

digital

OpenVMS License
Management Utility Manual

OpenVMS
OpenVMS



OpenVMS License Management Utility Manual

Order Number: AA-PVXUD-TK

November 1996

This manual describes the use of the License Management Facility (LMF), the software license management tool for the OpenVMS operating system.

Revision/Update Information: This is a revised manual. This manual supersedes *OpenVMS License Management Utility Manual*, OpenVMS AXP Version 6.2 and OpenVMS VAX Version 6.2.

Software Version: OpenVMS Alpha Version 7.1
OpenVMS VAX Version 7.1

**Digital Equipment Corporation
Maynard, Massachusetts**

November 1996

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

Digital conducts its business in a manner that conserves the environment and protects the safety and health of its employees, customers, and the community.

© Digital Equipment Corporation 1996. All rights reserved.

The following are trademarks of Digital Equipment Corporation: Alpha, AXP, Bookreader, DEC BASIC, DECdirect, DECnet, DECset, DECwindows, DECwrite, Digital, Digital Fortran, MicroVAX, OpenVMS, OpenVMS Cluster, VAX, VAX 9000, VAXcluster, VMS, VMScluster, VAXstation, and the DIGITAL logo.

The following are third-party trademarks:

Display POSTSCRIPT and POSTSCRIPT are registered trademarks of Adobe Systems Incorporated.

All other trademarks and registered trademarks are the property of their respective holders.

ZK4584

The OpenVMS documentation set is available on CD-ROM.

Contents

Preface	vii
1 Overview	
1.1 License Management	1-1
1.2 License Management Utility (LICENSE)	1-2
1.2.1 Displaying LICENSE Information	1-2
1.3 License Database	1-2
1.3.1 History Records	1-3
1.4 License Units and License Unit Requirement Tables	1-3
1.5 VMSLICENSE.COM	1-5
1.6 Types of Licenses	1-5
1.6.1 Availability Licenses	1-5
1.6.1.1 Providing Sufficient License Units	1-6
1.6.1.2 Providing More License Units	1-6
1.6.1.3 Providing Availability in an OpenVMS Cluster Environment	1-7
1.6.1.4 Providing More Availability	1-7
1.6.2 Activity License	1-7
1.6.2.1 Providing Enough License Units	1-8
1.6.2.2 Providing More License Units	1-9
1.6.2.3 Sharing License Units in an OpenVMS Cluster Environment	1-9
1.6.3 Personal Use License	1-9
1.6.4 User License	1-10
1.6.5 Group License	1-10
1.7 License Combination	1-11
1.7.1 Licenses That Can Be Combined	1-11
1.7.2 Include, Exclude, and Reservation Lists	1-12
1.7.3 Termination Dates and Version Numbers	1-12
2 Using LMF	
2.1 Steps to License Registration	2-1
2.2 Managing the License Database	2-2
2.2.1 Database Location	2-2
2.2.2 History Records	2-3
2.3 Getting a Product Authorization Key (PAK)	2-3
2.4 Registering Licenses	2-4
2.4.1 When To Perform Registration	2-5
2.4.2 Registration and Installation	2-5
2.4.3 Registration Methods	2-6

2.4.4	Using VMSLICENSE.COM	2-7
2.4.4.1	Using Data Files with VMSLICENSE.COM	2-10
2.4.4.2	Invoking VMSLICENSE.COM with a VMSLICENSE Data File	2-11
2.4.4.3	Creating VMSLICENSE Data Files	2-11
2.4.4.4	Using VMSLICENSE.COM Default Value Rules	2-12
2.4.5	Using the LICENSE REGISTER Command	2-13
2.4.6	Displaying License Information	2-14
2.5	Loading a License	2-14
2.5.1	Loading Licenses in an OpenVMS Cluster Environment	2-15
2.6	Managing Licenses After Registration	2-16
2.6.1	Restricting Product Use	2-17
2.6.1.1	Fastest Method	2-17
2.6.1.2	For Inactive Licenses	2-17
2.6.1.3	Permanent Methods	2-18
2.6.2	Controlling Node Access to Licenses in Clusters	2-18
2.6.2.1	Effect of the NO_SHARE Option	2-19
2.6.2.2	Editing Include and Exclude Lists	2-20
2.6.3	Controlling User Access	2-20
2.6.4	Controlling Loading Order	2-22
2.7	Logical Name LMF\$DISPLAY_OPCOM_MESSAGE	2-22

A Command Reference

LICENSE COPY	A-2
LICENSE CREATE	A-5
LICENSE DELETE	A-6
LICENSE DISABLE	A-9
LICENSE ENABLE	A-11
LICENSE ISSUE	A-13
LICENSE LIST	A-16
LICENSE LOAD	A-20
LICENSE MODIFY	A-23
LICENSE MOVE	A-29
LICENSE REGISTER	A-31
LICENSE START	A-35
LICENSE UNLOAD	A-36

B Examples

B.1	Registering a System Integrated Product	B-1
B.2	Difference Between LICENSE LIST and SHOW LICENSE	B-5
B.3	Restricting Product Use	B-7

Glossary

Index

Examples

2-1	Typical PAK Information	2-5
2-2	Sample VMSLICENSE Data File	2-11
2-3	Displaying License Database Information	2-13

Figures

2-1	PAK Transfer Methods	2-4
2-2	From a PAK to the License Database	2-5
2-3	The PAK and Software Routes to a License	2-6
2-4	Loading a License	2-15
2-5	Loading a License in an OpenVMS Cluster Environment	2-16

Tables

1-1	Sample License Unit Requirement Table F (Layered Products)	1-4
1-2	Types of Licenses	1-5
2-1	Parameters Used with VMSLICENSE.COM	2-12
A-1	LICENSE REGISTER /OPTIONS Keywords	A-32

Preface

The License Management Facility (LMF) is the software license management tool for the OpenVMS operating system. To run any software product on OpenVMS systems, its license must be registered and loaded. To perform these tasks, use LMF.

Intended Audience

This manual is for managers of licenses for software products that run on the OpenVMS operating system. Typically, the system manager has this responsibility.

Document Structure

This manual consists of the following parts:

- Chapter 1 provides an introduction to the licensing of OpenVMS layered products and to the OpenVMS LMF.
- Chapter 2 describes each task required to manage software product licenses. This chapter also discusses the License Management utility (LICENSE) commands and shows how to use them.
- Appendix A describes the syntax of the LICENSE commands.
- Appendix B provides examples.
- The Glossary defines the LMF-related terms used in this manual.

Related Documents

The following manuals contain information related to the License Management utility:

- *OpenVMS System Manager's Manual*
- *OpenVMS System Management Utilities Reference Manual*
- *OpenVMS Record Management Utilities Reference Manual*
- *OpenVMS DCL Dictionary*

For information about installing software, see the following documentation:

- Upgrade and installation manual for your version of OpenVMS software
- Upgrade and installation supplement for your computer
- The installation guides, release notes, and Software Product Descriptions (SPDs) for any software products you install

Reader's Comments

Digital welcomes your comments on this manual.

Print or edit the online form SYS\$HELP:OPENVMSDOC_COMMENTS.TXT and send us your comments by:

Internet **openvmsdoc@zko.mts.dec.com**
Fax 603 881-0120, Attention: OSSG Documentation, ZKO3-4/U08
Mail OSSG Documentation Group, ZKO3-4/U08
 110 Spit Brook Rd.
 Nashua, NH 03062-2698

How To Order Additional Documentation

Use the following table to order additional documentation or information. If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Location	Call	Fax	Write
U.S.A.	DECdirect 800-DIGITAL 800-344-4825	Fax: 800-234-2298	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	809-781-0505	Fax: 809-749-8300	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street, Suite 200 P.O. Box 11038 Metro Office Park San Juan, Puerto Rico 00910-2138
Canada	800-267-6215	Fax: 613-592-1946	Digital Equipment of Canada, Ltd. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	—	—	Local Digital subsidiary or approved distributor
Internal Orders	DTN: 264-4446 603-884-4446	Fax: 603-884-3960	U.S. Software Supply Business Digital Equipment Corporation 8 Cotton Road Nashua, NH 03063-1260

ZK-7654A-GE

Conventions

The name of the OpenVMS AXP operating system has been changed to OpenVMS Alpha. Any references to OpenVMS AXP or AXP are synonymous with OpenVMS Alpha or Alpha.

VMScluster systems are now referred to as OpenVMS Cluster systems. Unless otherwise specified, references to OpenVMS Clusters or clusters in this document are synonymous with VMSclusters.

Note

Even though the command procedures and displays in this manual use VMS terminology, they pertain to OpenVMS.

The following conventions are used in this manual:

Ctrl/ <i>x</i>	A sequence such as Ctrl/ <i>x</i> indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 <i>x</i> or GOLD <i>x</i>	A sequence such as PF1 <i>x</i> or GOLD <i>x</i> indicates that you must first press and release the key labeled PF1 or GOLD and then press and release another key or a pointing device button. GOLD key sequences can also have a slash (/), dash (-), or underscore (_) as a delimiter in EVE commands.
Return	In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)
...	Horizontal ellipsis points in examples indicate one of the following possibilities: <ul style="list-style-type: none">• Additional optional arguments in a statement have been omitted.• The preceding item or items can be repeated one or more times.• Additional parameters, values, or other information can be entered.
.	Vertical ellipsis points indicate the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
()	In command format descriptions, parentheses indicate that, if you choose more than one option, you must enclose the choices in parentheses.
[]	In command format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in an OpenVMS file specification or in the syntax of a substring specification in an assignment statement.)
{ }	In command format descriptions, braces indicate a required choice of options; you must choose one of the options listed.

text style	<p>This text style represents the introduction of a new term or the name of an argument, an attribute, or a reason.</p> <p>This style is also used to show user input in Bookreader versions of the manual.</p>
<i>italic text</i>	<p>Italic text indicates important information, complete titles of manuals, or variables. Variables include information that varies in system messages (Internal error <i>number</i>), in command lines (/PRODUCER=<i>name</i>), and in command parameters in text (where <i>device-name</i> contains up to five alphanumeric characters).</p>
UPPERCASE TEXT	<p>Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.</p>
-	<p>A hyphen in code examples indicates that additional arguments to the request are provided on the line that follows.</p>
numbers	<p>All numbers in text are assumed to be decimal unless otherwise noted. Nondecimal radices—binary, octal, or hexadecimal—are explicitly indicated.</p>

Overview

This overview introduces the OpenVMS License Management Facility (LMF) and outlines the tasks required to manage software licenses for software products.

A product license protects the intellectual property of the software vendor and provides customers with access to the product. Product authorization is usually defined in a contract with specific terms and conditions agreed upon by the software license issuer and the software user.

Digital Equipment Corporation and other software vendors provide software to their customers under an agreement called a **license**. In this document the term *license* means the authorization you have to run a software product on the OpenVMS operating system.

License Management Facility and License Agreements

The terms and conditions of your license agreement determine your legal use of software.

LMF is a management tool that can help you comply with your license agreement, but use of LMF does not indemnify you against noncompliance with the terms and conditions of your software license agreements. In other words, LMF offers options for many kinds of license agreements, but using some of these options may not be authorized by your specific license agreement. You must read your license carefully to determine which LMF options you can use legally.

This document describes some features of LMF that Digital Equipment Corporation does not currently authorize. Digital may in the future authorize the use of some features described herein but makes no commitment beyond the current Software Business Practices.

1.1 License Management

To use a software product that requires a license, you must typically perform the following steps:

- Obtain a **Product Authorization Key (PAK)**, which provides information required to register the license.
- Use LMF to register the license in the **License Database**.
- Use LMF to load the license.
- Install the product specified by the PAK.

LMF provides additional features to modify licenses to satisfy specific requirements of individual sites.

Overview

1.1 License Management

To manage software product licenses for OpenVMS layered software, you need to understand the following information about licenses and the tool to manage them on OpenVMS systems:

- License Management utility (LICENSE) (Section 1.2)
- License Database (Section 1.3)
- License Unit Requirement Tables (LURTs) (Section 1.4)
- VMSLICENSE.COM command procedure (Section 1.5)
- Types of licenses (Section 1.6)
- Combining licenses (Section 1.7)

1.2 License Management Utility (LICENSE)

The License Management utility (LICENSE) is the command line interface of LMF. LICENSE provides the commands used to manage interactively the licenses of OpenVMS layered software products made by Digital and, in many cases, by third-party vendors.

LICENSE is a system-level tool that you use at the DCL prompt.

Example: This example shows a LICENSE MODIFY command line at the DCL prompt. The command modifies the Encryption for OpenVMS license in the License Database so that users can encrypt files from the node DRAMA.

```
$ LICENSE MODIFY /ADD /INCLUDE=(DRAMA) -  
_ $ /COMMENT="Modified to add node DRAMA 10/23/96" -  
_ $ ENCRYPTION
```

For detailed information about using LICENSE commands, see Chapter 2.

1.2.1 Displaying LICENSE Information

Use the LICENSE LIST command and the DCL command SHOW LICENSE to track and monitor your system's license information. For information about these commands, see Section 2.4.6.

1.3 License Database

The License Database is a collection of information stored on a disk that contains information about each license on your system. The default database file is SYS\$COMMON:[SYSEXEC]LMF\$LICENSE.LDB, which is created by LMF when you install the OpenVMS software.

The terms of each license are stored in a collection of data fields in the database. Data is entered by:

- You, issuing LICENSE commands
- You, running VMSLICENSE.COM
- LMF, keeping records

The collection of data fields representing a license at any one time is called a **record**. When you first register a license, you create the first record with data specified in your PAK. If you later modify the license, LMF creates a new record to define the modified terms of the license, and includes a notation that the license was modified.

1.3.1 History Records

LMF keeps track of the licensing activity on your system by writing a history record to the License Database every time you modify a PAK. Each history record contains an exact copy of the following:

- License record before modification
- LICENSE command you used to modify the record
- Date and time that you made the changes

For information about viewing and purging these records, see Section 2.2.2.

1.4 License Units and License Unit Requirement Tables

A **license unit** is a measurement of the authorization granted for use of a product. License units define the size of each license. Each license has a size, specified in license units. Each processor has a series of license unit requirements, also specified in license units.

The license unit requirements of a processor are expressed in a *rating*. LMF contains ratings (in license units) for all available and appropriate processors in a table called the **License Unit Requirement Table (LURT)**. There is a LURT for each category of software products as identified on the PAK. The PAK contains two fields, the Activity Table Code, and the Availability Table Code, which contain information to identify the category of the software product. Typically, processors that provide more performance have greater license unit requirements, but ratings may be unrelated to performance.

The size of a software product license must be large enough to support the number of users or processes using the product and the processor on which the product is to run. LMF compares the size of a registered license to the rating of the current processor and authorizes product use when a license supplies sufficient license units.

A copy of each LURT is published in either the OpenVMS release notes or installation documentation. Use the following command to determine the license unit requirements of your system:

```
$ SHOW LICENSE/UNIT_REQUIREMENTS
```

LMF compares the size of a registered license to the license unit requirement for the current processor and authorizes product use when a license supplies sufficient license units. Thus, you cannot load a license registered with fewer license units than a processor requires.

To check whether your license has an appropriate license unit value (size) for your processor, LMF performs the following process:

1. It looks for a code (*A, B, C, D, E, F, G, H* or *I*) or a keyword (*CONSTANT integer*) next to either the Availability Table Code: field or the Activity Table Code: field in the registered license.
 - If the license specifies *CONSTANT* and an integer value, LMF stops and defines the license unit requirement as equal to the stated integer value. This value could be the decimal value 0, which means the license has no unit limitations.
 - If the license does not specify a constant unit requirement, LMF looks for a code that specifies license type and corresponds to a LURT.

Overview

1.4 License Units and License Unit Requirement Tables

Each LURT has a rating, in license units, for all available (and appropriate) processors. For example, the LURT for layered products includes the name of every processor that can run OpenVMS and associates a number of license units with each. The LURT for OpenVMS workstations includes the names of all OpenVMS workstations and their ratings. When Digital releases new processors, OpenVMS updates the tables as part of the new processor support.

2. LMF determines the model name of your processor. Although this manual generally refers to computer systems as processors, LMF actually identifies a system by its **System Marketing Model (SMM)** name, which is the model name of a computer system as used in marketing and pricing. The SMM is generally the name on the front panel of the processor cabinet.
3. LMF locates the SMM in the appropriate LURT and selects the value that specifies the number of units required for the named SMM and type of license.
4. LMF compares the number selected from the LURT to the number of units registered for your product license. If you have registered a value sufficient for your license and processor, the license is loaded successfully with the LICENSE LOAD command.

Table 1-1 shows an example of a License Unit Requirement Table. See the OpenVMS release notes or installation documentation for current LURTs.

Table 1-1 Sample License Unit Requirement Table F (Layered Products)

System Marketing Model	Number of License Units Required
VAX 8800	2500
VAX 8700	1800
VAX 8650	1500
.	.
.	.
.	.
MicroVAX A	135
MicroVAX 2000	135

The number of license units registered with any license should match or exceed the number of license units required for the specified product to run on the specified processor. For example, when you obtain a license for Digital Pascal to run on a VAXstation 3100 system, that Pascal license must specify at least the same number of license units as the LURT requires for a VAXstation 3100 system. The same Pascal license may not provide enough license units to authorize use of Pascal on a VAX 9000 system. The size of the license to run a software product on an OpenVMS Cluster environment must reflect the total number of concurrent users or processes and the processors on which the product will run.

Not all licenses have a specific number of units. Some licenses specify zero units, which is equivalent to unlimited units. Adjust the unit number as needed and as allowed by your license contract.

1.5 VMSLICENSE.COM

LMF provides the command procedure VMSLICENSE.COM to register, manage, and track software licenses. VMSLICENSE.COM provides a menu-driven interface that allows you to easily register and load your licenses. VMSLICENSE.COM executes LICENSE commands, which you can also issue manually. See Appendix A for a full description of each LICENSE command.

1.6 Types of Licenses

Different types of software product licenses enable you to allow access to each product in ways that range from access for a specific user on a specific processor to general access for all users on all processors within an OpenVMS Cluster.

Table 1–2 describes the licenses that LMF supports.

Table 1–2 Types of Licenses

Type	Identification on PAK	License by	See
Availability	Availability Table Code has a nonzero value.	Processor type (Requires Key Option ALPHA or VAX_ALPHA to load on Alpha processor.)	Section 1.6.1
Activity	Activity Table Code has a nonzero value.	Concurrent uses (not users)	Section 1.6.2
User	Activity constant and Key Options: USER.	Concurrent users (not uses)	Section 1.6.4
Personal Use	Activity constant and Key Options: RESERVE_UNITS.	Named user	Section 1.6.3

The license descriptions that follow provide information to help you understand and manage the product authorization process on VAX or Alpha computers using LMF, rather than to help you order software licenses. Digital provides licenses in many ways that may not always correspond to the examples in this manual. Check with your Digital support representative for ordering information, and check the terms and conditions of your license contracts for restrictions.

1.6.1 Availability Licenses

An **Availability License** makes a product available to all the users of a system. LMF can load a product when the number of license units specified by a license matches or exceeds the rating of the current processor. To authorize full availability on a processor, LMF checks the Availability Table Code on the registered license and interrogates the LURT to determine the rating of the processor. If the registered license provides enough license units, LMF loads the license, making the product available to all users on the named system.

For example, the PAK for fictional layered product ALLSUM provides 1000 license units (Number of Units: 1000) and refers to LURT F, (Availability Table Code: F). When you register and load the license, LMF selects LURT F and checks whether the size of the license is at least as big as the number of license units required by the current processor. If so, LMF authorizes full availability to ALLSUM on the current processor.

In addition to authorizing use on a processor, LMF allocates the required number of units to the processor that loads the license. If a 1000-unit Availability License is registered in a common OpenVMS Cluster environment License Database, LMF can allocate a total of 1000 license units among several processors. For

Overview

1.6 Types of Licenses

example, LMF can allocate the 1000 units to one processor that requires 500 units, one that requires 300, and a third that needs 200 license units. This is known as **license sharing**.

With an Availability License, LMF allocates license units to a processor when you load a license. LMF returns the license units when you use the LICENSE UNLOAD command or when the system is shut down.

Note

You can load an availability license on an Alpha system only if the PAK contains Key Option: ALPHA or Key Option: VAX_ALPHA. In addition, a license authorized by a PAK with Key Option: ALPHA can be loaded only on an Alpha system. However, the PAK can also safely reside in a License Database shared by both VAX and Alpha systems, and you can perform your License Database tasks from either a VAX or Alpha system.

1.6.1.1 Providing Sufficient License Units

The license you register in the License Database should provide enough license units to satisfy the requirements specified in the LURT. Before you purchase a license, work with your software representative to assess your software and hardware requirements and ensure that you obtain a license of the correct size.

For standalone systems (including multiprocessors), Digital offers licenses that exactly match the license requirements of a system. That is, there is a license size that matches each LURT entry.

Sometimes, users with multiple standalone systems cannot match their licenses to meet every circumstance. For example, you may manage two standalone processors: VAXBIG, which requires a 700-unit license, and VAXMID, which requires a 400-unit license. If you have one 700-unit license, you can load it on either processor (but not both). If you have one 400-unit license, you can load it only on VAXMID. You can, however, still register the smaller license in the License Database of VAXBIG.

Note

If your license specifies the /MOD_UNITS option, you can change the size of the license (see Section 2.6.2).

1.6.1.2 Providing More License Units

You may need to provide more license units than are currently registered in the License Database for a product. For example, you cannot load a 400-unit license on a processor that requires 700 units. If you need more license units than are currently available, contact your software representative, who may recommend one of the following:

- A new license that provides all the required license units. This may involve deleting the current license (with the LICENSE DELETE command).
- Another license for the same product that provides 300 additional license units. If your license agreement allows it, you can register both licenses. LMF may combine the license units to produce the equivalent of a 700-unit license. For a description of license combination, see Section 1.7.

1.6.1.3 Providing Availability in an OpenVMS Cluster Environment

To provide full availability for a product in a cluster environment with a common License Database, you must register licenses with a total of license units at least as large as the total license unit requirements of all the processors. For example, if the cluster consists of three VAX 8800 processors, each of which requires 1200 license units to run a specific product, you must register at least 3600 license units (1200 times 3) to provide full product availability across the cluster environment.

If you do not need product availability clusterwide, you can register licenses with total license units to authorize use by individual nodes. For example, in a cluster with three processors each requiring 1200 license units, a 1200-unit license allows any one node to run the product, and a 2400-unit license allows any two cluster nodes to run the product concurrently. You can also use the LICENSE MODIFY command to allow or deny access to specific cluster nodes (see Section 2.6.2).

Note that you cannot always manage licenses as previously described. For example, some licenses restrict a product to a certain processor type, and other licenses with the NO_SHARE option cannot share license units. As always, check the terms and conditions of your license contract.

1.6.1.4 Providing More Availability

If you change the configuration of an OpenVMS Cluster by adding a processor, or upgrading a processor to a more powerful one, you may need to increase the number of available license units. You can provide more license units in the following ways:

- Register additional licenses.
If you choose to register an additional license (sometimes referred to as an additive PAK), you gain flexibility; the new license can be moved to different cluster environments or standalone systems independently. When you register the new license, LMF automatically combines the license size of the new license with the license size of the existing license to establish a new, higher level of license authorization. Management of separate licenses for the same product can be somewhat more complicated, however.
- Replace the current license with a larger one.
A new license with more license units (sometimes referred to as a replacement PAK), is easier to manage but is not as flexible, because it cannot be split among different cluster environments or standalone systems. You typically delete or disable the older license, then register the new one.

Your software representative can help you choose the option that fits your needs.

1.6.2 Activity License

An **Activity License** defines the number of concurrent *uses* allowed for a product at any one time. Each product defines an activity as either an interactive user, a running process, or a job. For example, when you register a 4-Activity License, LMF authorizes four concurrent uses of the product. Each time the product is invoked by an activity, LMF checks whether there are sufficient license units available to use the product on the current processor, and if so, allocates the license units to that activity, reducing the number available to additional activities. When all license units are allocated, no new activity can invoke the product until an activity terminates use of the product (thereby deallocating license units).

Overview

1.6 Types of Licenses

As with an Availability License, an Activity License authorizes use through license units and LURTs. A product may require a certain number of license units per activity, regardless of processor. For example, a 4-Activity License for a product that requires 100 license units per activity has a size of 400 license units, and allows up to four activities, whether on a MicroVAX system or a VAX 8800 system. The license unit requirement of the product is designated on the PAK as Activity Table Code: Constant=100.

Other products require a certain number of license units per activity on a particular processor. A 4-Activity License for a product that requires 25 license units per activity on a VAX 8800 system requires 100 license units. The same 100-unit license provides five concurrent uses on a processor that requires only 20 units per activity (of that product).

One primary difference between an Availability License and an Activity License is the time at which LMF checks the number of license units authorized by a license, as follows:

- The Availability License allows unlimited access to the product after you successfully load the license on a processor.
- The Activity License requires allocating available license units each time the product is invoked, and denies access when the activity limit is reached.

1.6.2.1 Providing Enough License Units

As with Availability Licenses, you should try to match processor, product, and license to your user requirements. Software vendors offer a variety of licenses that can match the license requirements of your users and your system. Before you obtain a product license, consult your software representative to define your software and hardware requirements to ensure that you obtain a license of the correct size.

The license you register in the License Database should provide enough license units to allow some predetermined number of activities access to the product. For example, if a software product requires 25 license units per activity on your processor and PAKs come in 4-Activity increments, your license may provide units in a multiple of 100. Note that a 120-unit license provides no more use than a 100-unit license on such a processor.

Different processors can have different license unit requirements per activity. Therefore, the number of users authorized by a license varies according to the processor used. For example, you may manage the following standalone processors:

- VAXBIG, which requires 25 license units per activity to authorize a product
- VAXMID, which requires 20 license units per activity to authorize a product

If you obtain a 125-unit Activity License for VAXBIG, you can temporarily move that license (with the LICENSE COPY command) to VAXMID when you shut down VAXBIG for maintenance. The 125-unit license, which allows 5 concurrent activities on VAXBIG, provides 6 concurrent activities on VAXMID. Note that you can also move an 80-unit (4-Activity) license originally intended for VAXMID to VAXBIG. However, on VAXBIG, the license provides access to only 3 activities.

As with Availability Licenses, you can register a license in the License Database even if that license cannot be successfully loaded. For example, if you register a 40-unit license that provides product access to two activities on a MicroVAX A processor, the same license does not allow access to any activities on a processor that requires 50 units per activity.

1.6.2.2 Providing More License Units

You may need to provide more license units than are currently registered in the License Database for a product. Each time a user is denied access to a product because of insufficient license units, LMF produces the following message:

```
-LICENSE-F-EXCEEDED, attempted usage exceeds active license limits
```

Analyzing the frequency of these messages can help you determine your license needs.

If you need additional Activity License units, contact your Digital support representative, who may recommend one of the following:

- A new license with more license units. This may involve deleting the current license (with the LICENSE DELETE command).
- Another license for the same product that provides additional license units. If the terms of your license contract allow it, you can register the two licenses. This allows LMF to combine the license units, providing more product use. For a description of license combination, see Section 1.7.
- A different type of license. Some products offer Activity, Availability, Personal Use, and User licenses. Changing to different type of license may provide greater access to the product.

1.6.2.3 Sharing License Units in an OpenVMS Cluster Environment

Typically, all cluster activities can access a product that has an Activity License registered in the common License Database. If your PAK specifies a constant number of license units per activity regardless of processor size, the cluster always provides access to the same number of activities. A 4-Activity License provides access to 4 activities whether the cluster has 1 node or 12 nodes, 1 MicroVAX system or 12 VAX 9000 systems.

In other cases, an Activity License may not specify a constant number of license units per activity on all processors. Because the Activity License unit requirement can be different on each processor, the number of available activities depends on the processors involved.

For example, a particular Activity License might provide access to any 12 activities for a product on an OpenVMS Cluster with three VAX 8200 processors. If you add a processor to the cluster that has a higher license unit requirement (than a VAX 8200), the number of concurrent uses allowed can decrease, because LMF allocates more license units per activity of the product on the additional processor. You can modify the Activity License (using the LICENSE MODIFY command) to include or exclude specific nodes.

Note also that when the system starts up, LMF, by default, loads any licenses that do not have include or exclude lists. For predictable license loading, limit access with a LICENSE MODIFY/EXCLUDE or LICENSE MODIFY/INCLUDE command for each license that can be combined when licenses are loaded.

1.6.3 Personal Use License

A **Personal Use License** designates the names of specific users for unlimited use of a product. Before you load the license, you specify the users allowed access. LMF adds these users to a **reservation list**, which is checked before granting access to each user who tries to invoke the product. A PAK for a Personal Use License specifies RESERVE_UNITS in the Key Options field. This license shares some characteristics with both Availability and Activity Licenses.

Overview

1.6 Types of Licenses

Although a personal use PAK includes an Activity Table Code, it does not limit access to concurrent use. While an Availability License authorizes product use by processor, a Personal Use License authorizes product use by user name. LMF processes a list of authorized users when the license is loaded. After the license is loaded, any user on the list can access the product.

To calculate the allowed number of names on the reservation list, LMF divides the number of license units by the constant value listed in the Activity Table Code field of the PAK. If you register a 400-unit Personal Use License with a constant value of 100, LMF authorizes four specifically named users to access the product. If more than four names are associated with the license, LMF rejects extra names from the reservation list and denies access when those users attempt to access the product.

Personal Use Licenses are subject to combination rules that allow long lists of authorized users. See Section 1.7 for information about combining Personal Use Licenses.

Each Personal Use License must have an associated reservation list that specifies the name of each user to be authorized access to the product. You cannot load a Personal Use License that does not have an associated reservation list with at least one user name. See Section 2.6.3 for information about controlling access to licenses with reservation lists using the LICENSE MODIFY command.

1.6.4 User License

The **User License** shares some characteristics with the Activity and Personal Use Licenses, as follows:

- Each user is counted once towards the total number of concurrent users allowed. The terms of your license determine that total number.
- Each user granted access has unlimited access to the product until exiting the last invocation, but is counted against the authorized limit only once.
- Once the user has exited the last invocation of the product, renewed access will be denied if the total number of users of the product has reached the maximum authorized by the license.

As specified in the terms of your license, *users* can be people, disks, queues, applications, and others.

1.6.5 Group License

A **Group License** authorizes access to a group of software products — usually related — that are licensed as one product. This enables you to license a group of products by registering only one PAK. A Group License can be any type of license: Availability, Activity, Personal Use, or User License.

All LICENSE commands use the group name for the *product-name* parameter instead of the individual product names. For example, a Software Group called COMPILER_1 might include Digital Fortran, DEC Pascal, and DEC COBOL. You register the Group License as COMPILER_1 with one PAK and enter all LICENSE commands using COMPILER_1 as well.

Group PAKs do not look different from single product PAKs. For information about the products licensed by a group PAK, see the Software Product Description (SPD) for the group product.

1.7 License Combination

License combination allows LMF to create large licenses by adding together the license units of multiple licenses. For example, two 50-unit Availability Licenses become equal to one 100-unit Availability License. Ten 100-unit Personal Use Licenses become equal to one 1000-unit Personal Use License.

License combination and loading are controlled by both the terms of your PAK and options you set with LICENSE MODIFY/COMBINE and LICENSE MODIFY /NOCOMBINE commands.

LMF Version 1.2 automatically combines some licenses by default.

1.7.1 Licenses That Can Be Combined

When a processor loads a license, LMF scans the License Database for all **combinable licenses** and makes a pool of license units available for use. Licenses are combinable if they have matching data in each of the following data fields:

- Product name
- Producer
- Availability
- Activity
- Key Options: RESERVE_UNITS, USER, NO_SHARE (assigned node must match), ALPHA, or VAX_ALPHA
- Product Token
- Hardware-ID

VAX Availability, ALPHA Availability, VAX_ALPHA Availability, User, Activity, and Personal Use licenses are different types of licenses. Therefore, they do not combine.

LMF matches any two empty data fields and, in the Availability and Activity fields, also matches the entry CONSTANT=0 with an empty data field. Licenses with the NO_SHARE option can combine, but they must have matching include lists that assign each license to the same node. This is the only time either an include list or an exclude list has an effect on license combination.

By default, LMF does not automatically combine otherwise combinable licenses if any one of the following attributes do not match:

- Termination Date
- Release Date
- Version
- Reservation List

If two or more licenses are combinable except for the above attributes, you can force LMF to combine them with the following command:

```
LICENSE MODIFY product-name /COMBINE
```

Overview

1.7 License Combination

1.7.2 Include, Exclude, and Reservation Lists

If you register a combinable license without an include or exclude list, any processor can load the license with access to the entire pool of combined license units, with the following results:

- The entire pool of Availability License units becomes available to the processor that loads the license. LMF allocates the number of license units required for each processor as it loads the license. The appropriate LURT defines this number.
- The entire pool of Activity License units can be available to any user (or activity) that loads the product. LMF allocates the license units as each user accesses the product.

By default, when combining Activity Licenses, LMF combines those without reservation lists into one license without a reservation list, and those with reservations lists into one license with a reservation list that combines the separate reservation lists.

By default, when combining User Licenses, LMF combines those without reservation lists into one User License without a reservation list, and those with reservations lists into one User License with a reservation list that combines the separate reservation lists.

By default, when combining Personal Use Licenses, LMF combines any reservation lists associated with each license into one large reservation list that applies to all the combined licenses.

1.7.3 Termination Dates and Version Numbers

With forced combination of multiple licenses, LMF sets the termination date, release date, and version number of the combined license to the earliest dates and version numbers that apply to the individual licenses being combined. The following table shows the combined license that results from forced combination of two licenses:

	License 1	License 2	Combined License
Version Number	2.3	2.0	2.0
Release Date	1-JAN-1995	30-NOV-1996	1-JAN-1995
Termination Date	1-JAN-1999	30-SEP-1997	30-SEP-1997

This chapter provides details about the tasks involved in managing software licenses. Topics covered include:

- Steps to license registration (see Section 2.1)
- Managing the License Database (see Section 2.2)
- Getting a Product Authorization Key (PAK) (see Section 2.3)
- Registering licenses (see Section 2.4)
- Loading a license (see Section 2.5)
- Managing licenses after registration (see Section 2.6)

In addition, this chapter contains a clarification about using logical name LMF\$DISPLAY_OPCOM_MESSAGE (see Section 2.7).

2.1 Steps to License Registration

To license and use many software products on the OpenVMS operating system, follow at least these four steps:

1. Obtain a PAK for your product.

This is usually a hardcopy or electronic document containing information similar to that shown in Example 2–1. Order it from the software license issuer or software product producer.

2. Register information from the PAK into the License Database.

Use command procedure VMSLICENSE.COM to prompt for license registration information or enter the LICENSE REGISTER command directly. Example 2–3, produced with a LICENSE LIST command, shows a license registered in the License Database. In this manual the PAK information registered in the License Database is called a license.

3. Ensure that the system loads the registered license.

LMF requires that a registered license be loaded before you can use the product. When you register a license with VMSLICENSE.COM, you can confirm an option to load the license automatically. If you register a license with the LICENSE REGISTER command, you must also load it with a LICENSE LOAD command in order to use the product. At system startup, LMF automatically loads registered licenses.

4. Install the product corresponding to the license.

Although the terms and conditions of license contracts vary, generally a license correlates with a particular release of a product. Because LMF allows products to check the License Database for a properly registered license, you must match the license to the product.

Using LMF

2.1 Steps to License Registration

After performing these steps, you can modify the license for a system or involve multiple systems in a licensing scheme (if your license agreement allows it).

For example, you want to restrict to a specific node a license used in an OpenVMS Cluster environment. If you register a license that uses the NO_SHARE option (an OpenVMS operating system license, for instance), assign the license to a specific node. Either enter a LICENSE MODIFY/INCLUDE=*node-name* command or respond to the prompt for a System Communications Services (SCS) node name in VMSLICENSE.COM (see Section 2.6.2 for details).

2.2 Managing the License Database

LMF stores all information about licenses in the License Database. By default, LICENSE commands refer to the default license database, and you usually do not need to know the name and location of the database. However, for system management reasons, you may need to move the database. This section describes techniques for accessing license information and moving the license database.

Most of the data fields in the License Database correspond to the LICENSE qualifiers or to responses to command procedure prompts. For example, the authorization field contains the data entered with the following command:

```
$ LICENSE REGISTER /AUTHORIZATION=string product-name
```

If you enter USA1234 for the string, USA1234 becomes the data in that field.

When you first register a license, you create the first record with data matching your PAK. When you enter other LICENSE commands, LMF creates new records to include any changes you make. For example, when you enter a LICENSE MODIFY command, LMF creates a new record marked with the new information, including a notation that the license was modified.

For performance reasons, License Database information is duplicated in memory while your system is running. LICENSE commands impact the database stored on disk. To update the License Database information in memory, use the LICENSE LOAD or LICENSE UNLOAD commands.

2.2.1 Database Location

If you move the database to another directory or disk, or rename the database file, you must either define the logical name LMF\$LICENSE at the system level to point to the new database, or you must use the /DATABASE=*filespec* qualifier with all LICENSE commands. Place permanent systemwide logical name definitions in the file SYS\$COMMON:[SYSMGR]SYSLOGICALS.COM.

In some circumstances, OpenVMS licenses are in a separate License Database located in SYS\$SPECIFIC:[SYSEXE]LMF\$SYSTEM.LDB.

If you have multiple system disks in an OpenVMS Cluster environment where all the systems can access one of the system disks, put your common License Database on the readable disk. For any systems that boot from a separate system disk, you must redirect LMF to the License Database. Define the logical name LMF\$LICENSE to be the disk where the database exists.

If you have multiple system disks in an OpenVMS Cluster environment where some systems cannot access one of the system disks, you must keep separate identical License Databases. Whenever one database is modified, you must copy it to update the other databases.

Digital recommends you back up the License Databases after every modification.

2.2.2 History Records

Your system maintains history records. Each history record contains an exact copy of the license record before modification, the LICENSE command used to modify the record, and the date and time of modification.

History records accumulate over time and provide a comprehensive audit trail of all modifications you make to the License Database. Most software issuers, including Digital, require that you retain this information to demonstrate that you are complying with license terms and conditions.

To display history information, enter the following command:

```
$ LICENSE LIST /HISTORY
```

To create a hard copy, enter the following command:

```
$ LICENSE LIST /HISTORY /OUTPUT=LICENSE.LIS  
$ PRINT LICENSE.LIS
```

Over time, LICENSE commands, including the LICENSE START command issued automatically during system startup, might take longer than usual to execute. This could be due to an accumulation of license history records in the License Database.

If you notice delays, Digital recommends that you purge the history records in your active License Databases, but only after first preserving this information in one or more backup locations. Use the DCL command COPY or the Backup utility to make a copy of the License Database, thereby preserving the current version of the License Database, including history records.

To purge history records, enter the following command:

```
$ LICENSE DELETE /STATUS=EXTINCT *
```

Caution

Ensure that you do not omit the /STATUS=EXTINCT qualifier in the above command. If you do, all license records are deleted, leaving your License Database empty.

LICENSE DELETE deletes all history records, making them invisible to subsequent LICENSE commands.

Creating a new, compressed version of the License Database reclaims the disk space formerly occupied by the now deleted history records. To create a compressed License Database, use the DCL Convert utility (CONVERT).

2.3 Getting a Product Authorization Key (PAK)

Generally, you obtain both a PAK and the product from a representative of a company that distributes software. You order a PAK just as you order another product from Digital or another company. Digital provides PAKs on paper certificates, traditional media, compact disc read-only memory (CD-ROM), by telephone or network so that you can register product data in the License Database. LMF needs specific values from a PAK to identify the source of the PAK and the source of a product.

Using LMF

2.3 Getting a Product Authorization Key (PAK)

A PAK comes from a **PAK issuer**—the LMF name for the entity that supplies the PAK. Currently, most licenses specify DEC for the PAK issuer, and currently DEC is the default character string when you register a PAK with VMSLICENSE.COM. A PAK issuer string can also be DEC-USA or DEC-EUROPE to differentiate regions or department within Digital. Other software vendors provide their own PAK issuer strings with their licenses. LMF uses the string to differentiate between different sources of licenses.

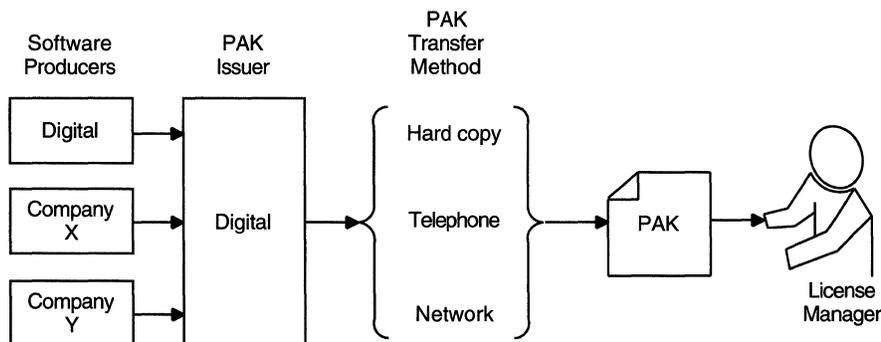
Digital may distribute and issue a PAK for a product that it does not produce. Thus, LMF also uses a string that identifies a **software producer**. A producer is the company that supplies the software product. Generally, a producer and a PAK issuer are the same. The current default producer name when you register a PAK with VMSLICENSE.COM is DEC.

The OpenVMS operating system and LMF use PAKs to authorize most products for use. For example, after you install OpenVMS, you may have all the software required to use the System Integrated Products (SIPs) such as networking, RMS Journaling, and Volume Shadowing. To enable a SIP, register its PAK and load the license (there is no separate installation media). Even when you receive multiple software products on one Digital CD-ROM, register a PAK for each product to enable the software.

Some products follow the older product distribution and license approach, providing installation kits that include distribution media and documentation. If a kit does not include the PAK, order it separately.

Figure 2-1 illustrates the PAK transfer process.

Figure 2-1 PAK Transfer Methods



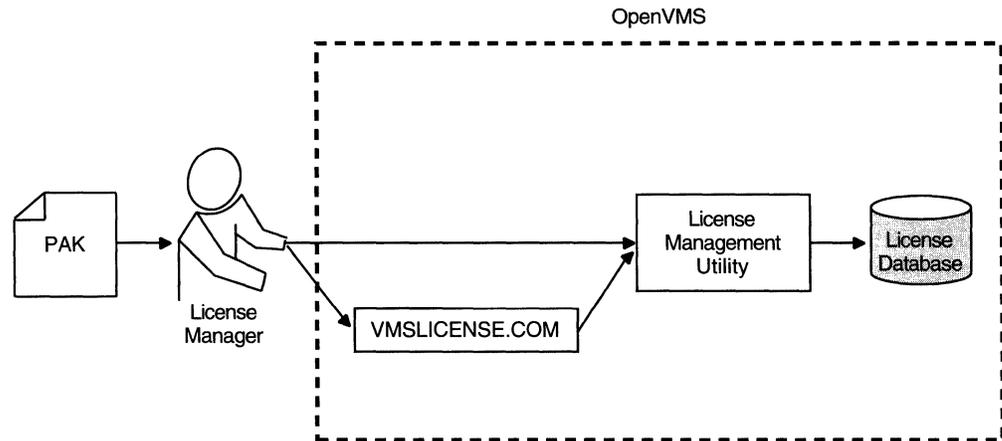
ZK-7455-GE

2.4 Registering Licenses

To run most Digital software products, including the OpenVMS operating system, you must first register the product license in the License Database and then load the registered license. In addition, many third-party vendors of OpenVMS layered software also require you to use LMF to complete the same licensing tasks for their products.

Section 2.1 describes the registration options and presents examples of registration. Figure 2-2 illustrates the routes from a PAK to the License Database.

Figure 2-2 From a PAK to the License Database



ZK-7454-GE

2.4.1 When To Perform Registration

Most Digital software that runs on OpenVMS systems and many third-party software layered products use LMF. To check a product's licensing requirements, see its installation manual or release notes. These documents explain which products use LMF registration.

If a product uses LMF, you must obtain a PAK, which includes the appropriate data for you to enter. A PAK typically contains the information shown in Example 2-1.

Example 2-1 Typical PAK Information

```
ISSUER: DEC
AUTHORIZATION NUMBER: USA126087
PRODUCT NAME: CRYPTICALMENT
PRODUCER: DEC
NUMBER OF UNITS: 460
VERSION: 7.1
PRODUCT RELEASE DATE:
KEY TERMINATION DATE: 31-DEC-1997
AVAILABILITY TABLE CODE: E
ACTIVITY TABLE CODE:
KEY OPTIONS: MOD_UNITS
PRODUCT TOKEN:
HARDWARE I.D.:
CHECKSUM: 1-BGON-IAMA-GNOL-AIKO
```

2.4.2 Registration and Installation

Follow the licensing and installation procedure provided with each product. You can save time if you consider the following variations and consequences for product installation and license registration:

- If you register a license before you install a product, the product installation can be somewhat faster. You should register the license first, even though some products may allow installation first.

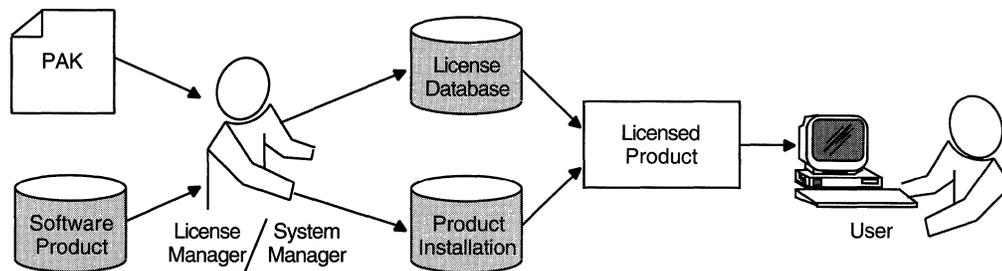
Using LMF

2.4 Registering Licenses

- If you start to install a product and realize you need to register a license for it first, you may be able to continue the installation but not use the product. After you register and load the license, you can use the product. Be sure to reply correctly to any licensing questions during the product installation. Check your product installation guide for specific restrictions.
- To add a new node to an OpenVMS Cluster, you can register the new OpenVMS license before you add the node. You do not usually have to install the product again, unless the new node uses a new system disk.
- If you are upgrading an OpenVMS Cluster environment, you may want to register all the OpenVMS licenses at one time after one node is operating. This eliminates some messages when the other nodes start up and keeps your nodes more available for interactive use. Typically, you assign licenses by processor type. For example, you should not assign a license intended for a VAX 8200 system to a VAX 8800 system.

Figure 2-3 illustrates the license registration and product installation route for processors running the OpenVMS operating system and for layered products.

Figure 2-3 The PAK and Software Routes to a License



ZK-7453-GE

2.4.3 Registration Methods

Licenses are registered in the License Database in one of the following ways:

- Before you install a product,¹ enter PAK information in response to prompts from `SYS$UPDATE:VMSLICENSE.COM`. This command procedure provides some default data and a menu-driven interface to help register the license.
- Before you install a product, enter PAK information with a `LICENSE REGISTER` command. The qualifier descriptions for the `LICENSE REGISTER` command describe the meaning of the PAK information. Each piece of PAK data correlates to a `LICENSE REGISTER` command qualifier.
- Some products register their licenses during their own installation procedure.

Unless you have a special circumstance, choose the registration method you prefer or the one recommended by your installation guide.

After a license is registered, it must be loaded to make it known on the current system. Section 2.1 describes the primary methods for registering and loading your licenses.

¹ With the OpenVMS operating system, you start the installation first. Although Digital does not recommend it, you can install some software products first and license them later. See your software product's documentation for details.

2.4.4 Using VMSLICENSE.COM

The following steps show how to use the VMSLICENSE.COM procedure to register a license for a product called CRYPTICALMENT. The PAK information is shown in Example 2-1.

1. Log in to the system manager's account, SYSTEM.
2. Enter the following command and press Return:

```
$ @SYSS$UPDATE:VMSLICENSE
```

The procedure displays the following menu:

```
VMS License Management Utility Options:
```

1. REGISTER a Product Authorization Key
2. AMEND an existing Product Authorization Key
3. CANCEL an existing Product Authorization Key
4. LIST the Product Authorization Keys
5. MODIFY an existing Product Authorization Key
6. DISABLE an existing Product Authorization Key
7. DELETE an existing Product Authorization Key
8. COPY an existing Product Authorization Key
9. MOVE an existing Product Authorization Key
10. ENABLE an existing Product Authorization Key
11. SHOW the licenses loaded on this node
12. SHOW the unit requirements for this node
99. EXIT this procedure

```
Type '?' at any prompt for a description of the information  
requested. Press Ctrl/Z at any prompt to return to this menu.
```

```
Enter one of the above choices [1]
```

3. Enter 1. The procedure displays the following message:

```
* Do you have your Product Authorization Key? [YES]:
```

4. Enter Y. The procedure displays the following information and prompts:

```
Use the REGISTER option to add a new license to a license  
database. A Product Authorization Key (PAK) provides the product  
name and information you need to register the license. You must  
enter all the information provided by your PAK exactly as it  
appears.
```

```
Issuer [DEC]:  
Authorization Number []:
```

5. Press Return to specify DEC as the issuer.

```
Enter USA126087 for the authorization number that appears on the PAK.  
The procedure prompts for the product name:
```

```
Product Name []:
```

6. Enter CRYPTICALMENT for the product name string that appears on the PAK. The procedure prompts for the producer:

```
Producer [DEC]:
```

Using LMF

2.4 Registering Licenses

7. Press Return to specify DEC as the producer. The procedure prompts for the number of units:

Number of Units []:

8. Enter 460 for the number of units. The procedure prompts for the version:

Version []:

9. Enter 7.1 for the version number from the PAK. The procedure prompts for the key termination date:

Key Termination Date []:

10. Enter 31-DEC-1997 for the key termination date. The procedure prompts for the following information:

Availability Table Code []:

Activity Table Code []:

11. Enter E for the Availability Table Code. Press Return after the Activity Table Code prompt. The procedure prompts for the following information:

Key Options []:

Product Token []:

Hardware-Id []:

12. Enter MOD_UNITS for the option after the Key Options prompt. Press Return after the Product Token Prompt and the Hardware-ID prompt. The procedure prompts for the checksum:

Checksum []:

13. Enter 1-BGON-IAMA-GNOL-AIKO for the checksum.

Note

The checksum string always begins with a number. The other 16 characters are always alphabetic characters from A through P.

The procedure displays the information you entered. For example:

Here is a list of the license information just entered:

```
Issuer: DIGITAL
Authorization: USA126087
Producer: DEC
Product Name: CRYPTICALMENT
Units: 460
Release Date:
Version: 7.1
Termination Date: 31-DEC-1997
Availability: E
Activity:
Options: MOD_UNITS
Token:
Hardware ID:
Checksum: 1-BGON-IAMA-GNOL-AIKO
```

Is that correct? [YES]:

14. Compare the information on the screen with the information on the PAK. If the information is correct, enter Y.

Note

If you enter PAK information incorrectly, you receive an error message, and the license is not registered. A checksum error can result when you enter incorrect information for the other items on the PAK. If you get an error, carefully check all the data that you entered.

If the information is incorrect, enter N.

When the procedure displays the following question, enter Y:

Do you wish to make corrections? [YES]:

15. To make corrections, the procedure asks all of the questions again but supplies the data just entered as defaults for each data field.
- If the procedure displays correct information, press Return.
 - If the procedure displays incorrect information, enter the new data.
 - If the procedure displays incorrect information that you wish to cancel without entering new data, enter the backslash (\) character.

If you entered all the information correctly, the procedure displays the following message:

```
Registering CRYPTICALMENT license in SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB...
```

If you entered some information incorrectly but do not choose YES to make corrections, the procedure may display the following message:

```
Registering CRYPTICALMENT license in SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB...
%LICENSE-F-BADCHK, checksum does not validate for CRYPTICALMENT
Please review all entered PAK data, including the checksum.
Do you wish to make corrections? [YES]:
```

To correct the data, enter Y.

If you enter an incorrect checksum string, the procedure responds as follows:

```
1-BGON-IAMA-GNOL-AIKO is not a valid license checksum string.
Press RETURN for more information
```

The license checksum is a 17-character verification string created by the PAK issuer for each PAK. The checksum string is presented in the format n-cccc-cccc-cccc-cccc, where n is an integer and c is a character from A through P. A PAK presents the checksum string with hyphen (-) characters for readability. Because the LMF does not count them for authorization, you can leave them out. Otherwise, you must enter the checksum string exactly as specified on your PAK.

If a default value is displayed and you wish to use it just press the RETURN key. To cancel the use of default data without entering new data, enter the backslash (\) character. The license checksum is a required field for the REGISTER and AMEND options.

Checksum []:

Enter the correct checksum at the prompt.

Using LMF

2.4 Registering Licenses

16. After the license is successfully registered, the procedure asks if you want to load the license on the current node, as follows:

Do you want to LOAD this license on this system? [YES]:

If you registered the PAK on a standalone system and you want to make the software available (active) immediately, enter Y. If you registered the license in an OpenVMS Cluster environment but do not want to make it available (active) on the current node, enter N. After you exit this procedure, you can enter the LICENSE LOAD command to load the license on the desired node.

If you enter Y and the license is successfully loaded, the procedure displays the following informational message and prompt:

```
%LICENSE-I-LOADED, DEC CRYPTICALMENT was successfully loaded with 460 units
```

```
VMS License Management Utility Options:
```

1. REGISTER a Product Authorization Key
2. AMEND an existing Product Authorization Key
3. CANCEL an existing Product Authorization Key
4. LIST the Product Authorization Keys
5. MODIFY an existing Product Authorization Key
6. DISABLE an existing Product Authorization Key
7. DELETE an existing Product Authorization Key
8. COPY an existing Product Authorization Key
9. MOVE an existing Product Authorization Key
10. ENABLE an existing Product Authorization Key
11. SHOW the licenses loaded on this node
12. SHOW the unit requirements for this node

99. EXIT this procedure

```
Type '?' at any prompt for a description of the information requested. Press Ctrl/Z at any prompt to return to the main menu.
```

```
Enter one of the above choices [1]
```

17. To register another PAK, enter 1. Then respond to the questions, again entering information from a license PAK.
18. Enter 99 to exit the procedure. You have registered the license for this product. The system may display an error message when the procedure attempts to load the license. This does not affect the license registration. Exit the procedure, and read the sections of this manual that describe loading a license. For example, read the LICENSE LOAD, LICENSE UNLOAD, and LICENSE MODIFY command descriptions.

2.4.4.1 Using Data Files with VMSLICENSE.COM

You can use VMSLICENSE.COM in batch mode as well as interactively. Instead of entering the PAK data interactively from your terminal or workstation, you can create a VMSLICENSE data file with an editor and enter data obtained from your product PAKs (Example 2-2 shows sample PAK data). You can then invoke VMSLICENSE.COM, specifying the name of the new VMSLICENSE data file as a parameter on the same command line. The procedure displays the license data and performs the requested operation or operations using data only from the VMSLICENSE data file.

You can create a file that registers multiple licenses. VMSLICENSE.COM exits only when it reaches the end of the VMSLICENSE data file.

2.4.4.2 Invoking VMSLICENSE.COM with a VMSLICENSE Data File

To pass a VMSLICENSE data file to VMSLICENSE.COM, use the following format:

```
$ @VMSLICENSE [license-option] vmslicense-data-file [license-database]
```

When you invoke VMSLICENSE with a data file, you must specify REGISTER as the *license-option*. You cannot use data files with any of the other options that are available when using VMSLICENSE interactively.

You can specify a License Database on the command line or in the data file. Any License Database you specify in the data file overrides a License Database you specify on the command line.

For example, you can pass the data in the sample VMSLICENSE data file, shown in Example 2–2, with the following command line:

```
$ @VMSLICENSE REGISTER NODES_A_AND_B_VMS_LICENSE.DAT
```

Example 2–2 Sample VMSLICENSE Data File

```
!
! License for AWESOME-OS on NODEA
!
DATABASE_FILE   = SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB
ISSUER          = DEC
PRODUCT NAME    = AWESOME-OS
AUTHORIZATION   = USA00326
UNITS           = 1200
VERSION         = 7.0
AVAILABILITY    = A
OPTIONS         = MOD_UNITS,NO_SHARE
CHECKSUM        = 1-DELM-EAHF-ONIH-MBAH
INCLUDE_NODE    = NODEA
!
<NEXT>
!
! License for AWESOME-OS on NODEB
!
AUTHORIZATION   = USA00327
UNITS           = 800
CHECKSUM        = 1-PATC-IDOH-EPOF-MOHG
INCLUDE_NODE    = NODEB
!
```

2.4.4.3 Creating VMSLICENSE Data Files

When you create a VMSLICENSE data file to be processed by VMSLICENSE.COM, you must follow these rules (refer to Example 2–2):

- Specify all PAK data as parameters in the form of DCL assignment statements. Use the following format:

```
parameter = parameter-value ! comment
```

Table 2–1 lists the only parameters allowed in the VMSLICENSE data file. You must use the exact parameter names, or VMSLICENSE ignores the line in the VMSLICENSE data file and returns an error message.

- Separate the end of one license registration from the beginning of another with the following special license data separator:

```
<NEXT>
```

Using LMF

2.4 Registering Licenses

- Precede each comment with an exclamation point; VMSLICENSE ignores everything to the right of an exclamation point when processing the line.
- You can separate words in the VMSLICENSE data file with any number of spaces or tabs.
- You can list parameters in any order in the VMSLICENSE data file.

Table 2–1 Parameters Used with VMSLICENSE.COM

Parameter	Description
ACTIVITY	License unit code that corresponds to a License Unit Requirement Table (LURT) or to a constant value.
AUTHORIZATION	String that, when used with the PAK issuer string, identifies the license you want to register.
AVAILABILITY	License unit code that corresponds to a LURT value or to a constant value.
CHECKSUM	Verification string of 17 characters created by the PAK issuer for each PAK with this format: <i>n-cccc-cccc-cccc-cccc</i> where <i>n</i> is an integer and <i>c</i> is an alphabetic character from A through P.
DATABASE_FILE	Location of the License Database to be used. The default file specification is defined by the logical LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system.
DATE	Product date specifying that the license authorizes use of all product versions released on or before the date.
HARDWARE_ID	Identification number of the hardware on which the product is licensed.
INCLUDE_NODE	Nodes in an OpenVMS Cluster environment that can access the licensed product.
ISSUER	Company that issued the PAK for this product.
OPTIONS	List of license options from a PAK.
PRODUCER	Company that owns this product.
PRODUCT_NAME	Product with a license to register. Use the product name exactly as it appears on the PAK.
RESERVE_LIST	Users (or activities) allowed to access the product license.
TERMINATION_DATE	Date on which the product license is no longer valid.
TOKEN	Product token string from a PAK.
UNITS	Number of license units from a PAK.
VERSION	Version limits from a PAK of the product for which you have a license.

2.4.4.4 Using VMSLICENSE.COM Default Value Rules

If you do not specify a value for a parameter in a VMSLICENSE data file, VMSLICENSE substitutes default values. The current default values are ISSUER=DEC and PRODUCER=DEC. All other license parameters have null values until you specify a value.

Using LMF 2.4 Registering Licenses

After you specify a license parameter in the VMSLICENSE data file, the parameter becomes the new default value until the next occurrence of the parameter sets a new default value. To set or reset the value of a parameter to null, use a line of the following form:

```
parameter = " "
```

For example, if the most recent PAK data set the INCLUDE_NODE parameter to a specific node, reset the parameter to null for the current and subsequent PAKs by entering the following:

```
INCLUDE_NODE = ""
```

Example 2-2 shows how using default data can eliminate typing for the second PAK listed in the VMSLICENSE data file.

2.4.5 Using the LICENSE REGISTER Command

You can directly register (and load) a license with LICENSE commands. For example:

```
$ LICENSE REGISTER CRYPTICALMENT /ISSUER=DEC -  
  $ /AUTHORIZATION=USA126087 /PRODUCER=DEC /UNITS=460 -  
  $ /VERSION=7.0 /TERMINATION_DATE=31-DEC-1997 /AVAILABILITY=E -  
  $ /OPTIONS=(MOD UNITS) /CHECKSUM=1-BGON-IAMA-GNOL-AIKO  
$ LICENSE LOAD CRYPTICALMENT  
LICENSE-I-LOADED, DEC CRYPTICALMENT was successfully loaded with 460 units  
$
```

After you successfully register a license in the License Database (the default file specification is SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB), you can enter the LICENSE LIST/FULL command to display the data in the database (see Example 2-3).

Example 2-3 Displaying License Database Information

```
$ LICENSE LIST /FULL /AUTHORIZATION=USA126087 CRYPTICALMENT
```

Use CTRL/Z to exit, PF3-PF4 for Previous-Next Screen and Arrow Keys to scroll.
License Management Facility V1.2

```
License Database File:  SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB  
Created on:             28-AUG-1996  
Created by user:       GRAHAM  
Created by LMF Version: V1.2
```

```
-----  
Issuer:                DEC  
Authorization:         USA126087  
Product Name:         CRYPTICALMENT  
Producer:             DEC  
Units:                460  
Version:              7.0  
Release Date:         (none)  
PAK Termination Date: 31-DEC-1997  
Availability:         E (System Integrated Products)  
Activity:             0  
Options:              MOD_UNITS  
Hardware ID:
```

(continued on next page)

Using LMF

2.4 Registering Licenses

Example 2-3 (Cont.) Displaying License Database Information

```
Revision Level:      1
Status:             Active
Command:            REGISTER
Modified by user:   GRAHAM
Modified on:        23-SEP-1996 14:32:23.41
```

When a license is successfully loaded with `VMSLICENSE.COM` or the `LICENSE LOAD` command, LMF displays an informational message.

2.4.6 Displaying License Information

To display information from the License Database about specific licenses, enter the `LICENSE LIST` command.

To display information in memory about loaded licenses, enter the `SHOW LICENSE` command. This command displays licenses loaded on the current node, and displays any reservation list associated with each license.

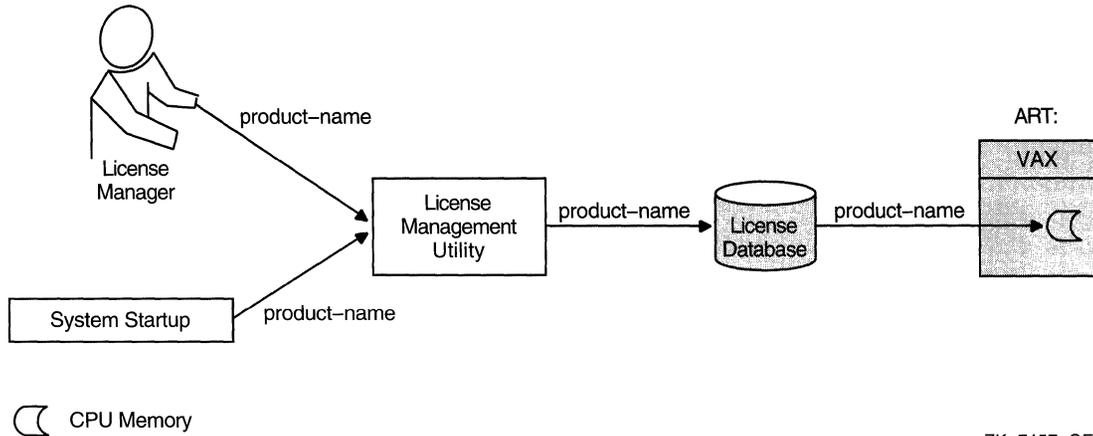
2.5 Loading a License

To allow users to access a product, you must load each registered license. Loading a license transfers data from the on-disk License Database into system memory. The following table shows the methods of license loading for LMF:

License Loading Options	
Status of Registered License	Loading Method
Registering with <code>VMSLICENSE.COM</code>	Confirm the load option.
Registered with the <code>LICENSE REGISTER</code> command	Use the <code>LICENSE LOAD</code> command.
Previously registered system starting up	System loads the license automatically.

Figure 2-4 illustrates the license loading process for a standalone system. Whether the license manager enters the `LICENSE LOAD` command or the system automatically loads all registered licenses during startup, the license information is transferred to the system memory of node ART.

Figure 2-4 Loading a License



2.5.1 Loading Licenses in an OpenVMS Cluster Environment

In an OpenVMS Cluster environment, multiple nodes load licenses from a single common database. LMF provides this capability to let you share licenses as well as control access to a product on a node-by-node basis (provided this is allowed by the terms and conditions of the license).

Typically, any node can load a license registered in the common License Database. For predictability, security, performance, or economic reasons, you can limit which nodes can access a product license intended to be shared among nodes. In an OpenVMS Cluster environment, you control access by modifying the license's **include list**. Use the LICENSE MODIFY command and either the /INCLUDE or /EXCLUDE qualifier to specify which cluster nodes can load the license. By changing the include list, you can create the effect of moving a product from node to node. Section 2.6.2 describes the process in detail. Note that you cannot share licenses that have the NO_SHARE option. They must be assigned to a single node.

You can also control which users in an OpenVMS Cluster environment can access a product license. Control access by modifying the license's reservation list. Use the LICENSE MODIFY command with the /RESERVE qualifier to specify which users can access the license.

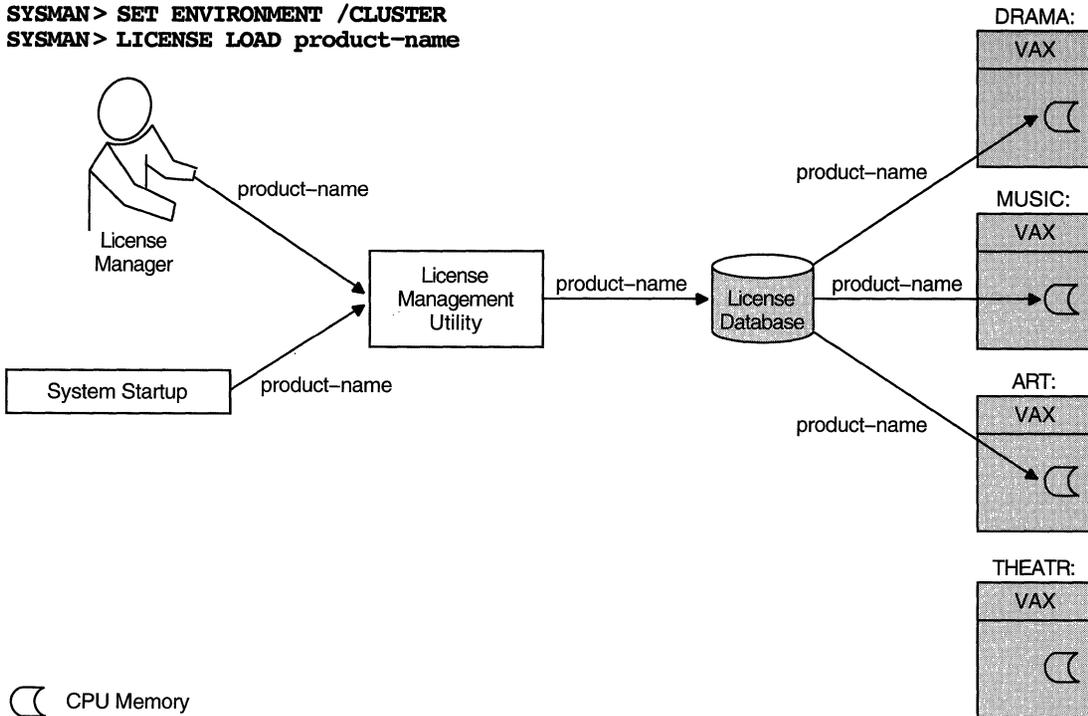
In an OpenVMS Cluster environment, license loading involves transferring, or loading, license information from the on-disk License Database into system memory of the current node. The number and combination of licenses for an OpenVMS Cluster environment can be complex, which in turn makes the loading process complex and sometimes confusing. The general rule is that nodes specified on include lists or not specified on exclude lists of the LICENSE MODIFY command can load a registered license. As each system starts up, it loads any licenses it has access to. If you need to load a license for all assigned nodes of a running cluster, you can use the OpenVMS System Management utility (SYSMAN), which is described in the *OpenVMS System Manager's Manual*. Figure 2-5 illustrates the loading process in an OpenVMS Cluster environment.

Using LMF

2.5 Loading a License

Figure 2-5 Loading a License in an OpenVMS Cluster Environment

```
$ LICENSE MODIFY /EXCLUDE=THEATR product-name
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT /CLUSTER
SYSMAN> LICENSE LOAD product-name
```



ZK-7456-GE

See Section 2.6.2 for further details of license loading.

2.6 Managing Licenses After Registration

After you register licenses, you can manage them. While system managers can perform these tasks, managers of cluster environments and large multiple-user systems may also have an interest in managing products and product licenses.

For example, a cluster manager can use LMF to control which nodes can access a database product using important data (see Section 2.6.2). All system managers, however, may want to monitor license data using **LICENSE** commands.

VMSLICENSE.COM, **LICENSE** commands, and the DCL command **SHOW LICENSE** can perform most license management tasks, including:

- Listing registered licenses
- Showing loaded licenses
- Making loaded licenses unavailable — either permanently or temporarily
- Transferring licenses from one database to another
- Modifying licenses — such as termination dates or reservation lists

2.6.1 Restricting Product Use

LMF provides the following commands and qualifiers for controlling access to licensed products.

2.6.1.1 Fastest Method

The quickest method to restrict access to products registered in the License Database is to unload the license with a LICENSE UNLOAD command. The product becomes unavailable to new users and unavailable to all users when the last process using the product finishes. The product remains inaccessible until you or the system reloads the license.

Although LICENSE UNLOAD is fast, it is not permanent. Typically, at system startup, LMF loads all the licenses registered in the License Database. In addition, a license that is loaded remains loaded until the system is shut down or until you intervene.

2.6.1.2 For Inactive Licenses

The following commands control license loading, which restricts product access. However, they do not unload loaded licenses or alter in-memory license data. For loaded licenses, use LICENSE UNLOAD.

- LICENSE DISABLE
Prevents the specified license from being loaded until you enter a LICENSE ENABLE command.
- LICENSE MODIFY/INCLUDE, LICENSE MODIFY/EXCLUDE
Makes the named nodes in an OpenVMS Cluster environment available or unavailable for license loading.
- LICENSE MODIFY/UNITS=*number*
 - Limits access when you set the unit number to be less than that required by your processor or cluster.
 - Use if your license has the MOD_UNITS option. Not all licenses have a specific number of units. Some licenses provide zero units, which is equivalent to unlimited units.
- LICENSE MODIFY/RESERVE=*user-name*
Makes the product available only to the users named in the reservation list. You must use this method if your license has the RESERVE_UNITS option.
- LICENSE COPY
Automatically disables a license as you reregister it in another License Database.
This command is similar in effect to LICENSE DISABLE except that it also registers the license in the other database.
- LICENSE ISSUE
Automatically disables a license so that you can then reregister it in another database.
This command is similar in effect to LICENSE DISABLE except that it also produces a PAK replica to be registered elsewhere.

Using LMF

2.6 Managing Licenses After Registration

2.6.1.3 Permanent Methods

For more permanent access restrictions, issue one of the following commands:

- LICENSE DELETE

Permanently deletes one or more licenses from a database, disallowing further license use. Use this command to eliminate terminated or unused licenses.

- LICENSE MOVE

Automatically deletes a license as you reregister the license in another License Database.

This command is similar in effect to LICENSE DELETE except that it also registers the license in the other database.

2.6.2 Controlling Node Access to Licenses in Clusters

In an OpenVMS Cluster environment, you can control which nodes have access to a product. Some product licenses require you to control access by creating include or exclude lists with the LICENSE MODIFY command.

In an OpenVMS Cluster environment, license units are available to all nodes by default. If you need to control which nodes in a cluster have access to a product license, you must use the LICENSE MODIFY command with the /INCLUDE or /EXCLUDE qualifiers. These qualifiers take an argument list of System Communications Services (SCS) node names. SCS node names are system parameters set with the System Generation utility (SYSGEN). For example, if your cluster includes nodes MUSIC, ART, DANCE, and THEATR, you can specify that nodes MUSIC and ART have access to the software products registered in the License Database, while nodes DANCE and THEATR do not have access. The following command allows two nodes access to Pascal:

```
$ LICENSE MODIFY/INCLUDE=(MUSIC,ART) PASCAL
```

You can perform the same task by using the /EXCLUDE qualifier. The following command specifies the same product access as the previous command:

```
$ LICENSE MODIFY/EXCLUDE=(DANCE,THEATR) PASCAL
```

If a license does not provide enough license units for full availability across the cluster, use LICENSE MODIFY and one of these qualifiers. Otherwise, product availability is unpredictable. Products are authorized for use on nodes in the order in which they load the licenses. Unless you use an include list to control which nodes can load a product, the authorization to use a product can move from processor to processor during a series of system shutdowns and startups. When you shut down a system, LMF automatically unloads all loaded licenses on that system. If another cluster member boots before the first system reboots, the second system, referring to the common License Database, can automatically load the same license. Although this can be helpful, it may not be your intention.

You can use the /INCLUDE and /EXCLUDE qualifiers with the LICENSE MODIFY command to determine who has access to the pool of units created by LMF when it loads combinable licenses. However, note the following:

- LMF combines units from combinable licenses (licenses with the same product name and type) into a single pool from which all authorized nodes may draw. By default, all nodes are authorized. To restrict the nodes that are authorized, assign an include or exclude list to each license for the product.

2.6 Managing Licenses After Registration

- Adding the NO_SHARE option to PAKs prevents license units from being shared. You can use this option to reserve all the units on a PAK for a particular node in an OpenVMS Cluster.
- LMF unionizes all the include and exclude lists associated with combinable licenses. The resulting master list is all-inclusive; when combining licenses, a less restrictive list always overrides a more restrictive list.
- Assigning an include list to a license does not reserve the license units for the nodes in the list.

For example, if you assign an include list to four out of five combinable licenses, the default—all nodes are authorized—is in effect for the fifth license, and it overrides the lists on the other licenses. All nodes then have access to the product despite the include lists. Units for the product are allocated on a first-come, first-served basis as the nodes in the cluster are booted, until the pool of units is depleted.

To ensure access exactly as you intend it, assign the same include or exclude list to each license. Purchasing more license units to fulfill requirements to run across the cluster is another option.

2.6.2.1 Effect of the NO_SHARE Option

Some licenses, such as OpenVMS operating system licenses, have the NO_SHARE option; they cannot be shared in a cluster environment. If you are using a common License Database, you must register a separate license for each cluster node and modify each to establish which node can load it.

To ensure that LMF can load a NO_SHARE license in a cluster environment, you have two choices, as follows:

- When you register with VMSLICENSE.COM, you are prompted for a node name. Enter the correct SCS node name at the prompt.
- If you use LICENSE REGISTER, you must follow up by entering a LICENSE MODIFY/INCLUDE=*node-name* command.

A cluster environment typically uses a single License Database, which should include one OpenVMS license for each cluster node. You can change the association of license and node as long as the number, type, and size of the licenses match the processors present. For example, the cluster environment with nodes ART, MUSIC, DANCE, and THEATR should have four licenses, each one designated for a specific node. If you remove node THEATR and replace it with a node named CRAFTS, you must modify the license intended for THEATR to specify node CRAFTS.

Because an OpenVMS Cluster License Database often contains multiple licenses with the identical product name and producer, you should use the /AUTHORIZATION qualifier with LICENSE commands to uniquely identify a specific license. For example:

```
$ LICENSE MODIFY/INCLUDE=THEATR BASIC /AUTHORIZATION=USA123456
.
.
.
$ LICENSE MODIFY/INCLUDE=CRAFTS AWESOME-OS
%LICENSE-E-AMBIG, information provided was ambiguous;
multiple licenses were found
$ LICENSE MODIFY/INCLUDE=CRAFTS AWESOME-OS /AUTHORIZATION=USA123456
```

Using LMF

2.6 Managing Licenses After Registration

2.6.2.2 Editing Include and Exclude Lists

Each time you enter a LICENSE MODIFY command with an /INCLUDE or /EXCLUDE qualifier, LMF creates a new list. To edit an existing list, use the /ADD or /REMOVE qualifier in your command line. The following example illustrates the required syntax without using the /ADD or /REMOVE qualifier:

```
$ LICENSE MODIFY/INCLUDE=(ART,MUSIC,DANCE,THEATR) BASIC -
_ $ /AUTHORIZATION=USA123456
.
.
.
$ LICENSE MODIFY/INCLUDE=(ART,MUSIC,DANCE,CRAFTS) BASIC -
_ $ /AUTHORIZATION=USA123456
$ LICENSE UNLOAD BASIC
%LICENSE-I-LOADED, DEC BASIC was successfully loaded with 400 units
```

You can also use the following commands:

```
$ LICENSE MODIFY/INCLUDE=(ART,MUSIC,DANCE,THEATR) BASIC -
_ $ /AUTHORIZATION=USA123456
.
.
.
$ LICENSE MODIFY/REMOVE/INCLUDE=(THEATR) BASIC /AUTHORIZATION=USA123456
$ LICENSE MODIFY/ADD/INCLUDE=(CRAFTS) BASIC /AUTHORIZATION=USA123456
```

If your license uses the MOD_UNITS option, you can also modify the size of a license in a cluster environment. To change the size of the license, enter a LICENSE MODIFY/UNITS=*number* command that specifies a number sufficient for your needs and allowed by your license agreement. For example, to change a license registered with 1000 license units to a 1500-unit license, enter:

```
$ LICENSE MODIFY/UNITS=1500 BASIC
$ LICENSE LOAD BASIC
```

2.6.3 Controlling User Access

To control which users have access to a product, use the LICENSE MODIFY command with the /RESERVE qualifier. This qualifier takes an argument list of user names called the reservation list. Although the definition of a user can differ from product to product, most products accept the user name that OpenVMS maintains for each account. This is the name you type at the Username prompt during login. See your product's Software Product Description (SPD) for details.

If your PAK specifies the RESERVE_UNITS option, you *must* assign one or more users to a reservation list. The number of user names allowed per list depends on the number of activity units available. Calculate this number as you would for any activity license. For example, if a software product requires 50 license units per activity and your PAK provides 100 license units, you have a 2-activity license. If the PAK also specifies the RESERVE_UNITS option, you have an unlimited activity, two-user license. For this license, you must create a reservation list with at least one, but no more than two, names.

Example: The following command assigns two users to a reservation list for the product called Terrapin:

```
$ LICENSE MODIFY /RESERVE=(R_HUNTER,J_BARLOW) TERRAPIN
```

Using LMF

2.6 Managing Licenses After Registration

Note that the LICENSE MODIFY command affects only data in the license database; it does not affect licenses already loaded. To affect a loaded license, reload it with a LICENSE LOAD command. For example:

```
$ LICENSE MODIFY /RESERVE=(R_HUNTER,J_BARLOW) TERRAPIN
$ LICENSE LOAD TERRAPIN
%LICENSE-I-UNLOADED
LICENSE-I-LOADED, DEC TERRAPIN was successfully loaded with 200 units
```

To add more user names to the reservation list, use the /ADD qualifier and the /RESERVE qualifier, as follows:

```
$ LICENSE MODIFY /ADD /RESERVE=(P_LESH,M_HART) TERRAPIN
```

This adds new users P_LESH and M_HART to any list already established for the specified product. You can remove a user name with the /REMOVE qualifier.

Note

LMF does not restrict you from creating incorrect reservation lists. If a user on a reservation list is being denied access to a product, check the reservation list (or reservation lists with multiple licenses for the same product) for the following:

- Too many names. If you repeat a user name, LMF can reject a valid user name entry after reaching the allowed number of users for the license. LMF provides a warning when it loads a license with a list that is too long.
- Incorrect spelling of user names. LMF simply compares and counts user names. If you misspell a name in the reservation list, LMF denies access to the user trying to access a product. LMF still counts each misspelled name as a potential user.

You can have many Personal Use Licenses for the same product. For license loading, LMF combines all of the license units and determines the number of users according to the total number of units. Therefore, the total number of names on combined reservation lists for this product must not exceed the number LMF authorizes from the unit count, because LMF authorizes only the correct number from the lists and rejects extra names.

Although you can find extra or repeated names using one or more LICENSE LIST/FULL commands, you cannot easily predict which users LMF will reject. Do not assume that LMF denies access to the third name listed on the reservation list of a two-user license. The total number of names and total number of license units is the important calculation.

To load corrections to a reservation list you must enter the LICENSE LOAD command for the licenses. The following example includes the warning message for too many names:

```
$ LICENSE MODIFY/RESERVE=(R_HUNTER,J_BARLOW) TERRAPIN
$ LICENSE LOAD TERRAPIN
LICENSE-I-LOADED, DEC TERRAPIN was successfully loaded with 200 units
$ LICENSE MODIFY/ADD/RESERVE=(P_LESH,M_HART) TERRAPIN
$ LICENSE LOAD TERRAPIN
%LICENSE-I-UNLOADED
LICENSE-W-LONGLIST, reserve list for DEC TERRAPIN exceeds maximum of 2, 2 names removed
LICENSE-I-LOADED, DEC TERRAPIN was successfully loaded with 200 units
```

Using LMF

2.6 Managing Licenses After Registration

Because LMF combines the license units, you can assign one long reservation list to a single license; LMF simply adds the license units from the combinable licenses and counts the names in all reservation lists for those licenses. If you have three combinable licenses that authorize use to six users, you can modify one of them to have a 6-name reservation list. Note that this differs from the behavior of include and exclude lists with node names in an OpenVMS Cluster.

2.6.4 Controlling Loading Order

If you have many variations of licenses for a product (for example, with different versions, tokens, or hardware identifiers), and you think that you are not getting maximum use of the product, you can check the order of loading of the product licenses. By default, LMF assigns a selection weight to each license and loads each license in descending order of selection weight. The grant order is the order in which LMF checks licenses before loading one. To determine grant order, enter the DCL command `SHOW LICENSE/FULL`. You can then enter the `LICENSE LIST` command with the `/SELECTION_WEIGHT` qualifier to display the selection weight. Modify selection weights of licenses as needed with the `/SELECTION_WEIGHT` qualifier to the `LICENSE MODIFY` command.

To change the selection weight, use `LICENSE MODIFY/SELECTION_WEIGHT`, and then load the changed licenses with `LICENSE LOAD`.

2.7 Logical Name LMF\$DISPLAY_OPCOM_MESSAGE

A previous version of this manual incorrectly stated that you must define the logical name `LMF$DISPLAY_OPCOM_MESSAGE` in order for messages to appear.

If you have already defined this logical name, you should delete the definition.

Define the `LMF$DISPLAY_OPCOM_MESSAGE` logical name only if you are explicitly directed by Digital to do so (for debugging purposes). When defined, this logical name causes many “noise” messages to be displayed on the operator’s console. Some of these messages can be confusing and misleading to the point of suggesting that a product is not licensed when in fact it is.

To see if this logical name has been defined on your system, enter the following command:

```
$ SHOW LOGICAL LMF$DISPLAY_OPCOM_MESSAGE
```

To delete the logical name, enter the following command:

```
$ DEASSIGN LMF$DISPLAY_OPCOM_MESSAGE/EXEC/SYSTEM
```

Command Reference

This appendix describes the syntax of the following License Management utility (LICENSE) commands:

- LICENSE COPY
- LICENSE CREATE
- LICENSE DELETE
- LICENSE DISABLE
- LICENSE ENABLE
- LICENSE ISSUE
- LICENSE LIST
- LICENSE LOAD
- LICENSE MODIFY
- LICENSE MOVE
- LICENSE REGISTER
- LICENSE START
- LICENSE UNLOAD

LICENSE

LICENSE COPY

Copies licenses from one License Database to another. When you use LICENSE COPY, LMF disables the source license and registers a copy in the destination License Database as if it were a new license. If the terms and conditions of your license contract allow it, you can re-enable the source database license by using LICENSE ENABLE.

LICENSE COPY cannot be used to create a copy of a license in the same database as the source of the copy.

Format

```
LICENSE COPY product-name[...] output-database
```

Parameters

product-name[...]

Name or names of products with a license to be copied to the output License Database.

output-database

File specification of the License Database to which the license or licenses should be copied. This database must have been previously created using LICENSE CREATE.

If you enter a partial file specification (for example, specifying only a directory), LMF\$LICENSE is the default file name, and .LDB is the default file type. If you do not specify a device or directory, the current default device and directory are used.

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be copied. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to copy. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database from which the license should be copied. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE COPY displays the name of each license that it copies.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

Description

To copy a license from one database to another, use LICENSE COPY. The following conditions apply to a LICENSE COPY transaction:

- The status of the source database license changes to disabled.
- Network copies are supported within the limits of remote FAL access. If you use access control strings, such as “USERNAME password” within the file specification, the actual password string is not stored.
- LICENSE COPY does not transfer any user-supplied data such as reservation lists, modified termination dates, modified units, include or exclude node lists, or comments.

Examples

1. `$ LICENSE COPY FORTRAN BACKUP_DATA:BACKUP.LDB`

This command copies the Fortran license in the default License Database to the BACKUP_DATA:BACKUP.LDB License Database. This command fails if there is more than one Fortran license in the default database.

2. `$ LICENSE COPY FORTRAN/DATABASE=BACKUP_DATA:BACKUP.LDB -
_ $ BACKUP_DATA2:BACKUP2.LDB`

This command copies the Fortran license in the source License Database to the BACKUP_DATA2:BACKUP2.LDB License Database. This command fails if there is more than one Fortran license in the source database.

3. `$ LICENSE COPY FORTRAN/ALL BACKUP_DATA:BACKUP.LDB`

This command copies all Fortran licenses in the default License Database to the BACKUP_DATA:BACKUP.LDB License Database.

LICENSE LICENSE COPY

4. \$ LICENSE COPY FOR* BACKUP_DATA:BACKUP.LDB

This command copies all licenses whose product names begin with the string "FOR" from the default License Database to the BACKUP_DATA:BACKUP.LDB License Database. In this case, using the wildcard character (*) implies the use of /ALL.

5. \$ LICENSE COPY * BACKUP_DATA:BACKUP.LDB

This command copies all licenses from the default License Database to the BACKUP_DATA:BACKUP.LDB License Database. In this case, using the wildcard character (*) implies the use of /ALL.

6. \$ LICENSE COPY */PRODUCER=DEC BACKUP_DATA:BACKUP.LDB

This command copies all licenses with the producer name DEC from the default License Database to the BACKUP_DATA:BACKUP.LDB License Database. In this case, using the wildcard character (*) implies the use of /ALL.

7. \$ LICENSE COPY D%% BACKUP_DATA:BACKUP.LDB

This command copies all licenses beginning with a "D" followed by any two characters from the default License Database to the BACKUP_DATA:BACKUP.LDB License Database. In this case, using the wildcard character (%) implies the use of /ALL.

LICENSE CREATE

Creates a License Database with no license records. Because LMF provides a default License Database in SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB when OpenVMS is installed, you do not typically need to use this command.

To use LMF, you must have a License Database file *and* the License Unit Requirement Table (LURT) file (SYS\$COMMON:[SYSEXE]LMF\$LURT.DAT), which comes installed with OpenVMS.

Format

LICENSE CREATE

Parameters

None.

Qualifier

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system.

Example

```
$ LICENSE CREATE/DATABASE=SYS$MANAGER:LMF$LICENSE.LDB
```

This command creates the License Database named LMF\$LICENSE.LDB in the directory SYS\$MANAGER.

LICENSE LICENSE DELETE

LICENSE DELETE

Deletes one or more licenses and all history information for those licenses from the License Database. Deleted licenses are no longer available to the system for any use.

Format

LICENSE DELETE product-name[,...]

Parameter

product-name[,...]

Name or names of products with a license to be removed from the License Database. You can delete only licenses that have been registered.

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be deleted. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to delete. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database from which the license or licenses should be deleted. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE DELETE displays the name of each license that it deletes.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/STATUS=[(keyword)[,...]]

Positional qualifier.

Selects licenses to be deleted according to the *product-name* parameter specified and one or more license status keywords from the following list:

- ALL (default)—Deletes all specified licenses in the database.
- ACTIVE—Deletes all specified enabled licenses in the database. *ACTIVE* status means that the registered license is enabled for loading. For backward compatibility, the LICENSE LIST command identifies enabled licenses as having a status of active.
- DISABLED—Deletes all specified disabled licenses in the database.
- EXTINCT—Purges specified license information by deleting all extinct license records in the database. Extinct records are history records retained after a license is modified.
- CANCELED—Deletes all specified canceled licenses in the database. Note that current versions of LMF do not set license status to canceled. Old licenses may have this status.

If you enter more than one keyword, separate them with commas, and enclose the list in parentheses. You can abbreviate each option to the minimum number of characters needed to uniquely identify it.

Description

Use LICENSE DELETE to delete licenses from the License Database. To tailor your command, use options to the /STATUS qualifier and wildcard characters in product name strings.

File space is not released following LICENSE DELETE commands. For information on retrieving Record Management Services (RMS) file space, see the *OpenVMS Record Management Utilities Reference Manual*.

Examples

1. \$ LICENSE DELETE FORTRAN

This command deletes the Fortran license from the default License Database.

2. \$ LICENSE DELETE FORTRAN, COBOL, PASCAL

This command deletes the Fortran, COBOL and Pascal licenses from the default License Database.

3. \$ LICENSE DELETE FORTRAN/DATABASE=MY\$DISK:MYDATA.LDB

This command deletes the Fortran license from the MY\$DISK:MYDATA.LDB License Database.

LICENSE LICENSE DELETE

4. \$ LICENSE DELETE FORTRAN/ISSUER=XYLASOFT

This command deletes all licenses for the product named Fortran issued by XYLASOFT from the default License Database. If there are licenses for products named Fortran issued by companies other than XYLASOFT, they are not deleted.

5. \$ LICENSE DELETE */STATUS=(EXTINCT)

This command deletes all license records with a status of EXTINCT from the database. This is effectively a purge of all historical information.

LICENSE DISABLE

Disables a license currently registered in the License Database.

Format

LICENSE DISABLE product-name[,...]

Parameter

product-name[,...]

Name or names of products with a license that you want to disable. You can disable only licenses that currently exist in the License Database. Enter the product name exactly as it appears on your Product Authorization Key (PAK).

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be disabled. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to disable. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE DISABLE displays the name of each license that it disables.

LICENSE LICENSE DISABLE

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

Description

LICENSE DISABLE does not immediately affect loaded licenses. To affect a loaded license, you must first enter a LICENSE UNLOAD command, which unloads the license, but allows current processes to finish using the product. Note that to immediately disable all loaded licenses, you must shut down the system.

You cannot use LICENSE LOAD to activate a disabled license; you must first use LICENSE ENABLE.

LMF does not display error messages when you or the system attempts to unload a disabled license.

Example

```
$ LICENSE DISABLE ABCD /PRODUCER=DEC
```

This command disables the license for ABCD software produced by Digital. Because no database is specified, LMF uses the default database.

LICENSE ENABLE

Enables an existing license in the License Database so that you can load it with LICENSE LOAD. This command cancels the effect of LICENSE DISABLE, LICENSE COPY, and LICENSE ISSUE, which leave the license disabled. Newly registered licenses are enabled by default.

Format

LICENSE ENABLE product-name[,...]

Parameter

product-name[,...]

Name or names of products with a license to enable. You can enable only licenses that currently exist in the License Database. Enter the product name exactly as it appears on your PAK.

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be enabled. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to enable. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE ENABLE displays the name of each license that it enables.

LICENSE LICENSE ENABLE

/PRODUCER=string
Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

Description

Use LICENSE ENABLE to reestablish disabled licenses as available for loading with a LICENSE LOAD command.

Enabled licenses can combine with other licenses when loaded for use. Do not enable a license that has expired, and be sure that all include, exclude, and reservation lists are up to date.

Use LICENSE LIST to inspect each license before you enable it. Use LICENSE MODIFY to change include, exclude, and reservation lists.

Because errors do not occur until enabled licenses are loaded, consider entering LICENSE LOAD immediately to load each newly enabled license on each appropriate node in an OpenVMS Cluster. If another combinable license for the same product is already loaded, first unload it with LICENSE UNLOAD. Use the DCL command SHOW LICENSE to see which licenses are currently active on your system. After you unload the other license, enter LICENSE LOAD to load the combination of the newly enabled license and the previously active license.

Example

```
$ LICENSE ENABLE DECSET /PRODUCER=DEC
```

This command enables the license for DECset software. Because no database is specified, LMF uses the default database. Next, load the license with LICENSE LOAD.

LICENSE ISSUE

Produces a replica of a Product Authorization Key (PAK) that is sent to a file or displayed on your terminal (the default). If the terms and conditions of your license contract allow it, you can then enter this PAK replica in the License Database of another processor. When you enter **LICENSE ISSUE**, LMF disables the license in the current License Database and marks the license **DISABLED**. To enable a license that has been marked **ISSUED**, enter **LICENSE ENABLE**.

For License Databases connected to a network, consider using **LICENSE MOVE**.

Format

LICENSE ISSUE product-name[,...]

Parameter

product-name[,...]

Name or names of products with a license to be issued. You can issue only licenses that currently exist in the License Database. Enter the product name exactly as it appears on your PAK.

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be issued. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to issue. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name **LMF\$LICENSE**, which points to **SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB** on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, **DEC**) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

LICENSE LICENSE ISSUE

/LOG

/NOLOG (default)

Controls whether LICENSE ISSUE displays the name of each license that it issues.

/OUTPUT[=filespec]

Specifies the name of the file to which your PAK replica is written. If you do not specify the /OUTPUT qualifier, or if you do not supply a file specification with this qualifier, the output is sent to SYS\$OUTPUT.

If you specify a file name that already exists, this command creates a new version of the file. If you specify a complete file name and version that already exists, this command appends the PAK replica to the existing file.

/PROCEDURE

/NOPROCEDURE (default)

Specifies that the PAK replica is to be written in the form of a DCL command procedure. Use /PROCEDURE with the /OUTPUT qualifier to create a command procedure in a file. Then you can invoke the procedure to register the PAK replica in the License Database of another processor.

If you do not specify the /OUTPUT qualifier with /PROCEDURE, or if you do not supply a file specification with the /OUTPUT qualifier, the procedure is sent to SYS\$OUTPUT.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

Description

If your license contract allows it, use LICENSE ISSUE to move a license from a License Database on one processor (or OpenVMS Cluster environment) to a License Database on another processor. To move a license, enter LICENSE ISSUE, including enough PAK information to clearly identify the license. LICENSE ISSUE automatically disables the current license but does not immediately unload it; LMF does not terminate any active processes. To unload the license, enter a LICENSE UNLOAD command.

After you issue the PAK replica, read the information, and register it on the new processor as you would any PAK, or, if you used the /PROCEDURE qualifier with the /OUTPUT qualifier, invoke the new DCL command procedure to register the license.

Note that the PAK replica includes only PAK information registered with a LICENSE REGISTER command. The replica does not include any changes made with other LICENSE commands.

Examples

1. \$ LICENSE ISSUE /OUTPUT=SYS\$MANAGER:FORTRAN.PAK -
_ \$ /PRODUCER=DEC FORTRAN

This command issues a PAK replica, which you can use to register the Fortran license on a new processor (or OpenVMS Cluster environment), and puts it into the file named FORTRAN.PAK. The next step is to print the file, read the information, and, using a LICENSE REGISTER command or VMSLICENSE.COM, enter the correct information in the License Database of the new processor. The Fortran license in the current License Database is marked ISSUED and is disabled.

2. \$ LICENSE ISSUE /PRODUCER=DEC VOLSHAD

This command displays, at the current terminal, a PAK replica with the information from the VOLSHAD (VAX Volume Shadowing) license. This display is reproduced below. The license registered in the current License Database is marked ISSUED and is disabled. You can register the data from this replica of a PAK in the License Database of another processor using either VMSLICENSE.COM or LICENSE REGISTER.

```
Software Product Authorization Key Replica
Issued by CASPER
Issued on 24-FEB-1996 14:23
```

```
-----
Issuer:          DEC
Authorization:   ALS-WM-93166-5573
Product Name:   VOLSHAD
Producer:       DEC
Units:          460
Version:        5.4
PAK Termination Date: 31-DEC-1996
Availability:    E
Options:        MOD UNITS
Checksum:       1-ADEB-DOJ-NENC-KDBM
```

3. \$ LICENSE ISSUE/PROCEDURE /OUTPUT=FORTRAN-USA10.COM -
_ \$FORTRAN /AUTHORIZATION=USA-10

This command generates a DCL command procedure such as the following to be used for registering the specified license in a License Database:

```
$! Software Product Authorization Key Replica
$! Issued by CASPER
$! Issued on 23-Oct-1991 14:23
$ LICENSE REGISTER FORTRAN -
  /ISSUER=DEC -
  /PRODUCER=DEC -
  /AUTHORIZATION=USA-10 -
  /UNITS=400 -
  /VERSION=5.4 -
  /AVAILABILITY=F -
  /CHECKSUM=1-HIDN-INDA-COMP-DAH
```

LICENSE LICENSE LIST

LICENSE LIST

Displays information from the License Database on disk about the specified license or licenses. Use one or more qualifiers to control the form, content, and location of information displayed.

The SHOW LICENSE command, described in the *OpenVMS DCL Dictionary*, displays information from the License Database in memory.

Format

```
LICENSE LIST [product-name[,...]]
```

Parameter

product-name[,...]

Name or names of products with a license that you want to list. You can list only licenses that currently exist on disk in the License Database. You can specify one product name or use wildcard characters to display licenses. The *product-name* parameter is optional; the default is to display all of the licenses.

Qualifiers

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to list. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license.

This qualifier affects only the product name that immediately precedes it in the command string.

/BEFORE

Used with /TERMINATION_DATE and /RELEASE_DATE, selects only those licenses whose times are before the time specified with the other qualifiers.

The /BEFORE qualifier cannot be used with the /SINCE qualifier.

/BRIEF (default)

Specifies a listing from the License Database that includes only the license product and producer names.

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXEC]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/FULL

Specifies a listing from the License Database that includes a full display of the specified license or licenses.

/HISTORY

Specifies a listing from the License Database that includes the history records in the License Database for the specified license or licenses.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/OUTPUT[=filespec]

Specifies the name of the file to which your list is written. If you do not specify the /OUTPUT qualifier, or if you do not supply a file specification with this qualifier, the output is sent to SYS\$OUTPUT.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license.

This qualifier affects only the product name that immediately precedes it in the command string.

/RELEASE_DATE=date

Used with /BEFORE or /SINCE, specifies a listing from the License Database that includes only licenses with a release date on or after the date specified. The date must be presented in the standard OpenVMS format: *dd-mmm-yyyy*. The default value is /SINCE /RELEASE_DATE=TODAY.

/SELECTION_WEIGHT

Produces a full display that includes the current selection weights assigned to individual PAKs.

/SINCE

Used with /TERMINATION_DATE and /RELEASE_DATE, selects only those licenses whose times are after the time specified with the other qualifiers.

/SINCE cannot be used with /BEFORE.

/TERMINATION_DATE[=date]

Used with /BEFORE or /SINCE, specifies a listing from the License Database that includes only licenses with a termination date on or after the date specified. The date must be presented in the standard OpenVMS format: *dd-mmm-yyyy*. The default value is /SINCE /TERMINATION_DATE=TODAY.

/VERSION=nn.nn

Positional qualifier.

Specifies the version number of the product for which you have a license. Versions use the format *integer.integer*. You can specify wildcard syntax as **.** but not *** alone. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

LICENSE LICENSE LIST

Description

LICENSE LIST displays license records as they appear on disk in the License Database. LICENSE LIST /BRIEF does not produce a display with history records. You can control the displays as follows:

- After you enter LICENSE LIST with the /BRIEF qualifier, you can scroll through the display with the arrow keys on your keyboard.
- After you enter LICENSE LIST with the /FULL or /HISTORY qualifier, which displays the first LICENSE record, you can see the other records one at a time by pressing Return. You can also scroll through the license records using the Previous Screen key (or PF3) and the Next Screen key (or PF4).

For any LICENSE LIST display, use the arrow keys to scroll vertically or horizontally one line at a time. Press Ctrl/Z to exit from the display.

Note that a LICENSE LIST command may display the status of a registered license as *Active*. This means the registered license is enabled for loading; it has not been disabled. It does not necessarily mean the license was loaded with a LICENSE LOAD command. The LICENSE LIST command displays only information on disk in the License Database; enter SHOW LICENSE to determine all active licenses on the current system.

You can also list licenses using the VMSLICENSE.COM command procedure.

Examples

1. \$ LICENSE LIST/FULL

This command displays a list of the names of product licenses in the License Database. Note that the LMF Version shown refers to the software that created the database.

Use CTRL/Z to exit, PF3-PF4 for Previous-Next Screen, Arrow Keys to scroll.

```
License Management Facility V1.2
License Database File:      WORK2:[BACKUP]LMF$LICENSE.LDB;1
Created on:                 20-JUL-1995
Created by user:            USER_1
Created by LMF Version:     V1.1
-----
Issuer:                     DIGITAL-IM&T
Authorization:
Product Name:               C
Producer:                   DEC
Units:                      500
Version:                    0.0
Release Date:               4-MAY-1996
PAK Termination Date:      (none)
Availability:                0
Activity:                    000000100
Options:                     MOD_UNITS
Product Token:
Hardware ID:
Revision Level:             1
Status:                     Active
Command:                    REGISTER
```

LICENSE LICENSE LIST

2. \$ LICENSE LIST /HISTORY FORTRAN

This example displays a listing with full information of a current Fortran license issued by Digital.

The first screen, shown here, displays the most recent license record for the Fortran license. To see the history records one screen at a time, press Return. The revision level of the displayed record is 2, and the status is Active. The next screen would display the previous license record with a revision level of 1 and a status of Extinct.

Use CTRL/Z to exit, PF3-PF4 for Previous-Next Screen, Arrow Keys to scroll.

License Management Facility V1.2

License Database File: ART::SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB

Created on: 17-AUG-1995
Created by user: USER_2
Created by LMF Version: V1.2

Issuer: DEC
Authorization: USA-2468
Product Name: FORTRAN
Producer: DEC
Units: 0
Modified Units: 2000
Version: 5.4
Release Date: (none)
PAK Termination Date: 31-DEC-1997
Modified Termination Date: 30-NOV-1998
Availability: F (Layered Products)
Activity: 0
Options: MOD_UNITS
Hardware ID:

Revision Level: 2
Status: Active
Command: MODIFY
Modified by user: DEGAS
Modified on: 19-AUG-1996 14:32:23.41
Include: ART

LICENSE LICENSE LOAD

LICENSE LOAD

Loads licenses, making them available for product authorization on the current node. The product licenses must be registered and current in the License Database. That is, they must not have been disabled or issued.

If the license is already loaded, LMF returns an informational message, unloads the license, and then loads the license.

To use this command, you need CMKRNL, SYSNAM, and SYSPRV privileges.

Format

LICENSE LOAD [product-name][,...]

Parameter

[product-name][,...]

Name or names of products with a license to be loaded. You can load only licenses that are currently registered and enabled in the License Database. Enter the product name exactly as it appears on your Product Authorization Key (PAK). If you do not specify a product name, LICENSE LOAD loads all of the products that are registered and enabled.

You cannot use wildcard characters for *product-name*.

Qualifiers

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to register. You must enter the authorization string exactly as it appears on your PAK. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/ISSUER=string

Positional qualifier.

Name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license.

This qualifier affects only the product name that immediately precedes it in the command string.

/LOG (default)

/NOLOG

Controls whether or not LICENSE LOAD displays a message to acknowledge the loading of each license.

/PRODUCER=string

Positional qualifier.

Name of the company that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/UNLOAD (default)

/NOUNLOAD

When requested to load a license that is currently loaded, LMF first automatically unloads it and then loads the latest license.

You can specify **/NOUNLOAD** to verify whether or not there is already a license loaded; LMF issues the warning **LICENSE-W-ALREADYLOADED** and does not load the license. To then load the license, follow these steps:

1. Manually unload the current license with the **LICENSE UNLOAD** command.
2. Reissue the **LICENSE LOAD** command.

Description

The **LICENSE LOAD** command loads licenses registered in the License Database. To use a licensed product, ensure that the system loads the registered license. When you register a license with **VMSLICENSE.COM**, you can confirm an option to load the license, whereas if you register a license with **LICENSE REGISTER**, you must also load it with **LICENSE LOAD**.

Use **LICENSE LOAD** only after you register a new license; LMF automatically loads all registered licenses at each subsequent system startup. You can enter **LICENSE LOAD** at other times to load modifications made with other **LICENSE** commands.

You can enter one **LICENSE LOAD** command without *product-name* to load all the available registered licenses.

Note

Registered licenses are enabled for loading by default. You can, however, disable a registered license to prevent loading.

A **LICENSE START** command entered interactively or when the system reboots also loads all licenses that are registered and enabled.

If you register multiple licenses for a single product, **LICENSE LOAD** loads all of the matching licenses. You do not typically load individual licenses, and you cannot unload individual licenses for a product. The Availability, Activity, Personal Use, and User license units of the multiple licenses work in concert to provide more product availability.

LICENSE LICENSE LOAD

In an OpenVMS Cluster environment, each system loads licenses when it reboots. If you need to load a license for all assigned nodes of a running cluster, you can do one of the following:

- Log in to each OpenVMS Cluster node, and enter LICENSE LOAD.
- Invoke the OpenVMS SYSMAN utility to execute the LICENSE LOAD command on the desired OpenVMS Cluster nodes. See the *OpenVMS System Manager's Manual* for details on defining your management environment and executing commands on a list of nodes.

A LICENSE LOAD command can fail, sending a message to the operator communication manager (OPCOM) for any of the following reasons:

- Insufficient license units are registered for the current node.
- The current date is later than the license termination date.
- A license checksum does not match the rest of the license data. Check for data corruption in the License Database.

If you attempt to load a disabled license or a license modified to exclude the current node in an OpenVMS Cluster environment, OPCOM does not display an error message.

If licenses for more than one product are being loaded, LICENSE LOAD continues with the next license following a failure.

Examples

1.

```
$ LICENSE MODIFY/INCLUDE=MUSIC FORTRAN
$ LICENSE LOAD FORTRAN
```

The commands in this example illustrate a situation where you enter a LICENSE LOAD command interactively. LICENSE LOAD loads the product Fortran on the node MUSIC. Data in the License Database determines whether the license is successfully loaded on the specified node.

2.

```
$ LICENSE LOAD BASIC
%LICENSE-W-NOLOAD, license was not loaded for BASIC
-LICENSE-F-EXCEEDED, attempted usage exceeds active license limits
```

This command attempts to load the product BASIC, but LICENSE LOAD fails because too few license units are registered to authorize use on the current processor.

LICENSE MODIFY

Modifies a license for system management and license-sharing purposes. Immediately changes data in the License Database, but your modifications do not affect the running system until you load the modified license.

Format

LICENSE MODIFY *qualifier*[,...] *product-name*[,...]

Parameter

product-name[,...]

Name or names of products with a license to be modified. You can modify only licenses that currently exist in the License Database.

Qualifiers

/ADD

Used with the **/INCLUDE** or **/EXCLUDE** qualifier, specifies that the node names provided are to be added to the previously established include or exclude lists.

Used with the **/RESERVE** qualifier, specifies that the user names provided are to be added to the previously established reservation lists.

When you use **/ADD**, you do not need to retype the entire list to add a new node name or user name.

/ALL

Positional qualifier.

Modifies all the licenses with the given product name. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to modify. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/COMBINE

/NOCOMBINE

Modifies a PAK by adding or removing the **COMBINE** option. If the PAKs are combinable, LMF combines them during license loading.

/COMMENT=string

Specifies a string of text. Use this comment field of up to 63 characters to associate information for this transaction with the license. History records for the license retain this license information. If you specify more than one word, enclose the text in quotation marks (" "). This qualifier is optional.

The text in the comment field is replaced only when you enter new comments with another **LICENSE MODIFY** command. At this point the old comment text is available as a history record.

LICENSE LICENSE MODIFY

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

/EXCLUDE=(node-name[,node-name,...])

Specifies that the named node or nodes in an OpenVMS Cluster environment cannot access the licensed product. The excluded nodes cannot load (with a LICENSE LOAD or LICENSE START command) the license registered in the License Database. Each **node-name** argument must be a System Communications Services (SCS) node name or a system parameter set with the System Generation utility (SYSGEN). The node name might not be the same as the DECnet node name. If you specify more than one node name, separate them with commas, and enclose the list in parentheses. This qualifier is optional.

To modify previously defined lists without having to retype all of the node names, use the /ADD or /REMOVE qualifiers with /EXCLUDE.

You can control license access to nodes with /EXCLUDE and control user access with /RESERVE, but you cannot use these qualifiers on the same command line. To use both types of control with the same license, you must enter separate LICENSE MODIFY commands.

/INCLUDE=(node-name[,node-name,...])

Specifies that the named node or nodes in an OpenVMS Cluster environment can access the licensed product. Only the included nodes can load (with a LICENSE LOAD or LICENSE START command) the license registered in the License Database. Each **node-name** argument must be an SCS node name, or a system parameter set with SYSGEN. The node name might not be the same as the DECnet node name.

Licenses for the OpenVMS operating system usually specify the NO_SHARE option on their PAKs. In a cluster environment you must restrict each of these OpenVMS licenses to a single node. If you did not do this when registering with VMSLICENSE.COM, enter LICENSE MODIFY/INCLUDE=*node-name*, specifying one SCS node name for each OpenVMS license.

To specify more than one SCS node name for a license that does not specify NO_SHARE, separate the names with commas, and enclose the list in parentheses. This qualifier is optional.

To modify previously defined lists without having to retype all of the node names, use the /ADD or /REMOVE qualifiers with /INCLUDE.

You can control license access to nodes with /INCLUDE and control user access with /RESERVE, but you cannot use these qualifiers on the same command line. To use both types of control with the same license, you must enter separate LICENSE MODIFY commands.

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this qualifier only if it is required to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE MODIFY displays the name of each license that it modifies.

/NO_SHARE

/NONO_SHARE

Specifies whether to add or subsequently remove /NO_SHARE from a PAK. Adding /NO_SHARE prevents the sharing of the PAK units with other cluster nodes.

PAKs with /NO_SHARE require you to provide the SCS node name of the cluster node that will be using this particular license. See the /INCLUDE qualifier for more information.

Note that if /NO_SHARE is present on your PAK when you register it, you cannot remove the option using /NONO_SHARE. Only if you add /NO_SHARE with the MODIFY command, can you subsequently remove it.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/REMOVE

Used with the /INCLUDE or /EXCLUDE qualifier, specifies that the node names provided are to be removed from the previously established include or exclude lists.

Used with the /RESERVE qualifier, specifies that the user names provided are to be removed from the previously established reservation lists.

When you use /REMOVE, you do not need to retype the entire list to remove a node name or user name.

/RESERVE=(user-name[,user-name,...])

Specifies that the license or licenses are to be reserved for use by the users listed in the *user-name* parameter. Users not listed are denied access to the product. The value applied to *user-name* differs from product to product. See your Software Product Description (SPD) for details.

Most products define **user-name** to be the user name OpenVMS maintains for each account. This is the name you type at the Username prompt during login.

If your PAK specifies the RESERVE_UNITS option, you *must* assign one or more users to a reservation list. The number of user names allowed per list depends on the number of activity units available and a constant value or the License Unit Requirement Tables (LURTs). Calculate this number as you would for any Activity License. For example, a 200-unit license with a constant value of 100 is a two-user license.

You can also create and modify a reservation list for Availability and regular Activity Licenses that do not specify the RESERVE_UNITS option. Because these licenses do not limit the number of names on the list, you can assign as many names as you like to the reservation list. All users not on the list are denied access.

LICENSE LICENSE MODIFY

Although you can control license access to nodes with `/INCLUDE` and `/EXCLUDE` qualifiers and control user access with the `/RESERVE` qualifier, you cannot use these qualifiers on the same command line. If you want to use both types of control with the same license, you must enter separate `LICENSE MODIFY` commands.

Use the `/ADD` and `/REMOVE` qualifiers for further control in modifying previously established reservation lists.

`/SELECTION_WEIGHT=number`

Modifies the selection weight. Selection-weight values determine the order in which LMF checks multiple licenses when a product makes a license grant request. LMF checks higher-weighted licenses before lower-weighted ones. Specify arbitrary numbers between 1 and 1000.

Note

You cannot modify selection weights for Availability Licenses.

To restore the selection weight of a PAK to the default value, enter the `LICENSE MODIFY` command with `/SELECTION_WEIGHT=0`. For example, you can use either of the following commands:

```
$ LICENSE MODIFY FORTRAN /SELECTION_WEIGHT=0
$ LICENSE MODIFY FORTRAN /NOSELECTION_WEIGHT
```

`/TERMINATION_DATE=date`

Date at which the product license is to be terminated. If your PAK supplied a license termination date, LMF uses the earliest date to determine the termination date. The date must be presented in the standard OpenVMS format: *dd-mmm-yyyy*. If you want to restrict a product from further use today, enter yesterday's date; LMF terminates the license at the end of the day specified.

`/UNITS=n`

Number of license units you want on a license that includes the `MOD_UNITS` option. If your PAK allows you to modify the license units, use this qualifier to change the value in the License Database.

Description

Use the `LICENSE MODIFY` command to modify a license. To control which nodes in a cluster environment have access to what software, use `LICENSE MODIFY` with the `/INCLUDE` or `/EXCLUDE` qualifier. For example, you can load licenses for products used less often or requiring limited access on one node.

If you do not specify which nodes can load a license (with a `LICENSE LOAD` or `LICENSE START` command), LMF loads a license on a first-come, first-served basis. When your license has insufficient license units for full cluster environment use, control product access with an include list.

Because most OpenVMS PAKs use the `/NO_SHARE` option, in a cluster environment you must restrict these operating system licenses to one node. Enter `LICENSE MODIFY/INCLUDE=node-name`, specifying only one SCS node name for each OpenVMS license.

LICENSE LICENSE MODIFY

To control which users have access to a product, use LICENSE MODIFY with the /RESERVE qualifier. You can create and modify a reservation list for any kind of license. Only users on the reservation lists are allowed access to the product.

If your PAK specifies the RESERVE_UNITS option, you *must* assign one or more users to a reservation list. The number of user names allowed per list depends on the number of activity units available and a constant value or the License Unit Requirement Tables (LURTs). Calculate this number as you would for any Activity License. For example, a 200-unit license with a constant value of 100 is a two-user license.

Use the /ADD and /REMOVE qualifiers in conjunction with the /INCLUDE, /EXCLUDE, and /RESERVE qualifiers when you modify existing include, exclude, and reservation lists.

To add comments about a license in the License Database, use LICENSE MODIFY with the /COMMENT qualifier.

If your PAK includes the MOD_UNITS option, you can use the /MOD_UNITS qualifier to specify the number of license units you want for your registered license.

Use the other LICENSE MODIFY command qualifiers only as needed to identify the correct license.

You can also modify a license record using the VMSLICENSE.COM command procedure.

List Size Restrictions

Two restrictions apply to the size of lists (reservation lists, include lists, or exclude lists). These restrictions apply to PAKs of all license types.

- On any single PAK, the sum of characters contained in all lists must not exceed 5000 characters.

Because the length of names vary and some overhead is used for each name, this 5000-character limit cannot be expressed as an exact number of permissible names. However, Digital guarantees that at least 400 names, in total, can be specified in the various types of lists. For example, each of the following represents the minimally guaranteed number of names:

- Reservation list with up to 400 user names
- Reservation list with up to 200 user names plus an include list with up to 200 node names (totaling up to 400)
- Reservation list with up to 200 user names plus an exclude list with up to 200 node names (totaling up to 400)
- Include list with up to 400 node names
- Exclude list with up to 400 node names

Note

If you enter more names than are permitted, LICENSE LIST might not be able to display all names entered. In this case, you receive the error message LICENSE-F-CORRUP. However, the License Database is not actually corrupt, and the PAKs can still be loaded into memory (though the names are not displayed).

LICENSE LICENSE MODIFY

- The LICENSE LOAD and LICENSE START commands can load into memory a reservation list with no more than 30,000 characters. (Include and exclude lists, which are not loaded into memory, are irrelevant to the 30,000-character limit.)

Because the length of names vary and some overhead is used for each name, this 30,000-character limit cannot be expressed as an exact number of permissible names. But Digital guarantees that, for each product, at least 2000 user names can appear on reservation lists. In the case of an OpenVMS Cluster, this is a per-node limit.

Note that, because 2000 user names is a per-product limit and because there can be more than one PAK per product, the number of user names on a per-product basis is the *sum* of the user names specified on each PAK.

For example, if three activity PAKs for the DECwrite product were registered on a system and each PAK specified a reservation list with 200 user names, the total number of user names for that product is 600. This is safely below the 30,000-character (2000 user name) limit and below the 5000-character (400 user name) limit.

Examples

1. \$ LICENSE MODIFY /EXCLUDE=(DANCE,THEATR) -
_ \$ /COMMENT="Modified to exclude nodes DANCE & THEATR 10/23/96" -
_ \$ FORTRAN

This command modifies the Fortran license in the License Database so that users cannot access Fortran from the nodes named DANCE and THEATR. A comment is added to the database record for future reference.

2. \$ LICENSE MODIFY /ADD /INCLUDE=(DRAMA) -
_ \$ /COMMENT="Modified to add node named DRAMA 10/23/96" -
_ \$ FORTRAN

This command modifies the Fortran license in the License Database so that users can access Fortran from the node DRAMA in addition to any nodes previously named in the license include list.

3. \$ LICENSE MODIFY /UNITS=1200 FORTRAN
\$ LICENSE LOAD FORTRAN

This command changes the license units on a license with the MOD_UNITS option.

4. \$ LICENSE MODIFY/ TERMINATION_DATE=1-JAN-1999 FORTRAN

Unless an earlier termination date exists, this command sets a new termination date of 1-JAN-1999 for the Fortran license.

5. \$ LICENSE MODIFY/EXCLUDE="" FORTRAN

This command removes all nodes from the previously established exclude list. All nodes now have access to the Fortran license.

LICENSE MOVE

Moves one or more licenses from one License Database to another. When you use LICENSE MOVE, LMF deletes those licenses from the source License Database.

For License Databases not connected to a network, consider using the LICENSE ISSUE /PROCEDURE command.

Format

LICENSE MOVE product-name[...] output-database

Parameters

product-name[,...]

Name or names of products with a license to be moved to the output License Database.

output-database

File specification of the License Database to which the license or licenses should be moved. This database must have been previously created using LICENSE CREATE.

If you enter a partial file specification (for example, specifying only a directory), LMF\$LICENSE is the default file name, and .LDB is the default file type. If you do not specify a device or directory, the current default device and directory are used.

Qualifiers

/ALL

Positional qualifier.

Specifies that all licenses with the given product name should be moved. This qualifier affects only the product name that immediately precedes it in the command string.

/AUTHORIZATION=string

Positional qualifier.

Specifies a string that helps identify the license you want to modify. You must enter the authorization string exactly as it appears on your PAK. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/DATABASE=filespec

Specifies the location of the License Database from which the license or licenses should be moved. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default License Database name and location.

LICENSE LICENSE MOVE

/ISSUER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

/LOG

/NOLOG (default)

Controls whether LICENSE MOVE displays the name of each license that it moves.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. Use this optional qualifier only if you need it to identify the license. This qualifier affects only the product name that immediately precedes it in the command string.

Description

If your license contract allows it, use LICENSE MOVE to move a license from one License Database to another. To move a license, enter LICENSE MOVE, including enough PAK information to clearly identify the license. LICENSE MOVE automatically deletes the license in the source License Database.

Note that the moved license includes only the general PAK information normally provided by LICENSE REGISTER. LICENSE MOVE does not transfer any user-supplied data such as reservation lists, modified termination dates, modified units, include or exclude node lists, or comments.

Examples

1. \$ LICENSE MOVE FORTRAN ALT_SYS2:LMF\$LICENSE.LDB

This command moves the Fortran license in the default License Database to the ALT_SYS2:LMF\$LICENSE.LDB output License Database. This command fails if the default database contains more than one Fortran license.

2. \$ LICENSE MOVE FORTRAN/DATABASE=LMFDATA:LMF\$LICENSE.LDB -
_ \$ ALT_SYS:LMF\$LICENSE.LDB

This command moves the Fortran license in the source License Database, LMFDATA:LMF\$LICENSE.LDB, to the destination License Database, ALT_SYS:LMF\$LICENSE.LDB. This command fails if the source License Database contains more than one Fortran license.

3. \$ LICENSE MOVE FORTRAN/ALL ALT_SYS2:LMF\$LICENSE.LDB

This command moves all Fortran licenses in the default License Database to the output License Database, ALT_SYS2:LMF\$LICENSE.LDB.

4. \$ LICENSE MOVE * ALT_SYS2:LMF\$LICENSE.LDB

This command merges two databases by moving all licenses in the default License Database to the output License Database, ALT_SYS2:LMF\$LICENSE.LDB.

LICENSE REGISTER

Adds a new license to the License Database. A Product Authorization Key (PAK) provides the product name and information you need to register the license. You must enter all information provided by your PAK exactly as it appears.

You can also register a new product license with the command procedure SYS\$UPDATE:VMSLICENSE.COM, which provides a prompt-based interface to the LICENSE REGISTER command.

Format

LICENSE REGISTER *product-name*

Parameter

product-name

Name of the product with a license to register. You can register only licenses that do not currently exist in the License Database. You can register multiple licenses for the same product when they have different authorization numbers. Enter the product name exactly as it appears on your PAK.

You cannot use wildcard characters for the *product-name* parameter with this command.

Qualifiers

/ACTIVITY= { **code**
CONSTANT=integer }

Specifies a license unit code that corresponds to a License Unit Requirement Table (LURT) or to a constant value. If your PAK supplies an activity code, you must enter the code exactly as it appears. The current codes are A, B, C, D, E, F, G, H, and I. If your PAK specifies the keyword CONSTANT, then you must also specify the integer value. This denotes a constant requirement for all System Marketing Models (SMMs) equal to the value given. If your PAK specifies the decimal value 0, then the license has no requirement for that license type. PAK issuers determine the value for this element.

/AUTHORIZATION=string

Specifies a string that helps identify the license you want to register. You must enter the authorization string exactly as it appears on your PAK.

/AVAILABILITY= { **code**
CONSTANT=integer }

Specifies a license unit code that corresponds to a License Unit Requirement Table (LURT) or to a constant value. If your PAK supplies an availability code, you must enter the code exactly as it appears. The current codes are A, B, C, D, E, F, G, H, and I. If your PAK specifies the keyword CONSTANT, then you must also specify the integer value from your PAK. PAK issuers determine the value for this element.

/CHECKSUM=string

Specifies a 17-character verification string created by the PAK issuer for each PAK. The checksum string is presented in the format *n-cccc-cccc-cccc-cccc*, where *n* is an integer and *c* is an alphabetic character from A through P. A PAK

LICENSE LICENSE REGISTER

presents the checksum string with hyphen (-) characters for readability. Because LMF does not count hyphens for authorization, you do not have to enter them. Otherwise, you must enter the checksum string exactly as it appears on your PAK.

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB in an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default database.

/HARDWARE_ID=string

Specifies the identification number of the hardware on which the product is licensed. If your PAK supplies a hardware identification number, you must enter the information exactly as it appears.

/ISSUER=string

Specifies the name of the company (for example, DEC) that issued the PAK for the product. Note that the PAK issuer is often the same as the producer. You must enter the information exactly as it appears on your PAK.

/OPTIONS=[(keyword[,...])]

Specifies LICENSE REGISTER options. If your PAK supplies any license options, you must enter this information exactly as it appears.

Table A-1 describes the available keywords.

Table A-1 LICENSE REGISTER /OPTIONS Keywords

Keyword	Meaning
ALPHA	Identifies Availability Licenses for Alpha systems.
MOD_UNITS	You can modify the number of license units.
NO_SHARE	You cannot use the license on more than one processor in an OpenVMS Cluster environment.
RESERVE_UNITS	To use this license in a cluster, designate it for one node. Issue LICENSE MODIFY with the /INCLUDE qualifier. The license must be assigned to one or more users. Reserve the license using LICENSE MODIFY with the /RESERVE qualifier.
USER	Designates a User License.
VAX_ALPHA	Identifies Availability Licenses that are valid for both OpenVMS VAX and OpenVMS Alpha systems.

If you enter more than one keyword, separate them with commas, and enclose the list in parentheses. You can abbreviate each option to the minimum number of characters needed to uniquely identify it.

/PRODUCER=string

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. You must enter the information exactly as it appears on your PAK.

LICENSE LICENSE REGISTER

/RELEASE_DATE=date

Specifies a product release date such that the license authorizes use of all product versions released on or before the date. If your PAK supplies a product release date, you must enter the information exactly as it appears. The date must be presented in the standard OpenVMS format: *dd-mmm-yyyy*.

/TERMINATION_DATE=date

Specifies the date on which the product license terminates. If your PAK supplies a license termination date, you must enter it exactly as it appears. The date must be presented in the standard OpenVMS format: *dd-mmm-yyyy*.

/TOKEN=string

Specifies a string of information associated with some products. This option can enable or disable certain product features. See your product documentation for details. If your PAK provides token information, you must enter it exactly as it appears.

/UNITS=number

Specifies the number of license units for your license. You must enter the number exactly as it appears on your PAK even if your PAK specifies the MOD_UNITS option.

/VERSION=nn.nn

Limits the version number of the product for which you have a license. Use the format *integer.integer*. If your PAK supplies version information, you must enter it exactly as it appears.

Description

LICENSE REGISTER is the primary LICENSE command. Before you enter a LICENSE REGISTER command, you need a PAK that supplies the information required to enter a license in the License Database.

You can register additional licenses for products that already exist in the License Database. If you register another combinable license in the License Database, LMF combines the license units during a LICENSE LOAD or LICENSE START command. This allows more product availability or activity for the same product.

The checksum number supplied with your PAK is calculated from the other information supplied with the PAK. Thus, you must enter each qualifier necessary to supply information from your particular PAK. If you enter LICENSE REGISTER without a required qualifier, LMF returns a checksum error.

Examples

1. \$ LICENSE REGISTER FORTRAN /ISSUER=DEC /AUTHORIZATION=USA-10 -
_ \$ /PRODUCER=DEC /UNITS=400 /VERSION=5.4 -
_ \$ /AVAILABILITY=F /CHECKSUM=1-HIDN-INDA-COMP-DAHH

This command adds the license for the product Fortran to the default License Database. Fortran becomes licensed using the availability formula with 400 license units available.

LICENSE LICENSE REGISTER

2. \$ LICENSE REGISTER DVNETRTG /ISSUER=DEC /AUTHORIZATION=USA-15 -
_ \$ /PRODUCER=DEC /UNITS=1000 /VERSION=4.0 -
_ \$ /AVAILABILITY=E/CHECKSUM=1-COOD-AGON-EFIC-HING

This command adds the license for the product DVNETRTG (DECnet for OpenVMS Routing) to the default License Database. In the example, DVNETRTG is licensed using the availability formula with 1000 license units.

LICENSE START

Sets up the License Unit Requirement Table (LURT) for your system and loads all licenses that are registered and enabled in the License Database. Because the OpenVMS operating system issues a LICENSE START command during system startup, you should need this command only if system startup fails.

To use this command, you need CMKRNL, SYSNAM, and SYSPRV privileges.

To load the licenses in the License Database of a system with LMF already started, use LICENSE LOAD.

Format

LICENSE START

Parameters

None.

Qualifiers

/DATABASE=filespec

Specifies the location of the License Database. The default file specification is defined by the logical name LMF\$LICENSE, which points to SYS\$COMMON:[SYSEXE]LMF\$LICENSE.LDB on an unmodified OpenVMS system. Use this optional qualifier only if you do not use the default database.

/LOG (default)

/NOLOG

Controls whether LICENSE START displays a message to acknowledge the loading of each product license.

Example

```
$ LICENSE START
```

This command sets up the LURT for your system and loads all the licenses that are registered and enabled in the License Database.

LICENSE LICENSE UNLOAD

LICENSE UNLOAD

Unloads a license, making the product unavailable from the current node. The product license or licenses must be registered in the License Database and must have been previously loaded with an interactive or automatic LICENSE LOAD command. Running processes are allowed to continue to completion.

To use this command, you need CMKRNL, SYSNAM, and SYSPRV privileges.

Format

LICENSE UNLOAD product-name[,...]

Parameter

product-name[,...]

Name of the product to be unloaded. You can unload only licenses that have been loaded. Enter each product name exactly as it appears on its Product Authorization Key (PAK). You cannot use wildcard characters for **product-name**.

Qualifiers

/LOG

/NOLOG (default)

Controls whether LICENSE UNLOAD lists the name of each unloaded license.

/PRODUCER=string

Positional qualifier.

Specifies the name of the company (for example, DEC) that owns the product for which you have a license. The default string for this qualifier is *DEC*. If DEC is not the producer of the product, you must use this qualifier to identify the product. This qualifier affects only the product name that immediately precedes it in the command string. Wildcard characters are not allowed.

Description

LICENSE UNLOAD affects all units for a single product even if the loaded units are combined from multiple licenses. In such a case, you cannot unload the units from a single license. You must unload all the units.

You are not required to unload a loaded license before modifying data in the License Database. To maximize product availability, modify licenses in the License Database first, and then load the changes by entering a LICENSE UNLOAD command followed by a LICENSE LOAD command.

Examples

1. \$ LICENSE UNLOAD/PRODUCER=DEC FORTRAN

This command unloads the Digital Fortran license on the node from which it is entered.

2. \$ LICENSE UNLOAD PASCAL, FORTRAN

This command unloads the DEC Pascal and Digital Fortran licenses on the node from which it is entered.

This appendix provides the following examples of license-related management tasks using the LICENSE commands:

- Registering a system integrated product (Section B.1)
- Issuing LICENSE LIST and SHOW LICENSE commands (Section B.2)
- Restricting product use (Section B.3)

B.1 Registering a System Integrated Product

The following example provides a step-by-step procedure for registering a PAK for Volume Shadowing, which is a System Integrated Product (SIP). This example uses command procedure VMSLICENSE.COM and the following PAK information:

```
ISSUER: DEC
AUTHORIZATION NUMBER: ALS-WM-45789-6666
PRODUCT NAME: VOLSHAD
PRODUCER: DEC
NUMBER OF UNITS: 400
VERSION: 7.1
PRODUCT RELEASE DATE:
KEY TERMINATION DATE: 31-DEC-1997
AVAILABILITY TABLE CODE: E
ACTIVITY TABLE CODE:
KEY OPTIONS: MOD_UNITS
PRODUCT TOKEN:
HARDWARE I.D.:
CHECKSUM: 2-EBID-GOOD-NIGH-OJG
```

Examples

B.1 Registering a System Integrated Product

Execute the VMSLICENSE.COM command procedure to register the Volume Shadowing PAK as follows:

1. Log in to the system manager's account, SYSTEM. Enter the command:

```
$ @SYS$UPDATE:VMSLICENSE
```

The procedure displays the following menu:

VMS License Management Utility Options:

1. REGISTER a Product Authorization Key
2. AMEND an existing Product Authorization Key
3. CANCEL an existing Product Authorization Key
4. LIST the Product Authorization Keys
5. MODIFY an existing Product Authorization Key
6. DISABLE an existing Product Authorization Key
7. DELETE an existing Product Authorization Key
8. COPY an existing Product Authorization Key
9. MOVE an existing Product Authorization Key
10. ENABLE an existing Product Authorization Key
11. SHOW the licenses loaded on this node
12. SHOW the unit requirements for this node

99. EXIT this procedure

Type '?' at any prompt for a description of the information requested. Press Ctrl/Z at any prompt to return to the main menu.

Enter one of the above choices [1]

2. Enter 1. The procedure displays the following message:

```
* Do you have your Product Authorization Key? [YES]:
```

3. Enter Y. The procedure displays the following information and prompts:

Use the REGISTER option to add a new license to a license database. A Product Authorization Key (PAK) provides the product name and information you need to register the license. You must enter all the information provided by your PAK exactly as it appears.

Type '?' at any prompt for a description of the information requested. Press CTRL/Z at any prompt to return to the main menu.

```
Issuer [DEC]:
```

```
Authorization Number []:
```

4. Press Return to specify DEC as the issuer.

Enter the authorization number from the PAK, ALS-WM-45789-6666. The procedure prompts for the following information:

```
Product Name []:
```

5. Enter the product name string, VOLSHAD from the PAK. The procedure prompts for the producer:

```
Producer [DEC]:
```

6. Press Return to specify DEC as the producer. The procedure prompts for the number of units:

```
Number of Units []:
```

7. Enter the number of units, 400. The procedure prompts for the version:

```
Version []:
```

B.1 Registering a System Integrated Product

8. Enter the version number from the PAK, 7.1. The procedure prompts for the key termination date:

Key Termination Date []:

9. Enter the key termination date, 31-DEC-1997. The procedure prompts for the following information:

Availability Table Code []:

Activity Table Code []:

10. Enter the availability table code, E. Press Return after the Activity Table Code prompt. The procedure prompts for the following information:

Key Options []:

Product Token []:

Hardware-Id []:

11. Enter the option MOD_UNITS after the Key Options prompt. Press Return after the Product Token prompt and the Hardware-ID prompt. The procedure prompts for the checksum:

Checksum []:

12. Enter the checksum, 1-EBID-GOOD-NIGH-OJJG.

Note

The checksum string always begins with a number. The other 16 characters are always alphabetic characters from A through P.

The procedure displays the information you entered. For example:

Here is a list of the license information just entered:

```

      Issuer: DEC
  Authorization: ALS-WM-45789-6666
      Producer: DEC
   Product Name: VOLSHAD
         Units: 400
   Release Date:
     Version: 7.1
  Termination Date: 31-DEC-1997
     Availability: E
       Activity:
         Options: MOD_UNITS
           Token:
   Hardware ID:
     Checksum: 1-EBID-GOOD-NIGH-OJJG

```

Is that correct? [YES]:

13. Compare the information on the screen with the information on the PAK. If the information is correct, enter Y. Otherwise enter N.

Note

If you enter any of the information incorrectly, you receive an error message, and the license is not registered. A checksum error can result when you enter incorrect information for other items on the PAK. If you get an error, carefully check all the data that you entered.

Examples

B.1 Registering a System Integrated Product

14. When the procedure displays the following question, enter Y.

Do you wish to make corrections? [YES]:

15. To make corrections, the procedure steps you through all of the questions again with the data you just entered as defaults for each data field. For correct data, press Return. To replace incorrect data, enter the new data. To cancel data without entering new data, enter the backslash (\) character.

If you entered all the information correctly, the procedure displays the following message:

```
Registering VOLSHAD license in SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB...
```

16. After the license is successfully registered, the procedure asks if you want to load the license on the current node, as follows:

Do you want to LOAD this license on this system? [YES]:

- If you registered the PAK on a standalone system and want to make the software available (active) immediately, enter Y.
- If you registered the license in an OpenVMS Cluster environment but do not want to make it available (active) on the current node, enter N.

In this example, assume the license is being registered in an OpenVMS Cluster environment and that you do not want it loaded on the current node. Enter N to complete this license registration. Note that you must load the Volume Shadowing license before you can use the product. See Step 18.

The procedure returns you to the first menu and prompt as follows:

VMS License Management Utility Options:

1. REGISTER a Product Authorization Key
2. AMEND an existing Product Authorization Key
3. CANCEL an existing Product Authorization Key
4. LIST the Product Authorization Keys
5. MODIFY an existing Product Authorization Key
6. DISABLE an existing Product Authorization Key
7. DELETE an existing Product Authorization Key
8. COPY an existing Product Authorization Key
9. MOVE an existing Product Authorization Key
10. ENABLE an existing Product Authorization Key
11. SHOW the licenses loaded on this node
12. SHOW the unit requirements for this node
99. EXIT this procedure

Type '?' at any prompt for a description of the information requested. Press Ctrl/Z at any prompt to return to the main menu.

Enter one of the above choices [1]

17. To register another PAK, enter 1. Then respond to the questions, again entering information from a license PAK. For this example assume you are finished. Enter 99 to exit the procedure. You have registered the license for Volume Shadowing.
18. Load the license on the desired node by logging in to that node and entering the LICENSE LOAD VOLSHAD command at the DCL prompt (\$).

B.2 Difference Between LICENSE LIST and SHOW LICENSE

B.2 Difference Between LICENSE LIST and SHOW LICENSE

This example shows the differences between the LICENSE LIST command, which displays license information stored on disk, and the SHOW LICENSE command, which displays license information in memory. The first command registers a Fortran license, as follows:

```
$ LICENSE REGISTER FORTRAN /ISSUER=DEC /AUTHORIZATION=USA-10 -
_$ /PRODUCER=DEC /UNITS=400 /VERSION=5.4 -
_-$ /AVAILABILITY=F /CHECKSUM=1-HIDN-INDA-COMP-DAHH
```

This command adds the license for the product Digital Fortran to the default License Database. To see the information registered in the database, enter a LICENSE LIST command, as follows:

```
$ LICENSE LIST /FULL FORTRAN
Use Ctrl/Z to exit, PF3-PF4 for Previous-Next Screen and Arrow keys to Scroll.
```

```
License Management Facility V1.2
```

```
License Database File: SY$COMMON:[SYSEXE]LMF$LICENSE.LDB
Created on: 17-AUG-1992
Created by user: GRAHAM
Created by LMF Version: V1.1
```

```
-----
Issuer: DEC
Authorization: USA-10
Product Name: FORTRAN
Producer: DEC
Units: 400
Version: 5.4
Release Date: (none)
PAK Termination Date: (none)
Availability: F (Layered Products)
Activity: 0
Options:
Hardware ID:

Revision Level: 1
Status: Active
Command: REGISTER
Modified by user: LESH
Modified on: 21-AUG-1996 14:32:23.41
```

Notice that for status the LICENSE LIST command displays *Active*. This means the registered license is enabled for loading, that it has not been disabled or canceled. It does not necessarily mean the Fortran license was loaded with a LICENSE LOAD command. Because the LICENSE LIST command sees only the License Database on disk, you must enter a SHOW LICENSE command to refer to the License Database in memory to determine whether a license is active on the current system. For example:

```
$ SHOW LICENSE/FULL
Active licenses on node BIODTL:
```

Examples

B.2 Difference Between LICENSE LIST and SHOW LICENSE

```
CRYPTICALMENT
  Producer: DEC
  Units: 400
  Version: 7.1
  Release Date: (none)
  Termination Date: 31-DEC-1997
  Availability: E (System Integrated Products)
  Activity: 0
  MOD_UNITS
```

```
VAX-VMS
  Producer: DEC
  Units: 400
  Version: 5.4
  Release Date: (none)
  Termination Date: (none)
  Availability: A (VMS Capacity)
  Activity: 0
  MOD_UNITS
  NO_SHARE
```

The SHOW LICENSE command in this example displays all the active licensed products on the current node named BIODTL; the Fortran license has not yet been loaded.

After you load the Fortran LICENSE, the SHOW LICENSE command displays the license. For example:

```
$ LICENSE LOAD FORTRAN
LICENSE-I-LOADED, DEC FORTRAN was successfully loaded with 400 units
$ SHOW LICENSE /FULL
```

Active licenses on node BIODTL:

```
CRYPTICALMENT
  Producer: DEC
  Units: 400
  Version: 7.1
  Release Date: (none)
  Termination Date: 31-DEC-1997
  Availability: E (System Integrated Products)
  Activity: 0
  MOD_UNITS
```

```
FORTRAN
  Producer: DEC
  Units: 400
  Version: 5.4
  Release Date: (none)
  Termination Date: (none)
  Availability: F (Layered Products)
  Activity: 0
```

```
VAX-VMS
  Producer: DEC
  Units: 400
  Version: 6.0
  Release Date: (none)
  Termination Date: (none)
  Availability: A (VMS Capacity)
  Activity: 0
  MOD_UNITS
  NO_SHARE
```

B.3 Restricting Product Use

This example illustrates how LMF restricts use of a product when insufficient license units are registered for it. For the example, the product DEC BASIC is installed and its license is registered with zero availability units and the MOD_UNITS option. Zero-unit licenses provide authorization on any processor. In the first LICENSE MODIFY command, however, the license is changed to a 1500-unit Availability License:

```
$ LICENSE UNLOAD BASIC
$ LICENSE MODIFY/UNITS=1500 BASIC
```

The next command attempts to load the registered license on a VAX 8800 system:

```
$ LICENSE LOAD BASIC
%LICENSE-W-NOLOAD, license was not loaded for BASIC
-LICENSE-F-EXCEEDED, attempted usage exceeds active license limits
```

Because the VAX 8800 system in this example requires 2000 license units to authorize DEC BASIC, the LICENSE LOAD command fails. The next command attempts to invoke DEC BASIC despite the failed LICENSE LOAD command:

```
$ BASIC
%LICENSE-F-NOAUTH, DEC BASIC use is not authorized on this node
-LICENSE-F-NOLICENSE, no license is active for this software product
-LICENSE-I-SYSMGR, please see your system manager
```

Note that the attempt to invoke DEC BASIC fails. Because the LICENSE LOAD command failed, DEC BASIC use is unauthorized on the current node. The solution is to modify the license again using a value obtained by using the DCL command SHOW LICENSE /UNIT_REQUIREMENTS. The new number of license units provides sufficient units to allow a successful LICENSE LOAD command, which authorizes use of the product.

```
$ LICENSE MODIFY/UNITS=2000 BASIC
$ LICENSE LOAD BASIC
%LICENSE-I-LOADED, DEC BASIC was successfully loaded with 2000 units
```

After the license is loaded, the product can be invoked, as follows:

```
$ BASIC
DEC BASIC V3.2

Ready
```

Glossary

This glossary defines the LMF-related terms used in the *OpenVMS License Management Utility Manual*.

active license

A license that has been enabled. The term *active* appears in displays produced by LICENSE LIST and has been retained to prevent automated procedures from breaking.

Activity License

A license that defines the allowed number of concurrent uses of a product. Each product defines an activity as either an interactive user, a running process, or a job. For example, a 4-Activity license may have enough license units to allow four users to access the product simultaneously.

authorization number

The unique number assigned by the PAK issuer to a specific PAK. The PAK issuer name and authorization number identify a license.

Availability License

A license that makes a product available to all the users of a system. LMF can activate a product when the number of license units on a license matches or exceeds the license unit rating for the current processor. Every System Marketing Model (SMM) has a series of license unit requirements, typically related to performance, that define how many license units are required to make a product available.

checksum

An encoded number calculated from the other information supplied with a PAK. The checksum string always begins with the number 1, which is the only number in the string. The other sixteen characters are always alphabetic characters from A through P.

hardware identifier

An optional string that identifies a particular hardware unit.

Integrated Software Business Technologies

The name for Digital's business plan that integrates consolidated software distribution, online documentation, and software access management. With this plan more products will be available on compact disc read-only memory (CD-ROM) where software access is authorized by PAKs and LMF.

license

(In this manual) PAK information for a software product that is registered in the License Database.

license combination

A method for using the license units from two or more combinable licenses to provide more product availability. Two licenses with 100 units each combine to equal a 200-unit license. You may use license combination, for example, when you add a new processor to a VAXcluster environment.

License Database

A collection of interrelated data stored on a disk and accessed through LMF software. The default location for the database is SYS\$COMMON:[SYSEXEXE]LMF\$LICENSE.LDB. Each record in the License Database corresponds to a license. Sometimes OpenVMS licenses are registered in a second License Database located in SYS\$SPECIFIC:[SYSEXEXE]LMF\$SYSTEM.LDB.

License Management Facility (LMF)

A variety of system-level software components used to maintain software license information in the License Database of the OpenVMS operating system. LMF is a management tool; the terms and conditions of your product contract determine your legal use of software.

license registration

The entry in the License Database of a Product Authorization Key (PAK) to add a new license. To register a license, enter the LICENSE REGISTER command, or respond to prompts from the VMSLICENSE.COM command procedure.

license sharing

A method to allow more than one processor to use the license units from a single license. In OpenVMS, this refers to sharing licenses among nodes in an OpenVMScluster environment. Licenses that specify the NO_SHARE option cannot be shared.

license unit

A basic measurement that Digital uses to specify how much product use a license provides. Digital gives each license intended to be used with LMF a size, specified in license units. For example, a license can be a 50-unit license, a 200-unit license, or a 700-unit license.

License Unit Requirement Table (LURT)

Online tables provided by Digital that specify a series of license unit requirements, essentially performance ratings, for each System Marketing Model. Processors that provide more performance (other ratings may be unrelated to performance) have greater license unit requirements. The default file name for the LMF LURTs is SYS\$COMMON:[SYSEXEXE]LMF\$LURT.DAT.

LURT

See License Unit Requirement Table.

PAK

See Product Authorization Key.

PAK issuer

The LMF name for the company that creates the license contract for the software. The PAK issuer name and license authorization number uniquely identify a license. PAK issuers are usually the same as software producers but can operate under agreement with the producer.

Personal Use License

A license that designates the names of specific users for unlimited use of a product. Each product defines a user as either an interactive user, a running process, or a job. LMF requires user names associated with this kind of license.

Product Authorization Key (PAK)

License information that you must register in the License Database in order to use the product. It is produced by a PAK issuer and delivered to you by mail, electronic transfer, or by telephone.

product identification

The software producer name and product name. Together they uniquely identify a software product for licensing.

record

A collection of data fields in the License Database that define a license at any one time.

release level

Uniquely identified by either a product release date or product release version. To authorize a product for use by license version number, the product release level (in the form *nn.nn*) must be less than or equal to the license version number. For example, license version number 4.4 allows operation of product release levels 4.3 and 4.4, but not 4.5.

reservation list

A list that contains the names of users authorized access to a product that is registered with a Personal Use License.

selection weight

An arbitrary attribute of a license, assigned by LMF, to control the order in which different licenses for a product are loaded. You can modify the selection weight with the /SELECTION_WEIGHT qualifier to LICENSE MODIFY.

software license

A contract between a license issuer (Digital) and a license receiver (customer) that grants permission to use a specific software product as described by the applicable Software Product Description (SPD) and the terms and conditions of the license contract. A PAK supplies the information that results from a software license contract.

software producer

The company that owns the software being licensed. Software producers are usually the same as PAK issuers but can operate under agreement with the issuer.

Software Product Description (SPD)

The legal document that describes the software product. This document contains the precise product release level that comprises the product version and official product release date.

System Marketing Model (SMM)

The model name of a computer system as used in marketing and pricing. The SMM is generally the name on the front panel of the processor cabinet. LMF uses this value rather than hardware CPU-type because different marketing models may use the same CPU with different pricing and licensing rules.

termination date

Defines when a license contract is no longer valid, and when LMF no longer authorizes product use.

token

A text string specific to each product used to control additional licensing features. Digital does not currently use tokens; however, LMF accepts them for use by certain third-party products.

User License

Defines the number of users allowed unlimited use of a product. Each product defines a user as either an interactive user, a running process, or a job. LMF requires user names associated with this kind of license.

Index

A

Activity License, 1-7
Availability License
 restrictions for Alpha and VAX, 1-6

C

Checksum error, avoiding, A-33
Codes for license types, 1-3, 1-5, A-31
Combination rules, 1-11
 dates and version numbers, 1-12
Commands
 syntax descriptions, A-1 to B-1
Controlling license loading order, 2-22
COPY command, A-2 to A-4
CREATE command, A-5

D

DELETE command, A-6 to A-8
DISABLE command, A-9 to A-10

E

ENABLE command, A-11 to A-12

G

Group License, 1-10

H

History records
 in the License Database, 1-3, 2-3

I

Installing products, 2-1
 sequence with license registration, 2-5
ISSUE command, A-13 to A-15

L

License
 Activity
 sharing units, 1-9
 authorization
 by user, 1-7
 authorization time, 1-8
 automatic registration, 2-1
 combination, 1-6, 1-9, 1-11
 /COMBINE qualifier, A-23
 dates and version numbers, 1-12
 /NOCOMBINE qualifier, A-23
 NO_SHARE option, 1-11
 controlling loading, 2-15, 2-17
 controlling user access to, 2-20
 defined, 1-1
 examples of registration, B-1
 loading, 2-1, 2-14, A-20, A-35
 loading in an OpenVMS Cluster environment,
 2-15
 managing after registration, 2-16
 methods for registering, 2-6
 modifying, 2-1, A-23
 modifying to include a reservation list, 2-20
 modifying to include SCS node name, 2-2,
 2-18
 MOD_UNITS option, 2-17
 multiples of, 2-19
 multiples with LICENSE LOAD command,
 A-21
 multiples with LICENSE UNLOAD, A-36
 NO_SHARE option, 2-19
 order of checking
 /SELECTION_WEIGHT qualifier, A-26
 product self-registration, 2-6
 providing activity use, 1-8
 providing availability in an OpenVMS Cluster
 environment, 1-7
 registration, A-31
 registering a SIP, 2-4
 registering a System Integrated Product, B-1
 registration
 best time, 2-5
 registration and product installation, 2-5
 RESERVE_UNITS option, A-25

License (cont'd)

- restricting access to, 2-17
- sharing in an OpenVMS Cluster environment, 1-6, 2-18
- types, 1-5 to 1-10
 - Activity, 1-7
 - Availability, 1-5
 - Group, 1-10
 - Personal Use, 1-9
- unlimited units, 1-4, 2-17
- unloading, A-36
 - on shutdown, 2-18
- with the NO_SHARE option, 2-2
- zero units, 1-4, 2-17

License checking

- specifying order of, A-26

LICENSE commands

- entering long, A-33
- list of, A-1

License Database, 1-2, 2-1

- backing up, 2-2
- common, with multiple system disks, 2-2
- creating, A-5
- defining a logical name for, 2-2
- definition, 1-2
- fields, 1-2
- history records, 1-3, 2-3
- location, 2-2
- merging, A-30
- special location for OpenVMS system, 2-2
- versions, A-18

License Management Facility (LMF), 1-1

License types

- codes for, 1-3
- RESERVE_UNITS option, 1-9
- User, 1-10

License Unit Requirement Tables (LURTs)

- file name, 1-3

License units, 1-3, 1-8

- allocating, 1-5
- combination, 1-6, 1-11
- CONSTANT value, 1-3
- creating a pool of, 1-11
- modifying, 2-20
- providing enough, 1-6, 1-8
- providing more, 1-6, 1-9
- sharing of activity, 1-9

Licensing

- steps, 2-1 to 2-6

LIST command, A-16 to A-19

- active status, B-5
- difference from SHOW LICENSE command, A-16, B-5
- displaying a license with, 2-13
- example, 2-13

List size restrictions, reservation list, A-27

LMF\$DISPLAY_OPCOM_MESSAGE, 2-22

LOAD command, A-20 to A-22

- in an OpenVMS Cluster environment, A-22

LURT (License Unit Requirement Table)

- setting up, A-35

M

MODIFY command, A-23 to A-28

- controlling reservation list size, 2-21
- using the /ADD qualifier, 2-21
- using the /AUTHORIZATION qualifier, 2-19
- using the /EXCLUDE qualifier, 2-18
- using the /INCLUDE qualifier, 2-18, 2-19
- using the /RESERVE qualifier, 2-20
- using the /UNITS qualifier, B-7

MOD_UNITS option, 2-20

- example, B-7

MOVE command, A-29 to A-30

N

NO_SHARE option, 2-1

/NO_SHARE qualifier

- controlling license combination, 1-11

O

OpenVMS Cluster

- NO_SHARE, 2-19

OpenVMS Cluster environment

- controlling node access, 2-18
- managing licenses in, 2-18
- NO_SHARE option, 2-1
- PAK restrictions, 1-6
- providing more availability in, 1-7
- registering licenses for, 2-6
- sharing Activity License units in, 1-9

P

PAK, 2-1

- getting, 2-3
- issuers, 2-4
- registering, B-1
- replica, A-13
- transfer methods, 2-4

Performance ratings, 1-3

Personal Use License, 1-9, A-25

Producer

- difference between PAK issuer and, 2-4

Product

- installing, 2-1, 2-5
- matching release and license, 2-1

R

REGISTER command, 2-1, A-31 to A-34

Reservation lists, A-25

access denied to users on, 2-21

matching user number to combined units, 2-21

replacing many with one, 2-22

restrictions, A-27

used with LICENSE MODIFY command, 2-20

RESERVE_UNITS option, 1-9, A-25

S

SCS node names, used with LICENSE MODIFY command, 2-2, 2-18

Selection weight

controlling license loading order, 2-22

/SELECTION_WEIGHT, 2-22

SHOW LICENSE command

difference from LICENSE LIST command, A-16, B-5

examples, B-5

SMM (System Marketing Model), 1-4

START command, A-35

Syntax

of LICENSE commands

descriptions, A-1 to B-1

System disks, using multiple, 2-2

System Management utility (SYSMAN)

licenses

loading, 2-15, A-22

System memory, transferring from the License Database, 2-15

U

Unlimited units, 1-4, 2-17

UNLOAD command, A-36

restricting access with, 2-17

User Licenses, 1-10

V

VMSLICENSE.COM, 2-1

batch processing with, 2-10

canceling default data with, 2-9

creating data files for, 2-11

default value rules used in data files, 2-12

examples, 2-7, B-2

parameters used in data files, 2-12

registering a license with, 2-6, 2-7, B-2

using data files with, 2-10

Z

Zero units, 1-4, 2-17

