance](http://ni.com/alliance).

If you searched [ni.com](http://ni.com) and could not find the answers you need, contact your local office or NI corporate headquarters. Phone numbers for our worldwide offices are listed at the front of this manual. You also can visit the Worldwide Offices section of [ni.com/niglobal](http://ni.com/niglobal) to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

# Index

---

## Symbols

\$MTXHOME, 2-2  
\$NILM\_LICENSE\_FILE, 3-6  
\$PATH environment variable, 4-11

## A

Altia Design, 4-6  
Altia FacePlate, 4-7  
authorization key, 3-9

## B

boot file instructions for license daemon, 4-5

## C

CD-ROM, detailed installation, 4-2  
checksum, 3-9  
ClearCase, 1-4  
CM, 1-4, 4-12  
Color Setting, via Control Panel, 3-8  
compiler paths, 4-12  
compilers, supported, 1-3  
computer name, 3-9  
config.txt, 4-12  
configuration problems, 5-15  
conventions used in the manual, *iv*  
customer support, 1-5

## D

daemon, 4-6  
daemon log files, 5-13  
diagnostic tools (NI resources), A-1  
directory structure, MATRIXx software, 2-1  
disk space, minimum, 1-3

documentation  
    conventions used in the manual, *iv*  
    NI resources, A-1  
drivers (NI resources), A-1

## E

environment variable  
    \$MTXHOME, 2-2  
    \$NILM\_LICENSE\_FILE, 3-6  
    \$PATH, 4-11  
examples (NI resources), A-1

## F

feature, listing users of, 5-3  
FLEXlm, 1-5

## H

hardware requirements, 1-2  
help  
    MATRIXx Help, 4-10  
    technical support, A-1  
host, 3-7  
host ID numbers, 3-8  
HTML, 4-10

## I

informational messages (not errors), 5-13  
INSTALLALTIA, 1-2  
INSTALLALTIAFP, 1-2  
installation, detailed, 4-1  
installation directory, install-dir, 4-11  
installation program (INSTALLMX), 3-1, 4-1

installing

- Altia Design, 4-6
- Altia FacePlate, 4-7
- redundant license servers, 4-9
- standalone license manager, 4-7

INSTALLLM, 1-2, 4-10

INSTALLMX, 1-2, 3-1, 4-1

instrument drivers (NI resources), A-1

ISIREFNUM, 3-5

## K

KnowledgeBase, A-1

## L

license daemon, 4-6, 4-8

- lmgrd, 1-2, 3-1

- lmgrd, 5-1

- location, 1-2, 3-1

license daemon (lmgrd), 5-2

license manager, troubleshooting, 5-9

license server, 3-1

license.dat, 3-6, 3-7, 4-9

LICENSECHECKOUT, 3-3

LICENSEFILE, 3-3, 3-5

LICENSEINFO, 3-4

licenses

- evaluation, 3-1

- floating, 2-1, 3-1

- node-locked, 2-1, 3-1

LICENSEUSER, 3-3

lm\_startup, 1-2

lmdiag, 5-1

lmgrd, 1-2, 3-1

lmgrd, 5-1, 5-2

lmhostid, 1-2, 5-1, 5-3

lmreread, 5-1, 5-3

lmstat, 5-1, 5-2, 5-3

lmver, 5-1

log files (from daemons), 5-13

## M

MATRIXx Help, 4-10

MATRIXx License Server applet, 4-10

Merant PVCS, 1-4

mounting a CD-ROM drive, 4-2

mtx1md, 4-9

## N

National Instruments support and services, A-1

Netscape, 4-10

ni.log daemon log file, 5-13

NILM directory, 3-7

NILMD\_LICENSE\_FILE, 3-7

## P

platform support, MATRIXx, 1-2

port, 3-7

privilege requirements for installation, 4-1

programming examples (NI resources), A-1

proxy license file, 3-7, 4-4

## R

running multiple versions, 4-12

## S

servers, display status with lmstat, 5-3

setting timeout value, 5-3

setting up

- ClearCase, 4-13

- PVCS, 4-13

- user environment, 4-11

software (NI resources), A-1

software requirements, 1-2

standalone license manager, 4-7

starting license daemons, 4-6

support  
    technical, A-1  
    Web site, 1-5  
supported compilers, 1-3  
supported operating system versions, 1-3

## T

technical support, A-1  
tplic.dat, 3-6  
training and certification (NI resources), A-1  
troubleshooting, 4-14  
troubleshooting (NI resources), A-1  
tutorials, 4-6

## U

unmounting a CD-ROM drive, 4-10  
user environment, setting up, 4-11

## V

version management, 1-4  
views, dynamic and snapshot, 1-4

## W

Web resources, A-1