

**ARM<sup>®</sup> DS-5**

**Version 5**

**License Management Guide**

**ARM<sup>®</sup>**

**ARM® DS-5****License Management Guide**

Copyright © 2011-2016 ARM Limited or its affiliates. All rights reserved.

**Release Information****Document History**

<b>Issue</b>	<b>Date</b>	<b>Confidentiality</b>	<b>Change</b>
A	29 March 2011	Non-Confidential	First release
B	30 September 2011	Non-Confidential	Update for DS-5 version 5.7
C	29 February 2012	Non-Confidential	Update for DS-5 version 5.9
D	29 July 2012	Non-Confidential	Update for DS-5 version 5.11
E	12 October 2012	Non-Confidential	Update for DS-5 version 5.12
F	15 March 2013	Non-Confidential	Update for DS-5 version 5.14
G	14 March 2014	Non-Confidential	Update for DS-5 version 5.18
H	27 June 2014	Non-Confidential	Update for DS-5 version 5.19
I	17 October 2014	Non-Confidential	Update for DS-5 version 5.20
J	20 March 2015	Non-Confidential	Update for DS-5 version 5.21
K	15 July 2015	Non-Confidential	Update for DS-5 version 5.22
L	15 October 2015	Non-Confidential	Update for DS-5 version 5.23
M	15 March 2016	Non-Confidential	Update for DS-5 version 5.24
N	15 July 2016	Non-Confidential	Update for DS-5 version 5.25
O	18 November 2016	Non-Confidential	Update for DS-5 version 5.26

**Non-Confidential Proprietary Notice**

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of ARM. **No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.**

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED “AS IS”. ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, ARM makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, third party patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to ARM’s customers is

not intended to create or refer to any partnership relationship with any other company. ARM may make changes to this document at any time and without notice.

If any of the provisions contained in these terms conflict with any of the provisions of any signed written agreement covering this document with ARM, then the signed written agreement prevails over and supersedes the conflicting provisions of these terms. This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Words and logos marked with ® or ™ are registered trademarks or trademarks of ARM Limited or its affiliates in the EU and/or elsewhere. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners. Please follow ARM's trademark usage guidelines at <http://www.arm.com/about/trademark-usage-guidelines.php>

Copyright © 2011-2016, ARM Limited or its affiliates. All rights reserved.

ARM Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

LES-PRE-20349

### **Confidentiality Status**

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Unrestricted Access is an ARM internal classification.

### **Product Status**

The information in this document is Final, that is for a developed product.

### **Web Address**

<http://www.arm.com>

# Contents

## ARM® DS-5 License Management Guide

### **Preface**

<i>About this book</i> .....	10
------------------------------	----

### **Chapter 1**

#### **Supported Platforms**

1.1	<i>Supported platforms</i> .....	1-13
1.2	<i>Tool licensing schemes</i> .....	1-14
1.3	<i>Types of license</i> .....	1-15
1.4	<i>Node-locked license scheme</i> .....	1-16
1.5	<i>Floating license scheme</i> .....	1-17
1.6	<i>About the ARM License Manager</i> .....	1-20
1.7	<i>About the ARM web licensing portal</i> .....	1-21

### **Chapter 2**

#### **Requesting Your License**

2.1	<i>Information required for a license</i> .....	2-23
2.2	<i>Manually obtaining a license from the ARM web licensing portal</i> .....	2-24

### **Chapter 3**

#### **Installing a Node-locked License Using the ARM License Manager**

3.1	<i>Installing a node-locked license using the ARM License Manager</i> .....	3-28
3.2	<i>Installing a node-locked license manually (Windows only)</i> .....	3-30
3.3	<i>Installing a node-locked license manually (Unix/Linux only)</i> .....	3-31

### **Chapter 4**

#### **Using a Floating License**

4.1	<i>About configuring the client licensing</i> .....	4-33
4.2	<i>Setting up your licensing server environment with the ARM License Manager</i> .....	4-34

4.3	Setting up your licensing server environment from the Control Panel (Windows clients only) .....	4-36
4.4	Setting up your licensing server environment at the command line (Unix/Linux clients only) .....	4-37

## Chapter 5

### Frequently Asked Questions about Licensing ARM Products

5.1	How does FlexNet find a license file? .....	5-39
5.2	Can I use FlexNet with more than one product? .....	5-40
5.3	How do I merge license files? .....	5-41
5.4	How do I find my product serial number? .....	5-43
5.5	How do I find my host ID? .....	5-44
5.6	How do I move my license to another computer? .....	5-48
5.7	Do upgrade or update seats replace existing seats? .....	5-49
5.8	Do the ARM development tools support BORROW? .....	5-50
5.9	Do ARM licenses support queuing? .....	5-51
5.10	Why can I not find the lmhostid utility? .....	5-52
5.11	Format of a license entry .....	5-53

## Chapter 6

### Troubleshooting Client Configuration

6.1	License apparently missing error .....	6-55
6.2	My node-locked tools fail with FlexNet error -8: Invalid license key (inconsistent authentication code) .....	6-56
6.3	How can I change the order in which my license sources are accessed? .....	6-57
6.4	Optimizing floating license checkout times .....	6-58
6.5	Can I use a remote floating license server? .....	6-60
6.6	Why am I getting FlexNet error -15: Cannot connect to floating license server system? .....	6-61

## Chapter 7

### Troubleshooting Server Configuration

7.1	How can I run my FlexNet server more securely? .....	7-64
7.2	Server debug log reports "Invalid license key (inconsistent authentication code)" .....	7-65
7.3	Can I define the order in which licenses in a mixed version license file are issued? .....	7-66
7.4	How can I tell how many floating licenses are in use? .....	7-67
7.5	Server debug log reports "(armlmd) cannot open lock file" or "cannot communicate with lmgrd" .....	7-68
7.6	What version of the floating license server daemons must I use? .....	7-69
7.7	How can I limit user access to particular licensed features? .....	7-70
7.8	About the options file .....	7-71
7.9	Configuring your floating license server to use the options file .....	7-72
7.10	Why am I getting "DENIED" messages in the server log? .....	7-73
7.11	How can I manage the size of the server debug log file? .....	7-74
7.12	Suppressing messages from the server log .....	7-75
7.13	Restarting the floating license server .....	7-76
7.14	Starting a new server debug log .....	7-77

## Chapter 8

### Setting Up a Floating License Server

8.1	About the license management utilities .....	8-79
8.2	About installing the server software .....	8-80
8.3	Getting the ARM floating license server utilities .....	8-81

8.4	<i>Installing server software on Windows</i> .....	8-82
8.5	<i>Installing server software on Unix/Linux</i> .....	8-83
8.6	<i>Modifying your floating license file</i> .....	8-84
8.7	<i>Starting a Windows server from the graphical user interface</i> .....	8-86
8.8	<i>Starting a Windows server from the command line</i> .....	8-89
8.9	<i>Starting a Unix/Linux server from the command line</i> .....	8-90
8.10	<i>About shutting down a floating license server</i> .....	8-91
8.11	<i>Shutting down a Windows server</i> .....	8-92
8.12	<i>Shutting down a Unix/Linux server</i> .....	8-93

## **Chapter 9**

### **ARM License Manager Diagnostic Utility**

9.1	<i>About the arlmldiag utility</i> .....	9-95
9.2	<i>arlmldiag utility</i> .....	9-96

# List of Figures

## ARM® DS-5 License Management Guide

<i>Figure 1-1</i>	<i>Single-server model</i> .....	<i>1-18</i>
<i>Figure 1-2</i>	<i>Three-server model</i> .....	<i>1-19</i>
<i>Figure 2-1</i>	<i>View and edit licenses page</i> .....	<i>2-24</i>
<i>Figure 2-2</i>	<i>Add License dialog box</i> .....	<i>2-25</i>
<i>Figure 2-3</i>	<i>Choose network interface</i> .....	<i>2-25</i>
<i>Figure 2-4</i>	<i>Obtain a new license via www.arm.com</i> .....	<i>2-26</i>
<i>Figure 3-1</i>	<i>Add License dialog box</i> .....	<i>3-28</i>
<i>Figure 3-2</i>	<i>Enter existing license details page</i> .....	<i>3-29</i>
<i>Figure 4-1</i>	<i>Add License dialog box</i> .....	<i>4-34</i>
<i>Figure 4-2</i>	<i>Enter existing license details page</i> .....	<i>4-35</i>
<i>Figure 8-1</i>	<i>Selecting Configuration using Services</i> .....	<i>8-86</i>
<i>Figure 8-2</i>	<i>Server setup dialog</i> .....	<i>8-87</i>

# List of Tables

## ARM® DS-5 License Management Guide

<i>Table 5-1</i>	<i>Commands for obtaining host IDs .....</i>	<i>5-46</i>
------------------	--	-------------



# Preface

This preface introduces the *ARM® DS-5 License Management Guide*.

It contains the following:

- [About this book on page 10.](#)

## About this book

This book describes how to use FlexNet (formerly known as FLEXnet and FLEXlm) to enable ARM software tools. It contains generation and installation instructions for node-locked and floating licenses in addition to FAQs and hints for troubleshooting license problems.

## Using this book

This book is organized into the following chapters:

### **Chapter 1 Supported Platforms**

Gives an overview of the license management software.

### **Chapter 2 Requesting Your License**

Describes how to request a license for your ARM software.

### **Chapter 3 Installing a Node-Locked License Using the ARM License Manager**

Describes how to install a node-locked license for your ARM software product after you have retrieved the license from the ARM web licensing portal.

### **Chapter 4 Using a Floating License**

Describes how to configure your client computer to obtain the license for its ARM tools from a floating license server.

### **Chapter 5 Frequently Asked Questions about Licensing ARM Products**

Describes licensing issues that are not specific to a particular licensing configuration.

### **Chapter 6 Troubleshooting Client Configuration**

Describes issues relating to the configuration of clients in either node locked or floating license systems.

### **Chapter 7 Troubleshooting Server Configuration**

Describes issues relating to the configuration of floating license systems.

### **Chapter 8 Setting Up a Floating License Server**

Describes the installation of FlexNet server software and the configuration of the client machines.

### **Chapter 9 ARM License Manager Diagnostic Utility**

Describes the `armlmddiag` license manager diagnostic utility.

## Glossary

The ARM Glossary is a list of terms used in ARM documentation, together with definitions for those terms. The ARM Glossary does not contain terms that are industry standard unless the ARM meaning differs from the generally accepted meaning.

See the [ARM Glossary](#) for more information.

## Typographic conventions

*italic*

Introduces special terminology, denotes cross-references, and citations.

**bold**

Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

monospace

Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

monospace

Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

monospace *italic*

Denotes arguments to monospace text where the argument is to be replaced by a specific value.

**monospace bold**

Denotes language keywords when used outside example code.

<and>

Encloses replaceable terms for assembler syntax where they appear in code or code fragments.  
For example:

```
MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>
```

**SMALL CAPITALS**

Used in body text for a few terms that have specific technical meanings, that are defined in the *ARM glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

## Feedback

### Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

### Feedback on content

If you have comments on content then send an e-mail to [errata@arm.com](mailto:errata@arm.com). Give:

- The title *ARM® DS-5 License Management Guide*.
- The number ARM DUI05770.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

————— **Note** —————

ARM tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

## Other information

- [ARM Information Center](#).
- [ARM Technical Support Knowledge Articles](#).
- [Support and Maintenance](#).
- [ARM Glossary](#).

# Chapter 1

## Supported Platforms

Gives an overview of the license management software.

It contains the following sections:

- *1.1 Supported platforms* on page 1-13.
- *1.2 Tool licensing schemes* on page 1-14.
- *1.3 Types of license* on page 1-15.
- *1.4 Node-locked license scheme* on page 1-16.
- *1.5 Floating license scheme* on page 1-17.
- *1.6 About the ARM License Manager* on page 1-20.
- *1.7 About the ARM web licensing portal* on page 1-21.

## 1.1 Supported platforms

You must ensure that Eclipse for DS-5 is supported on your chosen platform before installing it.

If you are also setting up one or more floating license servers, you must confirm that the license management utilities are supported on the server.

### Client platforms

———— **Caution** ————

Attempting to use an unsupported client platform can result in unexpected behavior.

---

Check the product documentation to confirm what specific operating system variants are supported.

### Floating license server platforms

———— **Caution** ————

Attempting to use an unsupported server platform can result in licensing failures.

---

Check the product documentation to confirm what specific operating system variants are supported.

The floating license server platform is not required to be the same as the tools platform. For example, you might have your development tools installed on Windows and use a Linux floating license server. See your product documentation for further information.

License management of floating licenses for ARM applications requires TCP/IP software to be installed, configured, and running on every relevant computer.

### Cloud-based license servers

ARM supports locking floating licenses to the network interface MAC address only, and does not support any other cloud-specific hostid mechanisms.

To operate a FlexNet server on a cloud-based computer, it is therefore necessary to ensure that the computer has a stable MAC address permanently assigned to it, and that the FlexNet utilities can detect that MAC address. Use the usual FlexNet utilities to check that the MAC address can be found. For example, use the `lmutil lmhostid` command.

## 1.2 Tool licensing schemes

Eclipse for DS-5 is license managed by FlexNet Publisher software from Flexera, that uses FlexNet technology.

This means that a valid license file must be installed before DS-5 can be used. The following licensing schemes are available:

- Node-locked license scheme.
- Floating license scheme.

The licensing scheme that is used is selected when the product is purchased. If you want to change the licensing scheme that you are using, you must contact ARM License Support.

### Licensing options for remote working

The following options are available for remote users:

- Install a license server with a floating license. Remote users can check-out licenses from the server, provided that network access is available.
- For a node-locked license, obtain a license that is registered to:
  - The host ID of the machine used by the remote user.
  - The MAC address of a USB Ethernet card. This card can then be plugged into any machine that has the ARM DS-5 tools installed.

### Related concepts

[1.4 Node-locked license scheme on page 1-16.](#)

[1.5 Floating license scheme on page 1-17.](#)

## 1.3 Types of license

A license is a text file containing information about the type of license, how long the license lasts and what applications it is used for.

The following types of license are supported:

### Permanent

A permanent license does not have an expiry date. Using the `fromelf` utility as an example, the format of a permanent license is:

```
INCREMENT fromelf armlmd 5.0201105 permanent ...
```

If you fulfill your license in May 2010 you can use 5.0 and any new patches that are released up to end of May 2011.

If your license period is limited by a *Service and Maintenance* (S&M) agreement, and the period lapses, your version of the product, together with any patches you received within that period continues to function as normal. However, if you want to use a patch that is later than this date, you must purchase another S&M license. If you then purchase an additional two-year license, the new license includes:

```
UPGRADE fromelf armlmd 5.0201305 permanent ...
```

If you fulfill your license in May 2010, 5.0201305 gives a user three years entitlement to download and use patches.

### Term

A term license has a specific expiry date. For example:

```
INCREMENT fromelf armlmd 5.0201205 31-July-2012 ...
```

This means you can use version 5.0 and any new patches to `fromelf` that are released before the end of May, but you get a license expiry error after the 31st July 2012.

### Related references

[5.11 Format of a license entry on page 5-53.](#)

## 1.4 Node-locked license scheme

A node-locked license locks the tools to a particular computer, or more accurately to a particular host ID.

A node-locked license is a good choice for an installation that will be used by one user on a single computer.

The tools license is tied to a particular host ID. If the device you used for the host ID fails, you cannot use the licensed product. The host ID used cannot be changed without a replacement license being supplied by ARM.

The following example shows part of a typical node-locked license. This license allows the use of the specified ARM application, `fromelf`, on the single specific computer that is identified by its `hostid` value of `001F12FEAB43`.

### Typical node-locked license extract

```
INCREMENT fromelf arm1md 5.0201105 permanent uncounted 6AB6805Z33A6 \  
HOSTID=001F12FEAB43 ISSUER="ARM Limited" \  
NOTICE="For support see http://www.arm.com/support/" \  
SN=ABCDE-12345-ZYXWV SIGN="9280 B58A 6ED0 814C 4D0C C3AD \  
5D85 ABC7 5FDF 5555 7772 F529 52E4 0C52 C836 1812 4492 3263 \  
61F4 5E7B 373F A4F4 0E6C 824B CB71 B8D8 2DB8 3D39 994F 7E1E" \  
SIGN2="ED41 1634 F698 AC5E 6645 E5B4 CEDA 97D1 A5D2 93E1 6E01 \  
937F CDD7 27A9 921B 1C0C F281 4DDE F1A2 6A20 9308 C5E1 4613 \  
8C4F AA7E A3BC 2ACE C483 0F2D B7CB" \  
...
```

### Related references

[5.11 Format of a license entry on page 5-53.](#)



## 1.5 Floating license scheme

A floating license scheme is typically used if there are more potential users than purchased licenses.

A floating license scheme is more flexible than a node locked license but it requires more administration.

A floating license-managed installation of an ARM product consists of the following computers:

### Clients

These are computers that have your ARM development software installed on them. There can be more clients than purchased licenses. If all of the license seats are used, no additional users can use the product until one of the current users stops using the product.

### Server

This is the computer that has the license and floating license server utilities installed on it. A floating license server can also be a client, but this is not the typical mode of operation.

The floating license server controls the number of copies of the licensed product component in use at any one time, up to the maximum number of licenses purchased.

The license extract in the following figure shows part of a typical floating license. In this example the fully qualified SERVER name is `cumulus.mycompany.com`, the host ID is `785f2170` and port `8224` is being used.

This license enables up to ten concurrent users of the specified ARM application, and each user obtains permission to run the application from the floating license server. Because of the flexible way the individual parts of the tools are license managed, this floating license enables up to ten users to use the `armar` feature, while a different ten users use the `fromelf` feature. Some product licenses might restrict concurrent feature use, so check your tools documentation for details.

### Typical single-server floating license extract

```
SERVER cumulus.mycompany.com 785f2170 8224
VENDOR armlmd
USE_SERVER
INCREMENT fromelf armlmd 5.0201105 permanent 10 7E338B2771EA DUP_GROUP=NONE \
ISSUER="ARM Limited" NOTICE="For support see \
http://www.arm.com/support/" SN=ABCDE-98765-ZYXWV SIGN="8674 \
FEE3 5830 D004 2F8D 8C07 E75C 84AA 14EB 37A0 9AD7 4D73 D34A F83A \
EE22 B1F1 3579 54C8 54BA D9D7 5B87 E024 632C B7E0 1DF2 8526 2312 \
27F2 DD78 C020 ABB9" SIGN2="1E17 F524 7987 9CD4 3BC2 FB34 E2FB \
04AA 2C79 4B4C 776D 0FC3 9797 7653 2FBA 044B 2854 608B D8B9 \
6F8A 7B69 3BFA 4AE4 2234 B9A0 0ACA 3E21 1CF8 0F58 54FE"
INCREMENT armar armlmd 5.0201105 permanent 10 8EE2C29A2DB3 DUP_GROUP=UHD \
ISSUER="ARM Limited" NOTICE="For support see \
http://www.arm.com/support/" SN=ABCDE-98765-ZYXWV SIGN="2B77 \
780F 37C5 CAD3 ED05 AECA 0424 0876 6E3D 2EAC 64EB C78E 7015 6E99 \
2119 08ED 4889 AF20 CD05 9E76 EA75 72FE 0154 CC8F CD78 AA01 \
1824 C650 A7B5 0696" SIGN2="20EA 4015 0356 5FF8 2DE5 083A 2C9F \
5645 B14E 220C 00E1 8970 F32E B873 AC7E 182A 20EF 27EA 7BB8 \
B25E 08DB 70D3 AB8B 7847 DB87 B88A 5700 A079 BEB7 E3D2"...
```

Windows and Unix/Linux floating license servers and clients can be mixed in any installation. Client machines must be able to access the server machines over the *Local Area Network* (LAN).

### Note

ARM does not support the use of floating licenses across *Wide Area Networks* (WANs). For performance reasons, you are strongly advised to have your floating license servers and clients on the same LAN.

ARM software licenses support single-server and three-server floating license installations. You can only use the ARM License Manager to set up single-server floating licenses. A multiple server floating license configuration must be set up manually.

### Single-server model

The license file is locked to the host ID of a single server that is used for controlling the licensing process. A single-server floating license model requires that the server machine is running at all times and that the client machines can access this server machine over the local area network.

If the network or single server fails, the licensed tools cannot be used until the server is restarted.

The following figure illustrates the single-server model. On the server, a license file with the corresponding server details is loaded by the floating license server processes. The client must be configured so that it references the floating license server and the port through which the license is issued. The recommended method for doing this is setting the `ARMLMD_LICENSE_FILE` environment variable on the client to `port@server`. For the single-server model example in the figure, the environment variable must be set to `8224@Server_A`.

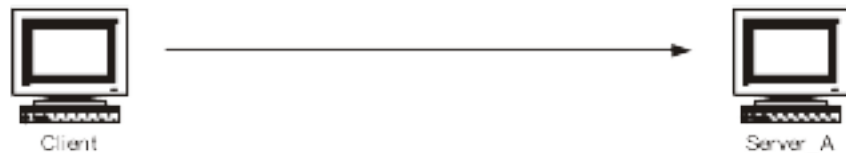


Figure 1-1 Single-server model

### Three-server model

The license file is locked to the host ID of three separate floating license servers. This permits more reliable licensing of the product so that the failure of one server does not stop users from checking out licenses. A three-server floating license model requires that:

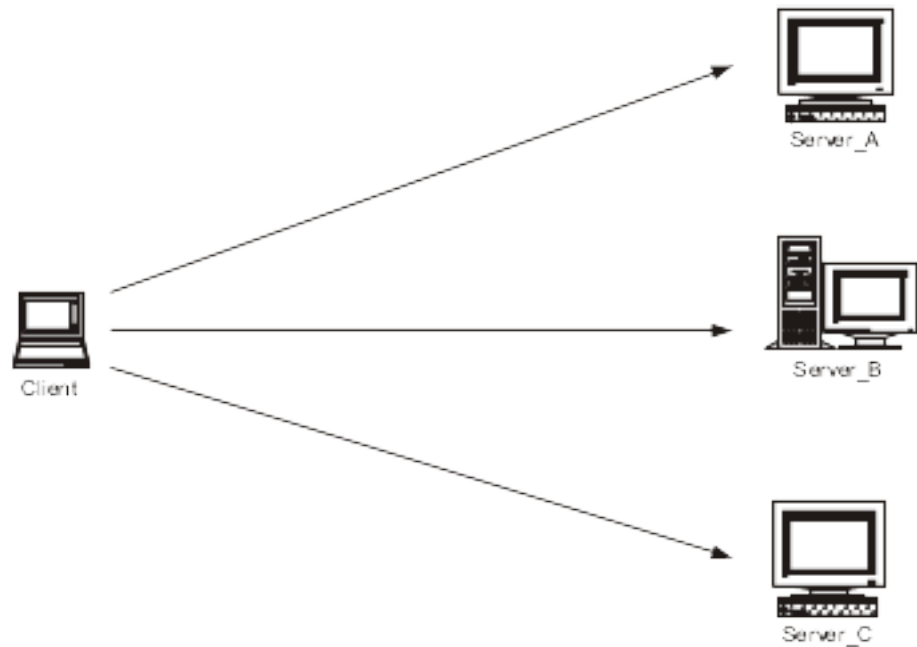
- All three servers must be available when the floating license server starts. One of the servers can be stopped after the initial start up.
- A minimum of two server machines are running at all times.
- All the servers can access each other over the LAN.
- The client machines can also access these server machines over the LAN.

The three-server model has the following limitations:

- It is more complicated to set up than a single-server system.
- Three servers are required.
- Two servers must be operational at all times.

Because of the additional complexity of the three-server model, it is more suitable for managing ten or more total seats, where many users rely on the servers being operational.

The following figure shows the three-server model. On each of the servers, an identical license file with details of all three servers is loaded by the floating license server processes. The client must be configured so that it references all the floating license servers in addition to the ports through which the license is issued. The recommended method for doing this is setting the `ARMLMD_LICENSE_FILE` environment variable.



**Figure 1-2 Three-server model**

For the three-server model example in the figure, on Windows, the environment variable must be set to `8224@Server_A;8224@Server_B;8224@Server_C`. The first server in the list, `Server_A`, is the master server, and must be started before the other two servers.

Depending on your operating system, the separators for the values in the environment variable might be different. On Unix or Linux, the separator is a colon “:” rather than a semicolon “;” as on Windows.

#### **Related references**

[5.11 Format of a license entry on page 5-53.](#)

## 1.6 About the ARM License Manager

The ARM License Manager is a utility for requesting or installing licenses to use DS-5.

You can use the License Manager to:

- Reach the ARM web licensing portal.
- Install node-locked licenses.
- Configure a client to use floating licenses.

### **Related concepts**

[4.1 About configuring the client licensing on page 4-33.](#)

### **Related tasks**

[2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.](#)

[3.1 Installing a node-locked license using the ARM License Manager on page 3-28.](#)

## 1.7 About the ARM web licensing portal

ARM provides a web-based system that you can use to generate and view licenses for your ARM products.

The ARM web licensing portal enables you to obtain a license for your product as soon as you receive a serial number.

Detailed information on how to use the ARM web licensing portal is provided on the main page of the website.

————— **Note** —————

If you are saving a copy of your license file onto your computer, you must save it in plain text format. Using other file formats, such as rich text format (*.rtf*), can result in an unreadable license file and subsequent licensing failure.

---

### **Related tasks**

[8.6 Modifying your floating license file on page 8-84.](#)

### **Related information**

[ARM Self-Service Portal.](#)

## Chapter 2

# Requesting Your License

Describes how to request a license for your ARM software.

It contains the following sections:

- *2.1 Information required for a license on page 2-23.*
- *2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.*

## 2.1 Information required for a license

To apply for a license from ARM, you must have the product serial number and a Host ID.

### Related concepts

[1.7 About the ARM web licensing portal](#) on page 1-21.

### Related tasks

[2.2 Manually obtaining a license from the ARM web licensing portal](#) on page 2-24.

[5.6 How do I move my license to another computer?](#) on page 5-48.

[5.5 How do I find my host ID?](#) on page 5-44.

## 2.2 Manually obtaining a license from the ARM web licensing portal

To obtain a license for an ARM product you can use the ARM License Manager utility to guide you through the required steps.

To manually obtain a license:

### Procedure

1. Start Eclipse for DS-5.
2. Select **Help > ARM License Manager** from the Eclipse for DS-5 menu to display the ARM License Manager dialog box as the figure shows.

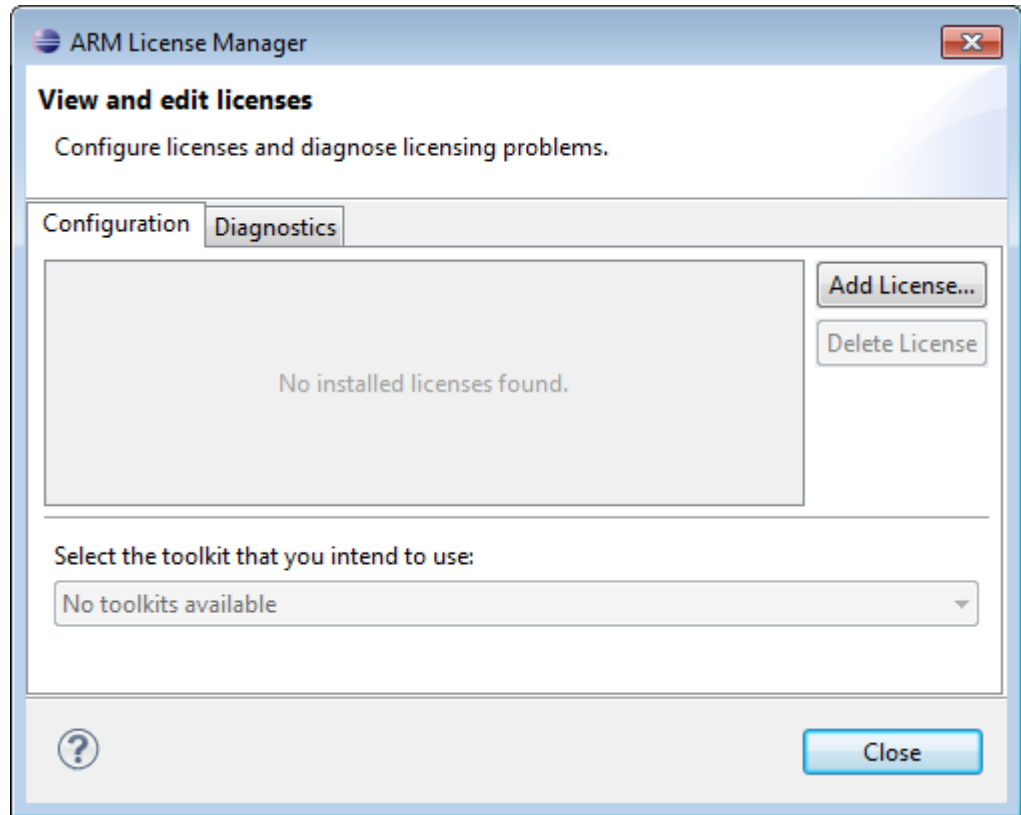


Figure 2-1 View and edit licenses page

3. Click **Add License...** to display the Add License dialog box as the figure shows.



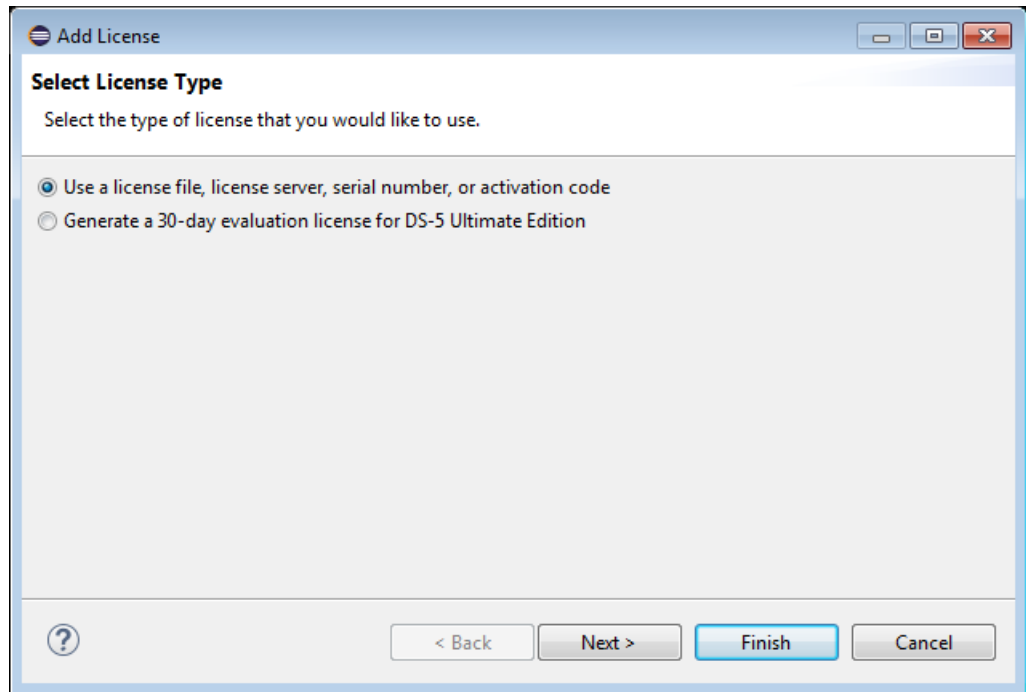


Figure 2-2 Add License dialog box

4. In the **Select License Type** page, select **Use a license file, license server, serial number, or activation code**. Then click **Next**.
5. In the **Obtain a new license** page, select **Manually obtain a license via www.arm.com website (advanced)**. Then click **Next**.

The **Choose network interface** page is displayed, as the figure shows.

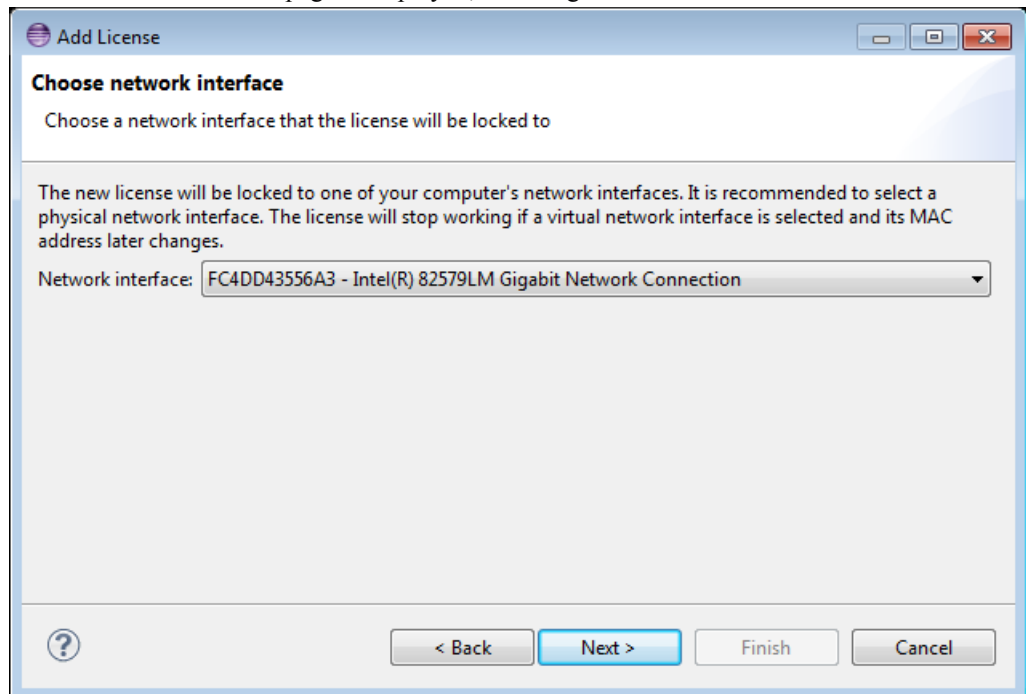
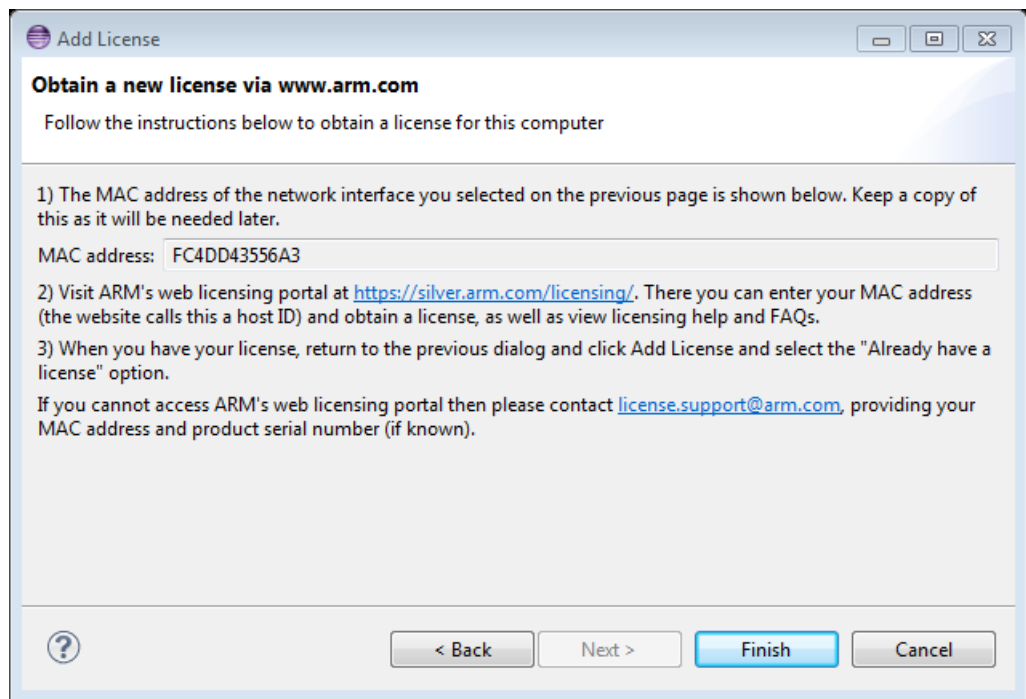


Figure 2-3 Choose network interface

6. Select the Network interface to which you want the license to be locked, then click **Next >**.  
The **Obtain a new license via www.arm.com** page is displayed, as the figure shows.



**Figure 2-4 Obtain a new license via www.arm.com**

7. Click the URL in step 2 of the Obtain License dialog box. This opens your default web browser and takes you to the ARM Login page.
8. Either log into your account, or register for an ARM account.
9. After you have logged into your ARM account, click **Generate** in the Resources, and follow the online instructions.  
See the documentation available on the portal for information on how to use it to obtain a license.
10. Click **Finish** to close this dialog box and return to the ARM License Manager dialog box.
11. Click **Close**.
12. After you have obtained your license from the web licensing portal, save it locally:
  - If you are using a node-locked license, you can install it and configure your licensing environment using the License Manager.
  - Windows floating license server administrators must follow the procedure to install the server license.

### Related concepts

- [1.7 About the ARM web licensing portal on page 1-21.](#)
- [2.1 Information required for a license on page 2-23.](#)

### Related tasks

- [3.1 Installing a node-locked license using the ARM License Manager on page 3-28.](#)
- [8.6 Modifying your floating license file on page 8-84.](#)

## Chapter 3

# Installing a Node-locked License Using the ARM License Manager

Describes how to install a node-locked license for your ARM software product after you have retrieved the license from the ARM web licensing portal.

It contains the following sections:

- [3.1 Installing a node-locked license using the ARM License Manager](#) on page 3-28.
- [3.2 Installing a node-locked license manually \(Windows only\)](#) on page 3-30.
- [3.3 Installing a node-locked license manually \(Unix/Linux only\)](#) on page 3-31.

### 3.1 Installing a node-locked license using the ARM License Manager

The ARM License Manager is a graphical tool that runs in Eclipse. It helps you to set up your ARM licensing environment for DS-5.

To install a node-locked license:

#### Procedure

1. Start Eclipse for DS-5.
2. Select **Help > ARM License Manager** from the Eclipse for DS-5 menu to display the ARM License Manager dialog box.
3. Click **Add License** to display the Add License dialog box as the figure shows.

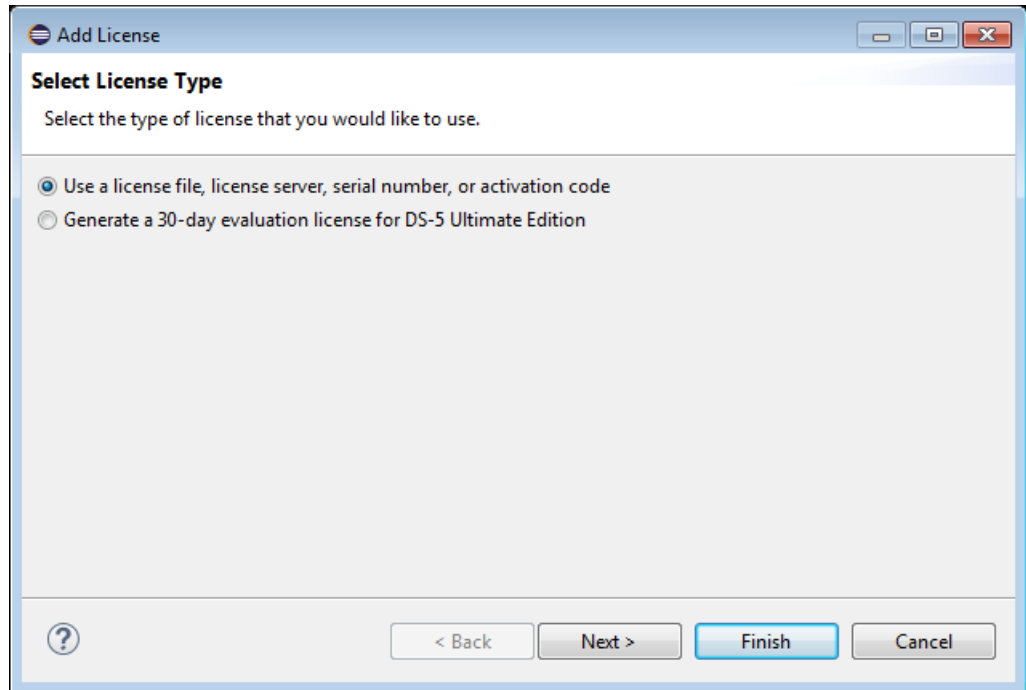


Figure 3-1 Add License dialog box

4. In the **Select License Type** page, select **Use a license file, license server, serial number, or activation code**. Then click **Next**.
5. In the **Obtain a new license** page, select **Use an existing license file or license server address**, then click **Next >**.

The **Enter existing license details** page is displayed, as the figure shows.

**Add License**

**Enter existing license details**

Enter the license details into the form below

License File

File:

License Server

Host:  Port:

**Figure 3-2 Enter existing license details page**

6. Download a license file from the ARM Web Licensing portal and then install the license as follows:
  - a. Click **Browse** to locate and select the license file you downloaded from the ARM Web Licensing portal.
  - b. Click **Open** to select the license file.
  - c. Click **Finish** to add the license file to the list of license sources for your computer.

This makes the requested changes to your ARM licensing environment.
  - d. Click **Close** to close the **License Manager** dialog box.

The license installation is complete.

#### **Related concepts**

[1.4 Node-locked license scheme on page 1-16.](#)

#### **Related tasks**

[2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.](#)

[4.2 Setting up your licensing server environment with the ARM License Manager on page 4-34.](#)

## 3.2 Installing a node-locked license manually (Windows only)

Instead of using the ARM License Manager, you can install your node-locked license manually.

To manually install a node-locked license on a Windows computer:

### Procedure

1. Retrieve your license from the ARM web licensing portal.
2. Save the license file as a `.lic` file (for example, `license.lic`) and place it in an appropriate directory on your computer.
3. Create or modify an environment variable called `ARMLMD_LICENSE_FILE` to point to the location of your license. This location can either be the full path of the license file, or it can be the full path to the directory containing the license file. If only the directory path is specified, the license file must be a `.lic` file (for example, `license.lic`).

————— **Note** —————

The length of `ARMLMD_LICENSE_FILE` must not be more than 260 characters.

For example, to create the variable from the Windows Control Panel:

- Select **Start > Control Panel** to open the Control Panel.
- Double click on the **System** icon in the Control Panel. An easy way to find this icon is to change the Control Panel view to Classic View.
- Click the **Advanced** tab in Windows 2000 or XP, or the **Advanced system settings** task in Windows Vista or Windows 7, then click the **Environment Variables** button.
- Create a new system environment variable, `ARMLMD_LICENSE_FILE`, and set its value to point to the location of your license.

### Related concepts

[1.4 Node-locked license scheme on page 1-16.](#)

### 3.3 Installing a node-locked license manually (Unix/Linux only)

Instead of using the ARM License Manager, you can install your node-locked license manually.

To manually install a node-locked license on a Unix/Linux computer:

#### Procedure

1. Retrieve your license from the ARM web licensing portal.
2. Save the license file as a `.lic` file (for example, `license.lic`) and place it in an appropriate directory on your computer.
3. Create or modify an environment variable called `ARMLMD_LICENSE_FILE` to point to the location of your license. This location can either be the full path of the license file, or it can be the full path to the directory containing the license file. If only the directory path is specified, the license file must be a `.lic` file (for example, `license.lic`).

You can create an environment variable on your Unix/Linux computer using one of the following methods:

- Shell commands.
- `.flexlsrc`.

#### Using shell commands

For example, if you are using `csh` or `tcsh`, you can set the environment variable by entering:

```
setenv ARMLMD_LICENSE_FILE pathname
```

If you are using `bash` or `sh`, enter:

```
ARMLMD_LICENSE_FILE=pathname
```

```
export ARMLMD_LICENSE_FILE
```

where *pathname* is the location of the license file.

See your operating system documentation for more information on setting up environment variables.

#### Using the `.flexlsrc` file

You can edit the `.flexlsrc` file in your home directory to add the environment variable. Add the line:

```
ARMLMD_LICENSE_FILE=pathname
```

where *pathname* is the location of the license file.

#### Related concepts

[1.4 Node-locked license scheme on page 1-16.](#)

# Chapter 4

## Using a Floating License

Describes how to configure your client computer to obtain the license for its ARM tools from a floating license server.

It contains the following sections:

- *4.1 About configuring the client licensing on page 4-33.*
- *4.2 Setting up your licensing server environment with the ARM License Manager on page 4-34.*
- *4.3 Setting up your licensing server environment from the Control Panel (Windows clients only) on page 4-36.*
- *4.4 Setting up your licensing server environment at the command line (Unix/Linux clients only) on page 4-37.*



## 4.1 About configuring the client licensing

The method you use to configure licensing on your computer depends on your platform and server configuration.

You can use one of the following methods to configure licensing on your computer:

- ARM License Manager.
- The Control Panel (Windows clients only).
- The command line (Unix/Linux clients only).

---

**Note**

You can use the license manager to configure licensing only for single-server license configurations. For a three floating license server configuration use the Control Panel or the command line, depending on the platform you are using.

---

### Related concepts

[1.5 Floating license scheme on page 1-17.](#)

## 4.2 Setting up your licensing server environment with the ARM License Manager

Use the ARM License Manager to configure licensing only for single-server license configurations.

To set up your ARM licensing environment for DS-5 using the ARM License Manager:

### Procedure

1. Start Eclipse for DS-5.
2. Select **Help > ARM License Manager** from the Eclipse for DS-5 menu to display the ARM License Manager dialog box.
3. Select **Add License** to display the **Add License** dialog box as the following figure shows:

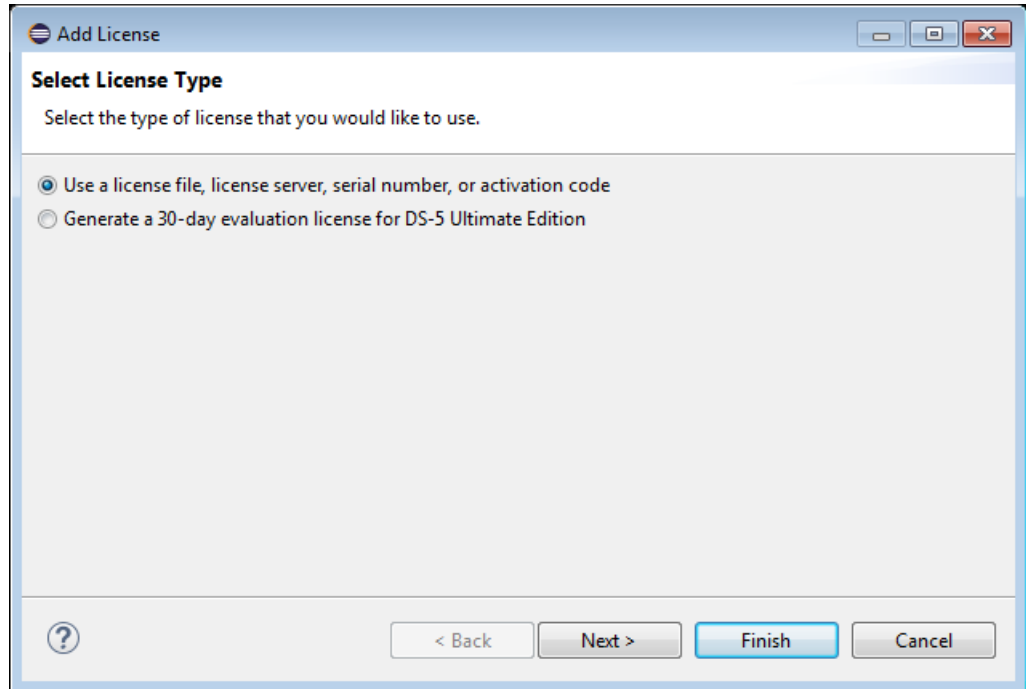
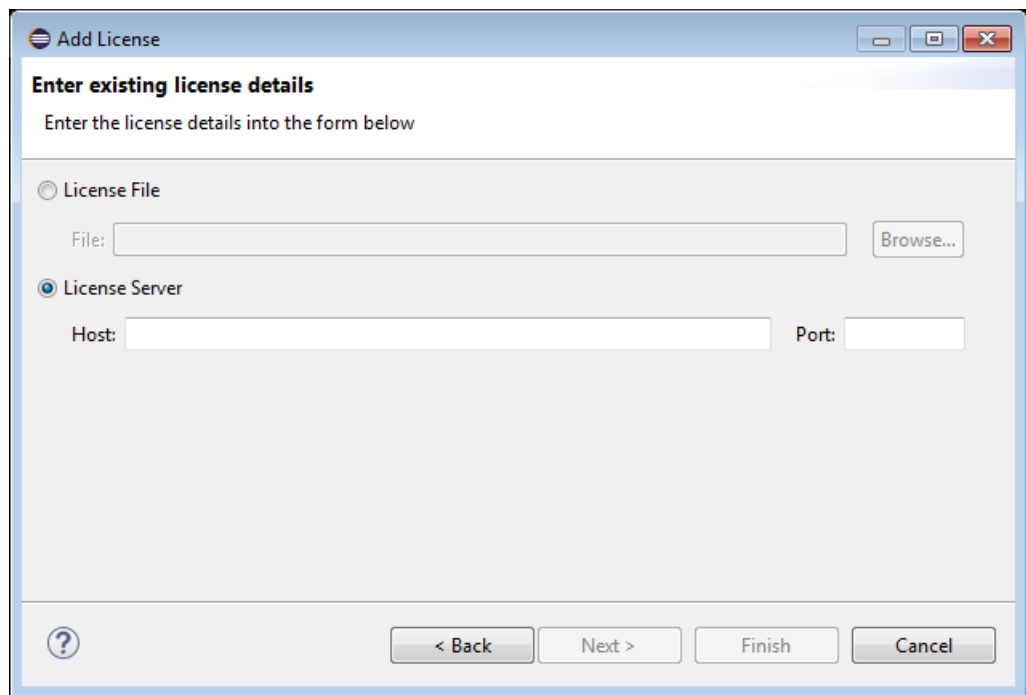


Figure 4-1 Add License dialog box

4. In the **Select License Type** page, select **Use a license file, license server, serial number, or activation code**. Then click **Next**.
5. In the **Obtain a new license** page, select **Use an existing license file or license server address**, then click **Next >**.

The **Enter existing license details** page is displayed, as the figure shows.



The screenshot shows a window titled "Add License" with a subtitle "Enter existing license details". Below the subtitle is the instruction "Enter the license details into the form below". There are two radio button options: "License File" (unselected) and "License Server" (selected). Under "License File", there is a "File:" text box and a "Browse..." button. Under "License Server", there are two text boxes: "Host:" and "Port:". At the bottom of the window, there are four buttons: a help button (question mark in a circle), "< Back", "Next >", "Finish", and "Cancel".

**Figure 4-2 Enter existing license details page**

6. Select **License Server** to specify the floating license server.
7. Enter the floating license server host name in the **Host** field and the port number in the **Port** field.  
If you are using a default server port in the range between 27000 and 27009 inclusive, omit the port number.
8. After you add your license sources, click **Finish** to add the license server to the list of license sources for your computer. The new license server is added to the list of server licenses shown in the **ARM License Manager** dialog box.

### Related tasks

[3.1 Installing a node-locked license using the ARM License Manager on page 3-28.](#)

## 4.3 Setting up your licensing server environment from the Control Panel (Windows clients only)

You can use the Control Panel to set environment variables on supported Windows variants.

Use the Windows Control Panel for a three floating license server configuration.

### Prerequisites

You must have administrator privileges to set system environment variables.

### Procedure

1. Click **Start > Control Panel** to open the Control Panel.
2. Double click on the **System** icon in the Control Panel. An easy way to find this icon is to change the Control Panel view to Classic View.
3. Click the **Advanced** tab in Windows 2000 or XP, or the **Advanced system settings** task in Windows Vista or Windows 7, then click the **Environment Variables** button.
4. Create a new system environment variable, `ARMLMD_LICENSE_FILE`, and set its value. This must include valid `port@server` information.

---

#### Note

The length of `ARMLMD_LICENSE_FILE` must not be more than 260 characters.

---



---

#### Note

If a default port in the range between 27000 and 27009 is used on the server, then the port number can be omitted in the environment variable, but you must include the '@' symbol before the name of the server. For example, `@my_server`.

---

For example:

- If you are using a single-server floating license, set the environment variable value to something similar to:

```
8224@my_server
```

- If you want to refer to more than one floating license server, set the environment variable value to something similar to:

```
8224@my_serverA;8224@my_serverB;@my_serverC
```

The master server appears first, and in this example is `my_serverA`. The entry for `my_serverC` is not preceded by a port number because in this example a default port number is being used.

## 4.4 Setting up your licensing server environment at the command line (Unix/Linux clients only)

On Unix or Linux clients, you can set up your licensing environment using an environment variable.

On Unix or Linux clients, you can set the environment variable `ARMLMD_LICENSE_FILE` to include `port@server` information.

---

### Note

---

If a default port in the range between 27000 and 27009 is used on the server, then you can omit the port number in the environment variable, but you must include the '@' symbol before the name of the server. For example, `@my_server`.

---

You can configure your Unix/Linux client to obtain its license from a server using shell commands, or by editing the `.flexlmrc` file.

### Setting `ARMLMD_LICENSE_FILE` with shell commands

If you are using `csh` or `tcsh`, you can set the environment variable for a single server by entering:

```
setenv ARMLMD_LICENSE_FILE 8224@my_server
```

If you are using `bash` or `sh`, enter:

```
ARMLMD_LICENSE_FILE=8224@my_server
```

```
export ARMLMD_LICENSE_FILE
```

If you want to refer to more than one floating license server, you can add these values to your environment variables. For example, in `csh` or `tcsh` enter:

```
setenv ARMLMD_LICENSE_FILE 8224@my_server1:8224@my_server2:@my_server3
```

In this example, the port number is omitted for `my_server3` because a default port is being used. In all cases you might want to add the environment variable settings to your startup scripts so that your licensing is set up automatically when you log in.

### Setting `ARMLMD_LICENSE_FILE` in the `.flexlmrc` file

You can edit the `.flexlmrc` file in your home directory to set up client licensing by adding the line:

```
ARMLMD_LICENSE_FILE=value
```

where *value* is the value for the environment variable, such as `8224@my_server`.

# Chapter 5

## Frequently Asked Questions about Licensing ARM Products

Describes licensing issues that are not specific to a particular licensing configuration.

It contains the following sections:

- *5.1 How does FlexNet find a license file?* on page 5-39.
- *5.2 Can I use FlexNet with more than one product?* on page 5-40.
- *5.3 How do I merge license files?* on page 5-41.
- *5.4 How do I find my product serial number?* on page 5-43.
- *5.5 How do I find my host ID?* on page 5-44.
- *5.6 How do I move my license to another computer?* on page 5-48.
- *5.7 Do upgrade or update seats replace existing seats?* on page 5-49.
- *5.8 Do the ARM development tools support BORROW?* on page 5-50.
- *5.9 Do ARM licenses support queuing?* on page 5-51.
- *5.10 Why can I not find the lmhostid utility?* on page 5-52.
- *5.11 Format of a license entry* on page 5-53.

## 5.1 How does FlexNet find a license file?

FlexNet uses the environment variable `ARMLMD_LICENSE_FILE` to find the license file.

The locations to be searched by ARM license-managed software are stored in `ARMLMD_LICENSE_FILE`. You can set `ARMLMD_LICENSE_FILE` so that it contains one or more file names, directory names, or `port@server` values. If you specify a directory name, you must not include the final slash /character. The locations are searched in order until a valid license is found. If a directory name is found then each file within the directory that has a `.lic` extension is searched. For ARM licensed tools, the contents of `ARMLMD_LICENSE_FILE` are searched before the contents of the generic FlexNet environment variable, `LM_LICENSE_FILE`.

### ARM license search algorithm

The following example shows an overview of the algorithm used by ARM licensed products when searching for a license file:

#### Search algorithm

```
if (ARMLMD_LICENSE_FILE env-var set)
{
  Search ARMLMD_LICENSE_FILE for feature.
  if (feature found)
  {
    Stop searching
  }
}
if (LM_LICENSE_FILE env-var set)
{
  Search LM_LICENSE_FILE for feature.
  if (feature found)
  {
    Stop searching
  }
}
else
{
  Search c:\FLEXlm\license.dat (Windows)
  Search /usr/local/FLEXlm/licenses/license.dat (Unix/Linux)
  Search ~/.flexlmrc (Unix/Linux)
  if (feature found)
  {
    Stop searching
  }
}
Print error saying that license was not found.
```

## 5.2 Can I use FlexNet with more than one product?

It is possible to use FLEXnet with more than one product.

FlexNet Publisher, formerly known as FLEXnet and FLEXlm, is a widely used product for license management, so it is possible that you have more than one product using FlexNet. If you are using floating licenses, the latest version of the FlexNet floating license server daemon `lmgrd` is backwards compatible with vendor daemons built using previous versions.

————— **Note** —————

If your products are supplied with different versions of FlexNet, you must use the latest version of the FlexNet server daemon, `lmgrd`, for all your products.

See the version of the *FlexNet License Administration Guide* supplied with your tools for further information on using licenses from multiple vendors together.

You might be able to combine multiple license files.

### **Related tasks**

[5.3 How do I merge license files? on page 5-41.](#)



## 5.3 How do I merge license files?

You can merge license files together if you are using more than one licensed application, or are adding a new feature to an existing license.

How you merge your license files depends on the type of license you have.

### Combining node-locked licenses

Node-locked licenses can be combined if the host IDs embedded in each INCREMENT are the same. That is, if they have been generated for the same host ID.

To combine multiple node-locked licenses into a single file, copy and paste all of the INCREMENT lines from your separate license files into a new license file.

### Combining floating licenses

Floating licenses can be combined when both the following conditions exist:

- The number of SERVER lines in each license file is the same.
- The host ID field of each SERVER line in one license file exactly matches the host ID field of each SERVER line in the other license file.

You can merge floating licenses together automatically using the ARM web licensing portal, or manually:

#### Automatic Method

In the ARM web licensing portal:

1. Navigate to: **Licensing > Licenses > Merge > Merge Licenses**.
2. Search for licenses for a particular host ID. All licenses with that particular host ID are returned.
3. Select the licenses you want to merge.
4. Click the **Merge** button.

#### Manual Method

To combine the license files:

1. Open one of the license files with a text editor.
2. Copy the other license file into the text editor and then remove any duplicate SERVER, VENDOR and USE\_SERVER lines.

For example:

```
SERVER this_host <hostid>
VENDOR arm1md
USE_SERVER
```

Always use the newest version of lmgrd and the newest version of each vendor daemon, for example, arm1md, provided with your ARM products.

There are some examples that show this in more detail in the *FlexNet License Administration Guide*. A PDF of this document is available in your DS-5 installation at:

`DS-5_install_directory\documents\license_administration_guide.pdf`.

#### Note

Be aware that you are not entitled to use the mechanism for merging licenses to also merge seats that have since been replaced by upgrade or update seats.

#### Note

There are examples of license file merging in the *FlexNet License Administration Guide*.

**Related information**

*How do I merge my license files.*

## 5.4 How do I find my product serial number?

The serial number can typically be found on a sticker on the product outer box, on a card inside the product packaging, or in the email containing the links to your product downloads.

Your product serial number is normally of the form `xxxxx-xxxxx-xxxxx`, where `x` represents an alphanumeric character.

————— **Note** —————

If you have an evaluation serial number, it might be of the form `LMxxx-xxxx-xxxxxx`, where `x` represents an alphanumeric character. The number of characters immediately after `LM` might be different. Serial numbers for some older ARM product are of the form `xxxxx-xxxxx-xxxxxxxxxxx` or `xxxxx-xxxxx-xxxxx-xxxxx`.

After you have installed your ARM software, you can find the serial number embedded within the license file you installed. Look for a string containing `SN=xxxxx-xxxxx-xxxxx`. The text following the `SN=` label is your serial number.

## 5.5 How do I find my host ID?

The method of obtaining a host ID depends on the type of license, the hardware platform, and the operating system used.

The host ID is an identifier unique to a particular computer. It is used by FlexNet to lock licenses to a specific client or server machine. If an incorrect host ID is provided at the time the license is generated, then the generated license does not enable the product.

The ARM License Manager displays a list of acceptable host IDs. You must select one host ID from the list. The selected host ID is used by the ARM License Manager to lock the license to the client or server machine.

---

**Note**

- In DS-5, the host ID refers to the network interface MAC address.
- Recent releases of DS-5 (v5.24 and above), support the *Consistent Network Device Naming* scheme available in the latest Linux operating systems (For example, Red Hat Enterprise Linux 7 or Ubuntu 16.04 LTS). *Consistent Network Device Naming* scheme is also sometimes known as *Predictable Network Interface Naming* scheme.

In some cases, be aware that `lmutil lmhostid` returns all zeros if the *Consistent Network Device Naming* scheme is implemented. ARM® recommends that you use `ifconfig` for these versions of Linux.

If you are working with older releases of DS-5, then `ethX` is recommended.

---

### Node-locking to network adapter cards

Node-locked licenses are normally locked to the physical address, also known as MAC address, of the computer network card. If you change your network card, you must apply to ARM for a new license.

You can lock your license to a portable network card such as a USB or PCMCIA network card if you require using your license on more than one computer. You can of course only use the software on one computer at a time.

To enable a network card address to be used as the host ID for a node-locked license, your Windows system must have one of the following installed:

- The SNMP service.
- The NETBEUI Transport Protocol.
- The NW Link (IPX/SPX) Transport Protocol.

If the specified conditions are not met, then the network card ID is either not returned, or might be incorrect.

To obtain the computer network card address, type the command `ipconfig /all` at the command prompt and look for the `Physical Address` line for the network card:

```
Description .....: 3Com 10/100 Mini PCI Ethernet Adapter
```

```
Physical Address .....: 00-00-12-42-41-BC
```

The host ID is the `Physical Address` with the “-” symbols removed, so in this example it is `0000124241bc`. Alternatively if you have installed the FlexNet software onto your machine you can instead type `lmutil lmhostid` at a command prompt.

————— **Note** —————

- To use the physical address of the network card as the host ID, you must choose the address of the actual network card, not a virtual network adapter address or a wireless adapter network address.
- You must ensure that the network device that you lock your license to is not shut down by your computer during any power saving operation, that might result in the network device not being available while you are trying to use the license.

### Floating license server host ID

The host ID or host IDs required belong to the floating license server or servers and not to the client machines running the ARM licensed software. If you have already installed the FlexNet server software onto the server, then you can find the host ID using the command `lmutil lmhostid`. If you have not yet installed the FlexNet server software, then you can use the standard operating system commands listed in the following table to obtain the correct host ID:

**Table 5-1 Commands for obtaining host IDs**

OS	Command	Server host ID type	Comments
Windows	<code>ipconfig /all</code>	ETHER NET	The host ID is returned as the 12 character hexadecimal Physical Address. Remove the hyphens, '-', for the host ID.
Linux (Using the ethx scheme)	<code>/sbin/ifconfig eth0</code>	ETHER NET	Returns a 12 character hexadecimal 'HWaddr'. Remove the colons, ':', for the host ID.
Linux (Using the predictable network interface names)	<code>/sbin/ifconfig</code>	ETHER NET	Select the 12 character hexadecimal value in 'ether'. Remove the colons, ':', for the host ID.  For example, from the below output, you can select either <code>70:f3:95:11:05:1a</code> or <code>00:50:b6:05:a2:cc</code> as your host ID:  <pre>[root@mymachine myuser]# ifconfig enp0s25: flags=4163&lt;UP,BROADCAST,RUNNING,MULTICAST&gt; mtu 1500     inet 10.2.200.105 netmask 255.255.255.0 broadcast 10.2.200.255     inet6 fe80::72f3:95ff:fe11:51a prefixlen 64 scopeid 0x20&lt;link&gt;         ether 70:f3:95:11:05:1a txqueuelen 1000 (Ethernet)         RX packets 21513157 bytes 5147802224 (4.7 GiB)         RX errors 0 dropped 0 overruns 0 frame 0         TX packets 29563309 bytes 42816516224 (39.8 GiB)         TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0         device interrupt 16 memory 0xfc500000-fc520000  enp0s29f7u6: flags=4163&lt;UP,BROADCAST,RUNNING,MULTICAST&gt; mtu 1500     inet 10.2.200.123 netmask 255.255.255.0 broadcast 10.2.200.255     inet6 fe80::250:b6ff:fe05:a2cc prefixlen 64 scopeid 0x20&lt;link&gt;         ether 00:50:b6:05:a2:cc txqueuelen 1000 (Ethernet)         RX packets 441396 bytes 29905844 (28.5 MiB)         RX errors 0 dropped 70231 overruns 0 frame 0         TX packets 0 bytes 5303 (5.1 KiB)         TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  lo: flags=73 &lt;UP,LOOPBACK,RUNNING&gt; mtu 65536     inet 127.0.0.1 netmask 255.0.0.0     inet6 ::1 prefixlen 128 scopeid 0x10&lt;host&gt;     loop txqueuelen 0 (Local Loopback)     RX packets 1012511 bytes 40700726584 (37.9 GiB)     RX errors 0 dropped 0 overruns 0 frame 0     TX packets 1012511 bytes 40700726584 (37.9 GiB)     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0</pre>

**Note**

- On Linux, the device used to obtain the server host ID must reside on `eth0`. You can reassign this address to a removable USB network device if you are using a Linux laptop, for example, and want to transfer the license between standalone computers.
- The Server Host ID Type is required when you request your license from the ARM web licensing portal.
- If you are requesting a three server, that is a redundant server, license then you must have a valid and different host ID for each of the three machines.

- You cannot use a volume serial number as the host ID for a Windows license server.
  - ARM no longer supports the HP-UX floating license server.
- 

### **Related concepts**

*1.4 Node-locked license scheme on page 1-16.*

*1.5 Floating license scheme on page 1-17.*

### **Related tasks**

*2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.*

## 5.6 How do I move my license to another computer?

To move your ARM tools license from the original computer for which it was generated to another computer within the same company, you must obtain a new license file from ARM with a new host ID.

A host ID is usually generated from an Ethernet card MAC address.

If you have already obtained your license file from ARM, but you have to move it to another computer within the same company, then you must obtain a new license file with a new Host ID. The move must be within the same legal entity, such as from pc1 at xyz Inc. to pc2 at xyz Inc.

---

**Note**

A reason for moving a license might be a PC hardware failure.

---

### Node Locked Licenses

If your host ID is obtained from your ethernet card, then a new license is required if your ethernet card is replaced.

If your host ID is obtained from your hard disk volume number, then a new license is required if your hard disk is reformatted or replaced.

### Floating Licenses

For Floating licenses, the Host ID is that of the license server or license servers. Therefore, a new license is only required if the Host ID of the license server or license servers change.

### Conversion of Node locked to Floating license

If you want to convert a node locked license to be a floating license then this always incurs a charge, as described in the rehost request pages.

---

**Note**

You are not entitled to attempt to rehost seats that have since been replaced by upgrade or update seats.

---

To rehost your license, you must submit a rehost request through the ARM web licensing portal:

### Procedure

1. Access the [ARM web licensing portal](#).
2. Click the **Rehost** link in the menu on the left of the web page.
3. Accept the Rehost License Request (Disclaimer).
4. Enter either your product serial number or the Host ID of your client machine.

### Related concepts

[2.1 Information required for a license on page 2-23.](#)

[5.7 Do upgrade or update seats replace existing seats? on page 5-49.](#)

### Related tasks

[5.5 How do I find my host ID? on page 5-44.](#)

### Related information

[ARM Self-Service Portal.](#)



## 5.7 Do upgrade or update seats replace existing seats?

If during the update or upgrade process you are provided with a new serial number, then this update replaces the previous entitlement.

You must delete the previous FlexNet license key from all relevant computers and license servers as a condition of receiving the update or upgrade license. This is because your new license is a replacement seat, not an additional seat, and acquisition of the new license automatically terminates the old one. Use of the old FlexNet license key in that situation is not authorized by ARM or its licensors. If you instead require an extra seat rather than simply upgrading or updating your existing FlexNet license key to enable the latest version, then you must purchase a new seat license, not an update or upgrade. Quotations for new seats are available on request from ARM or your local ARM authorized reseller.

————— **Note** —————

ARM FlexNet license keys are backwards compatible. For example, a license for DS-5 5.15 also enables DS-5 5.14.

—————

## 5.8 Do the ARM development tools support BORROW?

ARM license-managed tools do not support the use of the FlexNet BORROW attribute.

If you want to use your software tools away from the network on which you have your floating license server, then you require additional licenses.

If you use your ARM tools on a computer when away from your company network, you are advised to obtain a node-locked license that is locked to the MAC address of a removable network card. You can transfer the card between users if a different computer is used remotely at some other time.

If your ARM software does not support node-locked licenses, you require a new floating license. Although not normally recommended, you can have the server running on the same computer as the tools.

## 5.9 Do ARM licenses support queuing?

ARM floating licenses support license queuing when used with the appropriate version of the FlexNet server software supplied with the product.

If a server receives a request for a license but no seats are currently available, the request is queued. The request is fulfilled when a seat becomes available. The server log reports a DENIED message. However there is not a corresponding license failure on the client side, so for the end user, license queuing is transparent.

————— **Note** —————

License queuing fails if the server is hosting a combination of ARM Developer Suite™ (ADS) licenses and newer ARM tools licenses. If you require the license queuing feature, you must move the ADS licenses to a separate server.

### **Related concepts**

[7.10 Why am I getting "DENIED" messages in the server log? on page 7-73.](#)

### **Related tasks**

[5.6 How do I move my license to another computer? on page 5-48.](#)

## 5.10 Why can I not find the lmhostid utility?

If you are using Unix/Linux, you have probably not run the `makeLinks.sh` script.

The `makeLinks.sh` script creates a series of links to the `lmutil` program, one of which is for `lmhostid`. This script is in the appropriate directory for your platform. See the *Getting Started Guide* for your ARM product for the location of this directory.

If you are using Windows, the `makeLinks.sh` script is not available. Instead:

### Procedure

1. Open a command prompt.
2. Change to the directory where your license utilities are installed,. This is typically `C:\FLEXlm`.
3. Type `lmutil lmhostid` at the command prompt.

## 5.11 Format of a license entry

A license entry has a specific format that specifies the feature, version, and expiry date.

A node-locked or floating license has one of the following formats:

INCREMENT *feature* *armlmd* *version* *expiry\_date* ...

UPGRADE *feature* *armlmd* *version* *expiry\_date* ...

The fields are:

*feature*

The name of the licensed feature.

*version*

The version number of the licensed feature This can have one of the following formats:

*a.b*

*a.byyyymm*

Where:

**a**

major version number of the licensed feature

**b**

minor version number of the licensed feature

**yyyy**

year

**mm**

month number.

For example, 1.0201011 indicates that support for the license feature in version 1.0 expires in November 2010.

Typically used if you have a *Service and Maintenance (S&M)* agreement that limits the period you are allowed to install and use patches.

*expiry\_date*

The date that the license expires for the specified feature, and has one of the following values:

- permanent.
- *dd-mmm-yyyy*, for example 31-May-2012.

This determines the type of license.

The date embedded into the version controls your entitlement to patches, whereas the *expiry\_date* field limits your ability to use the tools.

### Related concepts

[1.4 Node-locked license scheme on page 1-16.](#)

[1.5 Floating license scheme on page 1-17.](#)

# Chapter 6

## Troubleshooting Client Configuration

Describes issues relating to the configuration of clients in either node locked or floating license systems.

It contains the following sections:

- *6.1 License apparently missing error on page 6-55.*
- *6.2 My node-locked tools fail with FlexNet error -8: Invalid license key (inconsistent authentication code) on page 6-56.*
- *6.3 How can I change the order in which my license sources are accessed? on page 6-57.*
- *6.4 Optimizing floating license checkout times on page 6-58.*
- *6.5 Can I use a remote floating license server? on page 6-60.*
- *6.6 Why am I getting FlexNet error -15: Cannot connect to floating license server system? on page 6-61.*

## 6.1 License apparently missing error

When you try to run an ARM licensed product, you might see a message that says that a license cannot be obtained.

### **FlexNet error -1: Cannot find license file**

If you get a FlexNet licensing error stating that the license file cannot be found, it might be because there is not a valid license file on the license search path shown in the error message.

If you are using a node-locked license, use the ARM License Manager to install the permanent license file.

If you are using a floating license, you can manually configure the client to use the permanent license file.

### **FlexNet error -5: No such feature exists**

If you see a license error message stating that no such feature exists, you might be trying to use a component for which there is no corresponding line in the license file. Possible causes are:

- You might have previously used a license for an evaluation version of the ARM tools but have since installed the full version. If you do not reconfigure your client computer to use the full tools license, you are not be able to use the full version, even if the evaluation license has not expired. You must use a license compatible with the type of tools you have installed on your computer. To solve this problem, remove any old or expired licenses and reconfigure your client computer to use your new license.
- You might be trying to use a feature that is available only in the Professional version of the product, but you do not have a license for the Professional version of the product. You must purchase license for the Professional version to use the feature.

### **Related tasks**

[3.1 Installing a node-locked license using the ARM License Manager on page 3-28.](#)

[3.2 Installing a node-locked license manually \(Windows only\) on page 3-30.](#)

[3.3 Installing a node-locked license manually \(Unix/Linux only\) on page 3-31.](#)

### **Related references**

[9.2 armlndiag utility on page 9-96.](#)

## 6.2 My node-locked tools fail with FlexNet error -8: Invalid license key (inconsistent authentication code)

This error can be reported if your license file has been corrupted, moved, or modified in a way that is not allowed.

If this has happened, reinstall a clean version of your license file. For example, You can use the ARM web licensing portal to retrieve your original license.

### Related tasks

[2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.](#)

### Related references

[9.2 armlndiag utility on page 9-96.](#)



## 6.3 How can I change the order in which my license sources are accessed?

You can change the order of the license sources using the `ARMLMD_LICENSE_FILE` environment variable.

If you have more than one license source, you might want to change the order that they are used by the client computer. For example, you might have two separate floating license servers, each with its own distinct license file, and one of these servers has more available licenses for a given product version than the other and you want to reference the server with more seats first.

If you are using a supported version of Windows, you can use the Windows Control panel to modify the `ARMLMD_LICENSE_FILE` environment variable.

If you are using Unix or Linux, you might have to modify the `ARMLMD_LICENSE_FILE` environment variable from the command line. Alternatively you can have the environment variable set automatically when you start a new shell.

---

**Note**

On Windows, the length of `ARMLMD_LICENSE_FILE` must not be more than 260 characters.

---

### Related concepts

[4.1 About configuring the client licensing on page 4-33.](#)

### Related tasks

[4.4 Setting up your licensing server environment at the command line \(Unix/Linux clients only\) on page 4-37.](#)

### Related references

[9.2 armlmdiag utility on page 9-96.](#)

## 6.4 Optimizing floating license checkout times

You can optimize checkout times by using the `ARMLMD_LICENSE_FILE` environment variable to point to your floating license server.

When using a floating license server or servers, each time a client computer starts one of the ARM development tool executables, a license checkout request is sent across the network to the floating license server to obtain a license, and similarly when the license is checked back in. There are a number of ways of configuring a client machine to check out a floating license from a floating license server and it is worth ensuring that your system is configured in such a way that the time taken for these checkout requests is minimized. These are:

- Use the `ARMLMD_LICENSE_FILE` environment variable to point to your floating license server or servers. This environment variable is used first by the ARM license utilities, so it is preferable to use `ARMLMD_LICENSE_FILE` rather than `LM_LICENSE_FILE`, the generic FlexNet environment variable.
- Set `ARMLMD_LICENSE_FILE` to `port@server` rather than pointing it to a local copy of the license file. This removes the delay incurred by the tools having to process the local copy of the license file to find out the name of the floating license server.

————— **Note** —————

You can also use this variable to specify multiple locations. You can use a list separated by semi-colons on Windows, or colons on Unix/Linux. Each location is checked in turn until a valid license is found, so ensure that the fastest or most frequently used license servers are listed first.

- Check for network problems. If everything is working normally, a license is granted in about the same time it takes to ping the server where the license resides.  
If the checkout takes a long time the following suggestions might help:
  - The domain name server might take along time to resolve the server name to an IP address. In this case try substituting the server IP address instead of the name where it appears in the license search paths.
  - Check the physical networking connection between the license server relative to the client computers. License checkouts directly increase in time the further away the license server is. Co-locate the server close to the clients for best performance.
  - Check the number of routers and switches between the license server and the client computers. Each network hop adds some delay to packet processing. The `tracert` utility can provide an indication of the network topology and the delay introduced at each stage. Change your network topology to reduce this.
- Check the load on the license servers. A highly loaded license sever can slow down processing of license requests. In this case, migrate other processes off of your license server computer or move your license server to a faster computer.
- Check for server problems, particularly if you have set up the `ARMLMD_LICENSE_FILE` environment variable to point to multiple servers. You might find it helpful to look through the server log.

If any of the license search paths point to a server which is no longer running the license management software, there will be a delay while FlexNet waits for the license request to time out before moving onto the next location in the search path.

Ensure that there are no spurious entries here that might cause such a delay and that all referenced servers are running.

To temporarily work around this problem, you can modify the order in which the servers appear in the `ARMLMD_LICENSE_FILE` environment variable.

- Set the environment variable TCP\_NODELAY to 1. This sometimes improves FlexNet floating license server system performance when processing license requests. You have to experiment to confirm that setting this option improves performance.

———— **Caution** ————

Setting this environment variable might cause an increase in network traffic.

---

- The license server processes license files in the order top to bottom. Order the license features in this file so that the most frequently used licenses are at the top.

———— **Note** ————

On Windows, the length of ARMLMD\_LICENSE\_FILE must not be more than 260 characters.

---

### **Related tasks**

[6.3 How can I change the order in which my license sources are accessed?](#) on page 6-57.

### **Related references**

[9.2 armlmddiag utility](#) on page 9-96.

### **Related information**

[How can I improve the build time of my application?](#)

## 6.5 Can I use a remote floating license server?

Obtaining a license from a remote floating license server is not a supported mode of operation for ARM development tools. You are strongly discouraged from attempting it.

It is technically possible for a client machine to obtain a license from a remote floating license server across a *Wide Area Network* (WAN). When you set up your licensing system, ensure that client machines have access to local floating license servers across your *Local Area Network* (LAN).

If you do attempt to use a WAN server, the additional network delays connecting to the server can dramatically increase the time taken by the ARM build tools to build your code. In many cases builds can actually fail if the time delays caused are too great. The high license traffic caused can also significantly increase the overall traffic on your WAN, that might lead to more general problems on your network.

## 6.6 Why am I getting FlexNet error -15: Cannot connect to floating license server system?

If you are using ARM software products with a floating license, your workstation must be able to communicate with a server running FlexNet server software. If such communication cannot be established, a commonly reported FlexNet error code on the client is -15.

Possible reasons for FlexNet error code -15 are:

- The wrong license file is being referenced by the application program.
- The floating license server specified in the license file has not been started.
- You are using the wrong `port@host` information.
- The vendor daemon specified in the license file is not running.
- The hostname in the license file is not recognized by the system.
- The network between the client machine and the server machine is down.

To solve these issues, check that you have started your floating license server or servers. You must also check that your client have been correctly configured. The Tool Licensing FAQs on the ARM Technical Support website might also be helpful.

You can try running tests on your server or client computers to identify possible causes of the failure:

1. Try running the `lmutil lmdiag` utility, which is designed primarily for this purpose.
2. Verify that the application is referencing the correct license file.
3. Verify that the vendor daemon, `arm1md`, is running. You can use `ps` on the server to look for the daemon on Unix/Linux, or the Windows Task Manager.
4. Examine the server log file to see if any problems are reported, particularly messages indicating that the vendor daemon has quit.
5. Run `lmutil lmstat -a` on the server machine to verify that the vendor daemon is alive.
6. Run `lmutil lmstat -a` on the client machine to verify the connection from client to vendor daemon across the network.

If none of the above tests identifies the cause of the licensing failure, check whether your client machine can communicate to the server over TCP/IP using a utility such as `ping`. If this fails, then it is possible that communication is being blocked between the server and client.

### Firewalls

Your floating license server and client might be on opposite sides of a firewall. If so, you must configure the firewall to allow access to fixed ports for both the `lmgrd` and `arm1md` license daemons. Define these ports in the server license file by modifying the top of the license file as shown, substituting your own values:

```
SERVER myserver server_hostid 8224
```

```
VENDOR arm1md port=portnumber
```

### Subnets

If your floating license server and client are on different subnets, then using the fully qualified domain name or IP address of the server might solve the problem. Using the IP address normally circumvents issues arising from domain name resolution.

### Server hostname length

There is a character length limit for server hostnames used in the license files. For FLEXlm 8.1b and older, this limit is 32 characters. If you are using newer versions of FlexNet, the limit is 64 characters. If your floating license server name is too long, you must use the IP address of the server instead of the hostname in the license file and client license environment variable.

**Intermittent failures**

You might encounter intermittent licensing failures if your server is under very heavy load, for example, if you use automated build scripts. These failures can be caused by intermittent networking failures. The current versions of the ARM development tools are more resilient with respect to such temporary network interruptions. If you are using older tools, consider adding retry capability to your build scripts to work around the behavior.

**Related references**

[9.2 armlndiag utility on page 9-96.](#)

# Chapter 7

## Troubleshooting Server Configuration

Describes issues relating to the configuration of floating license systems.

It contains the following sections:

- *7.1 How can I run my FlexNet server more securely?* on page 7-64.
- *7.2 Server debug log reports “Invalid license key (inconsistent authentication code)”* on page 7-65.
- *7.3 Can I define the order in which licenses in a mixed version license file are issued?* on page 7-66.
- *7.4 How can I tell how many floating licenses are in use?* on page 7-67.
- *7.5 Server debug log reports “(arlmnd) cannot open lock file” or “cannot communicate with lmgrd”* on page 7-68.
- *7.6 What version of the floating license server daemons must I use?* on page 7-69.
- *7.7 How can I limit user access to particular licensed features?* on page 7-70.
- *7.8 About the options file* on page 7-71.
- *7.9 Configuring your floating license server to use the options file* on page 7-72.
- *7.10 Why am I getting “DENIED” messages in the server log?* on page 7-73.
- *7.11 How can I manage the size of the server debug log file?* on page 7-74.
- *7.12 Suppressing messages from the server log* on page 7-75.
- *7.13 Restarting the floating license server* on page 7-76.
- *7.14 Starting a new server debug log* on page 7-77.

## 7.1 How can I run my FlexNet server more securely?

You can improve the security of your FlexNet server by following some guidelines.

Follow these guidelines to run your FlexNet server more securely:

- Run the `lmgrd` and vendor daemons in a least privileged security level.
- Limit access to administrative users only by running the license server manager in a restrictive mode.
- Do not use the default 27000 TCP port.
- Utilize the recommended security settings offered by the *Operating System* (OS) vendors that resist the buffer/stack overflow attacks.

For more details:

1. Open the [Flexera Software](#) web site in another browser window or tab.
2. Click **Support**.
3. Click **Knowledge Base**.
4. Click **Security Notices**.
5. Do the following:
  - a. Enter **lmgrd** in the Search field.
  - b. Change Product to **Security Notices**.
  - c. Select **KB Articles and Error Documents**.
  - d. Click **Go**.
  - e. Click on the title of the article to view the details.

### Related information

[Flexera Software](#).



## 7.2 Server debug log reports “Invalid license key (inconsistent authentication code)”

A common cause for an invalid license key error message is that you have changed non-user editable parts of your license file.

The only parts of the license you are allowed to edit are the host name and the floating license server daemon port numbers, and the daemon path name if it is set by default to `path_to_armlmd`. You can add certain content, such as a path to an option files or SORT syntax. See the *FlexNet License Administration Guide*.

The invalid license key message can also indicate that you are running an older version of the FlexNet floating license server software that is not compatible with the format of the license file that you have installed. This can happen if you upgrade your development tools to a later version.

See the FAQs in the ARM technical support website, under Tool Licensing, for more information about the versions of floating license server software that are compatible with versions of the ARM software development tools.

### Related references

[9.2 armlmddiag utility on page 9-96.](#)

## 7.3 Can I define the order in which licenses in a mixed version license file are issued?

Whether you can control the order in which licenses are issued by the server is dependent upon the version of the floating license server software you are using.

### FlexNet v11.12.1.0 floating license server software

The FlexNet v11.12.1.0 floating license server software has a defined default order in which license files are sorted when parsed by the floating license server at startup. The default basic sorting rules within a single license file, from highest to lowest priority, are:

1. INCREMENT entries are sorted alphabetically.
2. Uncounted INCREMENT entries are issued before counted INCREMENT entries.
3. Lower versions for an INCREMENT are issued before higher versions.

————— **Note** —————

The FlexNet documentation states that higher versions take precedence over lower versions but Flexera support has confirmed this is incorrect.

4. Licenses with the most recent `START= date`, if specified, are issued before older licenses.

To define the use order of your licenses, add a `SORT=nnn` entry in the required INCREMENT line, where `nnn` is an integer from 000 to 255. The lower the SORT value, the higher the priority. The default sort value is 100. As this value is used when no SORT is defined by the user, lines with values less than 100 appear before unmodified lines, and lines with values greater than 100 appear after unmodified lines.

For example, if you have `compiler` INCREMENT entries in your license file with versions 3.1 and 4.0, you can ensure that the 3.1 license is made available first if you add a SORT attribute to the 3.1 INCREMENT as follows:

```
INCREMENT compiler armlmd 3.1 permanent 5 271940CDEFA0 \SORT=010 DUP_GROUP=UHD ISSUER="ARM  
Ltd" ...
```

### FLEXlm v9.2 or earlier floating license server software

If you are using FLEXlm v9.2 or earlier floating license server software, there is no fixed order in which licenses for different versions of tools are issued. This might mean that older versions of the development tools check out licenses for a newer version, even though the older licenses are available. You might find that an options file allows you to limit which users have access to what versions of a tools license. See [7.7 How can I limit user access to particular licensed features? on page 7-70](#).

### Related tasks

[7.7 How can I limit user access to particular licensed features? on page 7-70](#).

## 7.4 How can I tell how many floating licenses are in use?

You can run the `lmutil lmstat` command, with appropriate arguments, on the server to determine the number of floating license seats that are in use.

The floating license server log is not suitable for the purpose of usage analysis, because at high server loading or with certain types of license file configuration, you might get incomplete or misleading results.

The `lmutil lmstat` command can generate text output that indicates which user is using what version of a license component, and when the license was granted. By selecting the appropriate arguments, such as `-i` for information from the FEATURE or INCREMENT line for the specified feature, or `-a` for all information, you can collect snapshot information of your license usage. You can then run the output through a text parser if you want to analyze the data in more detail. More information on the `lmutil lmstat` command can be found elsewhere. See the *FlexNet License Administration Guide*.

Be aware that there is a trade-off between increased temporal resolution of your usage data and server loading, especially if you have a high turnover licensing environment. The `lmutil lmstat` command can consume a significant fraction of the CPU resource of your server, especially if the `-a` switch is used. At very high loadings, such as greater than 1000 checkouts per minute, the resulting data are known to be inaccurate because not all transactions can be recorded during the data collection period.

## 7.5 Server debug log reports “(armlmd) cannot open lock file” or “cannot communicate with lmgrd”

The lock file error most commonly occurs on a Unix/Linux system if a previous instance of the `armlmd` license daemon was not shut down properly.

Only one copy of `armlmd` can run on your floating license server at any given time. Attempting to start another copy of this daemon causes an error.

When the ARM license daemon, `armlmd`, is started, it creates a file called `/var/tmp/lockarmlmd`. If `lmgrd` terminates abnormally, `armlmd` does not release the lock file because it might still be running. To solve this problem:

1. Locate any license daemon processes. You can do this using the following command on Unix/Linux:

```
ps -a | grep daemon_name
```

where *daemon\_name* is `lmgrd` or `armlmd`.

2. Shut down `lmgrd` first.
3. Next terminate these processes with:

```
kill PID
```

where *PID* is the process ID for the license daemon.

4. Delete the `/var/tmp/lockarmlmd` if it still exists.
5. You can now restart the floating license server.

If you are using Windows, use the Windows Task Manager to locate the `lmgrd` or `armlmd` processes. Shut down `lmgrd` first. Select the process, then click **End Process**.

## 7.6 What version of the floating license server daemons must I use?

ARM development tools use FlexNet (formerly FLEXnet and FLEXlm) license management from Flexera Software. Utilities are provided for running a license server on the platforms supported by the tools.

Floating license server software consists of two parts:

- The ARM floating license server vendor daemon `arm1md`.
- The FlexNet floating license server daemon `lmgrd`.

You can upgrade to later versions of `lmgrd`, but are advised to use the version of `arm1md` provided with the latest ARM software you have installed. The latest floating license server utilities, including `lmgrd` and `arm1md`, can be obtained through ARM, from your tools installation media, download site, or from ARM License Support.

Older releases of the ARM license daemon are forwards compatible with later releases of `lmgrd`. For example, you can use `arm1md` version 9.2 together with `lmgrd` version 11.12.1.0. However it is best practice to use the latest ARM license utilities available.

————— **Note** —————

If your floating license server is running Solaris 5.6, you cannot use FLEXnet version 9.0 utilities. Instead you must use version 9.2 utilities or higher.

The newer utilities, including the `arm1md` vendor daemon, can be requested from ARM if you do not already have a product DVD-ROM or CD-ROM with this version of FlexNet.

—————

The latest ARM license daemon version used by DS-5 is that shown in the directory `DS-5_install_directory\sw\FLEXnet_version`.

## 7.7 How can I limit user access to particular licensed features?

To be able to limit user access to particular licensed features, you must have license administrator privileges.

As a floating license administrator, you can use an options file to:

- Set aside specific features for named users.
- Set aside specific features for named client computers.
- Prevent certain users or computers from being issued with a license of a particular type.

This is useful if you have a license that enables more than one version of the ARM tools and you want to control what users have access to what versions.

To use an options file to limit user access:

### Procedure

1. Create the file.
2. Configure your floating license server to use the file.

### Related concepts

[7.8 About the options file on page 7-71.](#)

### Related tasks

[7.9 Configuring your floating license server to use the options file on page 7-72.](#)

### Related references

[9.2 armlmddiag utility on page 9-96.](#)

## 7.8 About the options file

An option file allows you to specify user access to particular licensed features. You can create an options file in any text editor.

ARM recommends that your options file has the extension `.opt`. When complete, you must copy the options file to each of your ARM floating license servers.

The syntax for the options is as follows:

- Some options, such as `RESERVE`, require that you specify the number of seats.
- The `RESERVE` option puts aside a defined number of seats. You can specify an exact version of the feature if required.
- The `MAX` option puts a limit on the number of seats of a given feature.
- The `EXCLUDE` option prevents a user or group from being granted a license. This overrides any settings to include the user.

For information about other options and the complete syntax to use for all options, see *Managing the Options File* in the *FlexNet License Administration Guide*.

### Example 7-1 Examples

The options file example demonstrates some of the key syntax. In this example, it is assumed that:

- There are two defined project groups with defined users, and a set of build computers.
- The administrator wants to limit the number of seats of a given feature on a group basis, and also wants to reserve ARM Compiler toolchain v5.0 compiler seats for particular users and clients.
- A specific user is not to be granted an ARM Compiler toolchain v5.0 compiler license.

The following permissions and restrictions are required:

- The build computers in the `host_group` `buildbox` must always have two compiler seats available, of no particular version.
- Users in group `project_talisker`, `user1` to `user5`, must not be given more than three fromelf licenses of any version between them.
- Users in group `project_macallan` must always have one seat of compiler version 5.0 available.
- `User7` cannot ever obtain a compiler license of version 5.0, despite being in group `project_macallan`.

The following example contains the corresponding options file that satisfies the restrictions:

```
# License Options File example by ARM Limited
# -----
# This example should be modified as required for your own licensing # environment.
# Turn off case sensitivity for group and hostnames
# (Only valid in FlexNet 10.x. In FLEXlm 9.x, you need to ensure that
# case of groups and hostnames is consistent, and comment out the following
# line.)
#GROUPCASEINSENSITIVE ON
# Groups are defined here
# --> use the form:  GROUP group_name user_list
GROUP project_talisker user1 user 2 user 3 user4 user5
GROUP project_macallan user6 user7 user8
# Host Groups are defined here
# --> use the form:  HOST_GROUP group_name host_list
HOST_GROUP buildbox buildpc01 buildpc02
# Define usage restrictions below
# --> use the form:  OPTION feature[:keyword=value] type {name | group_name}
RESERVE 2 compiler
HOST_GROUP buildboxMAX 3 fromelf
GROUP project_taliskerRESERVE 1 compiler:version=5.0
GROUP project_macallanEXCLUDE compiler:version=5.0
USER user7
```

## 7.9 Configuring your floating license server to use the options file

When you create an options file, you must configure your floating license server to use it.

To configure the floating license server:

### Procedure

1. You must edit your ARM tools license file so that the options file is called on the `VENDOR` or `DAEMON` line. For example, if your options file is called `arm1md.opt`, in your license file you must add `VENDOR arm1md option=arm1md.opt`.
2. If you have multiple servers, repeat the change for the license files on each of your ARM floating license servers.
3. Restart your floating license server or servers.
4. If your configuration is successful, the server log shows messages that the options file is being used and that your restrictions defined in the options file are being implemented.



## 7.10 Why am I getting "DENIED" messages in the server log?

DENIED is not necessarily an error message. It only indicates a real problem if there is a corresponding license failure reported on the client side.

The real failure can occur if a client invokes a tool and it cannot find a valid license file for that tool, and the failure is reported back to the client.

You might be getting a DENIED message if there are no seats available and the request is queued.

However, the message can also occur if you are using a merged license file with multiple versions of the tools. For instance, consider a floating license with both ARM Compiler toolchain v5.0 and ARM Compiler toolchain v4.1 components. If a user requests an ARM Compiler toolchain v5.0 compiler license and the floating license server locates the ARM Compiler toolchain v4.1 compiler license first, there is an immediate version mismatch and the request is denied and recorded in the server debug log. However, the server then continues to look through the available licenses and finds the ARM Compiler toolchain v5.0 license. The server then lets the user check out the license.

In the server debug log, a typical sequence of an initial license failure followed by a successful checkout looks like this:

```
6:35:09 (arm1md) DENIED: "compiler" user@mycomputer (License server does not support this
version of this feature (-25,334))6:35:09 (arm1md) OUT: "compiler" user@mycomputer6:35:09
(arm1md) IN: "compiler" user@mycomputer
```

The initial DENIED is followed immediately by a successful checkout.

If you want to suppress such messages in the server log, you can use the command `NOLOG DENIED` in a server options file. Further information on options files can be found elsewhere. See the *FlexNet License Administration Guide*.

### Related concepts

[5.9 Do ARM licenses support queuing? on page 5-51.](#)

## 7.11 How can I manage the size of the server debug log file?

If you are administering a floating license server that is heavily used, you might find that the floating license server debug log becomes unmanageably large after a relatively short time.

Very large log files in excess of 1GB in size can cause the license utilities to fail. You can control the size of the server debug log file by:

- Suppressing messages.
- Restarting the floating license server.
- Starting a new server debug log.

### **Related tasks**

[7.12 Suppressing messages from the server log on page 7-75.](#)

[7.13 Restarting the floating license server on page 7-76.](#)

[7.14 Starting a new server debug log on page 7-77.](#)

## 7.12 Suppressing messages from the server log

You can stop the server debug log from recording certain message types by using the NOLOG directive in an options file.

You can turn off logging of license checkouts (OUT), checkins (IN), and denied (DENIED) and queued (QUEUED) messages. If you normally suppress server debug log messages, it might be necessary for you to re-enable them to troubleshoot problems.

To turn off logging of checkins use:

To turn off logging of checkouts and queued requests two separate NOLOG lines are required:

```
NOLOG DENIED  
NOLOG QUEUED
```

## 7.13 Restarting the floating license server

When you restart the floating license server, you start a new floating license server debug log file.

Restarting the server can be useful if you suspect a memory leak or similar problem that only becomes noticeable after an extended period. The server shutdown and restart process normally takes a few seconds to complete.

## 7.14 Starting a new server debug log

You can create a new server debug log without restarting the floating license server by using the `lmswitch` command.

Regularly creating new debug logs, for example on a weekly basis, can help you to maintain organized records of floating license server activity. You can use this command on the Windows or Unix/Linux command line. Change to the directory where your floating license server utilities are installed, then type the command:

```
lmutil lmswitch -c license_file_name armlmd new_logfile_name
```

where:

*license\_file\_name*

specifies the fully qualified path name of the license file being used

*new\_logfile\_name*

specifies the fully qualified path name of the new server debug log file.

Your original log file is closed with a message that the server debug log is being switched to *new\_logfile\_name*. The top of *new\_logfile\_name* tells you the name of the previous server debug log file.

# Chapter 8

## Setting Up a Floating License Server

Describes the installation of FlexNet server software and the configuration of the client machines.

It contains the following sections:

- [8.1 About the license management utilities](#) on page 8-79.
- [8.2 About installing the server software](#) on page 8-80.
- [8.3 Getting the ARM floating license server utilities](#) on page 8-81.
- [8.4 Installing server software on Windows](#) on page 8-82.
- [8.5 Installing server software on Unix/Linux](#) on page 8-83.
- [8.6 Modifying your floating license file](#) on page 8-84.
- [8.7 Starting a Windows server from the graphical user interface](#) on page 8-86.
- [8.8 Starting a Windows server from the command line](#) on page 8-89.
- [8.9 Starting a Unix/Linux server from the command line](#) on page 8-90.
- [8.10 About shutting down a floating license server](#) on page 8-91.
- [8.11 Shutting down a Windows server](#) on page 8-92.
- [8.12 Shutting down a Unix/Linux server](#) on page 8-93.

## 8.1 About the license management utilities

The FlexNet licensing utilities provided by ARM are programs that allow you to install and manage ARM licenses.

The utilities include:

`armlmd`

The ARM vendor daemon.

`lmgrd`

The FlexNet server daemon.

`lmutil`

The FlexNet licensing utilities.

The `armlmd`, `lmgrd`, and `lmutil` utilities are available for Windows, Linux, and Mac host platforms.

`lmtools.exe`

The LMTOOLS graphical user interface (GUI) utility. A Windows only GUI tool for FlexNet licensing utilities.

Windows users can set up and manage their server using either the GUI-based `lmtools.exe` utility or the command-line tools.

The licensing utilities are available within a DS-5 installation, usually under `<install folder>\sw\FLEXnet<version number>`.

## 8.2 About installing the server software

Before any floating license can enable the use of your license-managed ARM product, you must install the FlexNet server software on your floating license server or servers, and start it.

————— **Note** —————

If the computer is acting only as a license management server, installation of the ARM software product on the server is not required.

You can download ARM floating license server utilities from the [licensing utilities download page](#).

For any licensing issues, contact ARM license support by email at [license.support@arm.com](mailto:license.support@arm.com).

### **Related tasks**

[8.4 Installing server software on Windows](#) on page 8-82.

[8.5 Installing server software on Unix/Linux](#) on page 8-83.

### **Related references**

[8.1 About the license management utilities](#) on page 8-79.

### **Related information**

[ARM Self-Service Portal](#).



## 8.3 Getting the ARM floating license server utilities

You can download the FlexNet software files from the ARM support site.

### Procedure

1. Browse to the [ARM Self-Service Portal](#)
2. Click **Downloads**.
3. Either register for an ARM account, or log in if you already have an account.
4. Click **Utilities**.
5. Click **FLEXnet binaries and utilities**.
6. Click **Download Now** to get the current version of the FlexNet software files.  
The download automatically starts.
7. If the File Download dialog box is displayed, click **Save** to save the file to your local machine, and follow the on-screen prompts.

## 8.4 Installing server software on Windows

Installing on Windows requires that you copy the FlexNet files and update the PATH environment variable.

### Procedure

To install the server software on a Windows floating license server:

1. Copy the FlexNet files for your ARM application into a directory, for example C:\FlexNet, on the server.
2. Add the directory to the PATH.

## 8.5 Installing server software on Unix/Linux

Installing on Unix/Linux requires that you copy the FlexNet files, update the PATH environment variable, and run a file on the license server.

The floating license server software is normally supported on platforms of the same version as supported by your ARM product. However there are some exceptions.

Alternatively the floating license server utilities are available on the website from where you downloaded your product. If you do not have access to the ARM floating license server utilities, contact ARM License Support.

On any Unix/Linux server, install the server software as follows:

### Procedure

1. Copy the files from the directory for your platform onto the server license machine.
2. Put the destination directory on your PATH environment variable.

————— **Note** —————

Installing the floating license server software on remotely mounted disks is not recommended because of the additional network dependency.

3. On the floating license server machine, change to your license utilities directory, and type:

```
sh ./makelinks.sh
```

### Related concepts

[7.6 What version of the floating license server daemons must I use?](#) on page 7-69.

## 8.6 Modifying your floating license file

Before you can use your license file, you must modify it.

### Prerequisites

You normally retrieve your license yourself from the ARM web licensing portal.

If you are adding this license to a floating license server that is already used for licensing FlexNet managed tools, the same utilities must be used for all tools. You must therefore ensure that all of the floating license server utilities are upgraded to the latest version used by any of the various tools that you have.

---

### Caution

You must not change anything already present in the license file other than the server name and port number, otherwise the license file becomes invalid.

---

### Procedure

1. Move the license file to a known location on the server. A suitable place for the license file is the same directory as your floating license server software.  
The suggested file name is `license.dat`, and this name is used throughout this document to refer to the license file. If you want to call the file something else, substitute your file name for `license.dat`.
2. Open a text editor to modify your `license.dat` file.
3. Replace `this_host` in the license file with the corresponding server name.

---

### Note

If the host ID in the license file is incorrect or if the server host ID has changed, you must apply to ARM for a new license file.

---

4. Add the port number after each host ID. By default, the port number is 8224. If you do not define a port, the operating system chooses one in the range from 27000 to 27009. This number might change if the server is rebooted.
5. Save your edited license file as a plain text file.

---

### Note

If you make a mistake when editing your license file, you can download the original license file again from the ARM web licensing portal.

---

### Example 8-1 Examples

---

For example, a web-generated license file might contain the following line:

```
SERVER this_host 001002A34BC5
```

If you want to use port number 8224 on your floating license server called `synchrotron`, you must change the line to:

```
SERVER synchrotron 001002A34BC5 8224
```

Although you cannot edit your license file except as permitted, you can add certain elements to your license file, such as paths to an options file or vendor daemon. See the *FlexNet License Administration Guide* for details.

---

**Related tasks**

*2.2 Manually obtaining a license from the ARM web licensing portal on page 2-24.*

## 8.7 Starting a Windows server from the graphical user interface

You can start a Windows server from the graphical user interface.

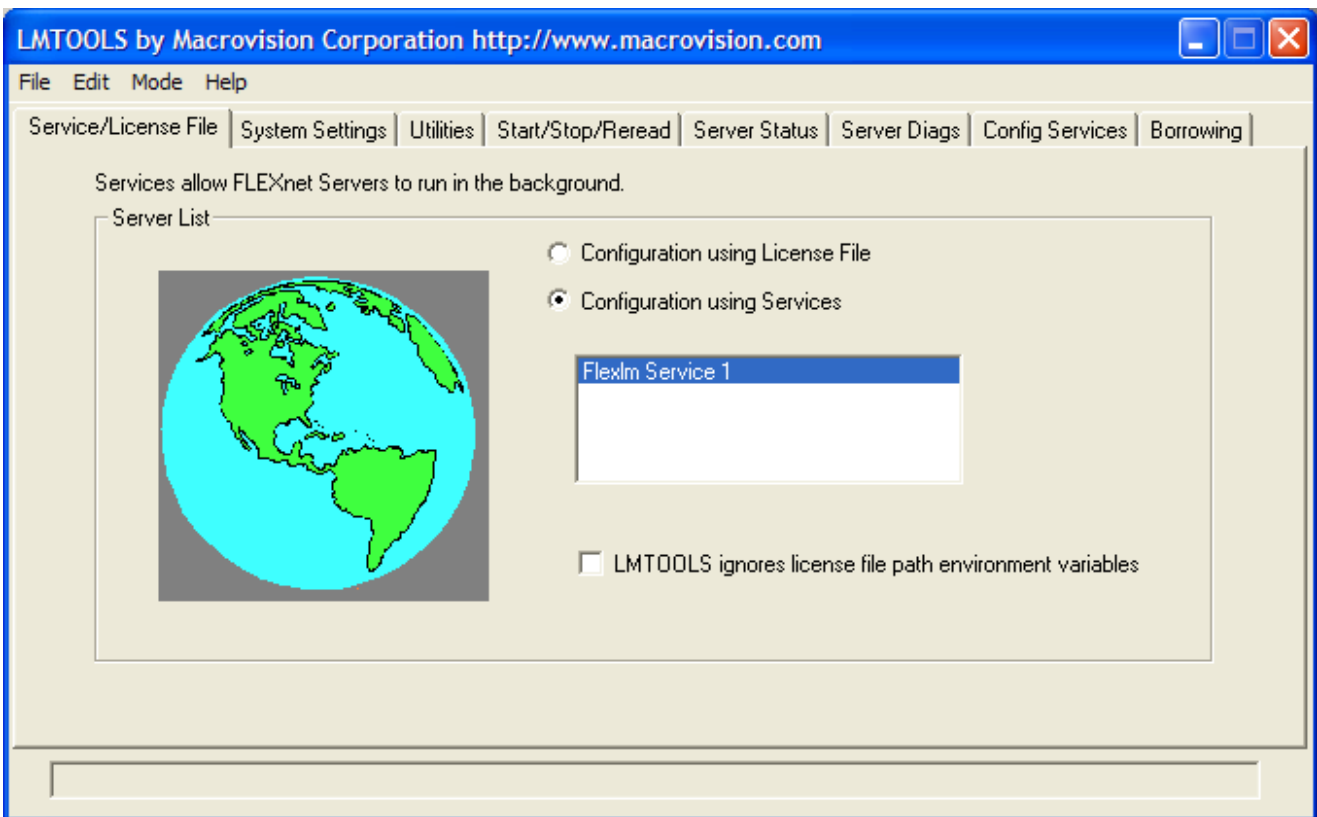
To configure and start the floating license server software on a Windows server using the graphical user interface:

### Prerequisites

If you are adding a new license to a system with a running floating license server, you must stop the server, point the server to the new license file, then restart the server. If this is not done, the new license information might not be recognized by the license daemons. ARM does not recommend the use of the `lmreread` command.

### Procedure

1. Start the `lmtools.exe` program..
2. On the **Server/License File** tab, select the **Configuration using Services** option, as the figure shows.



**Figure 8-1** Selecting Configuration using Services

3. Click the **Config Services** tab to display the dialog as the following figure shows. This example shows the dialog with typical paths entered.

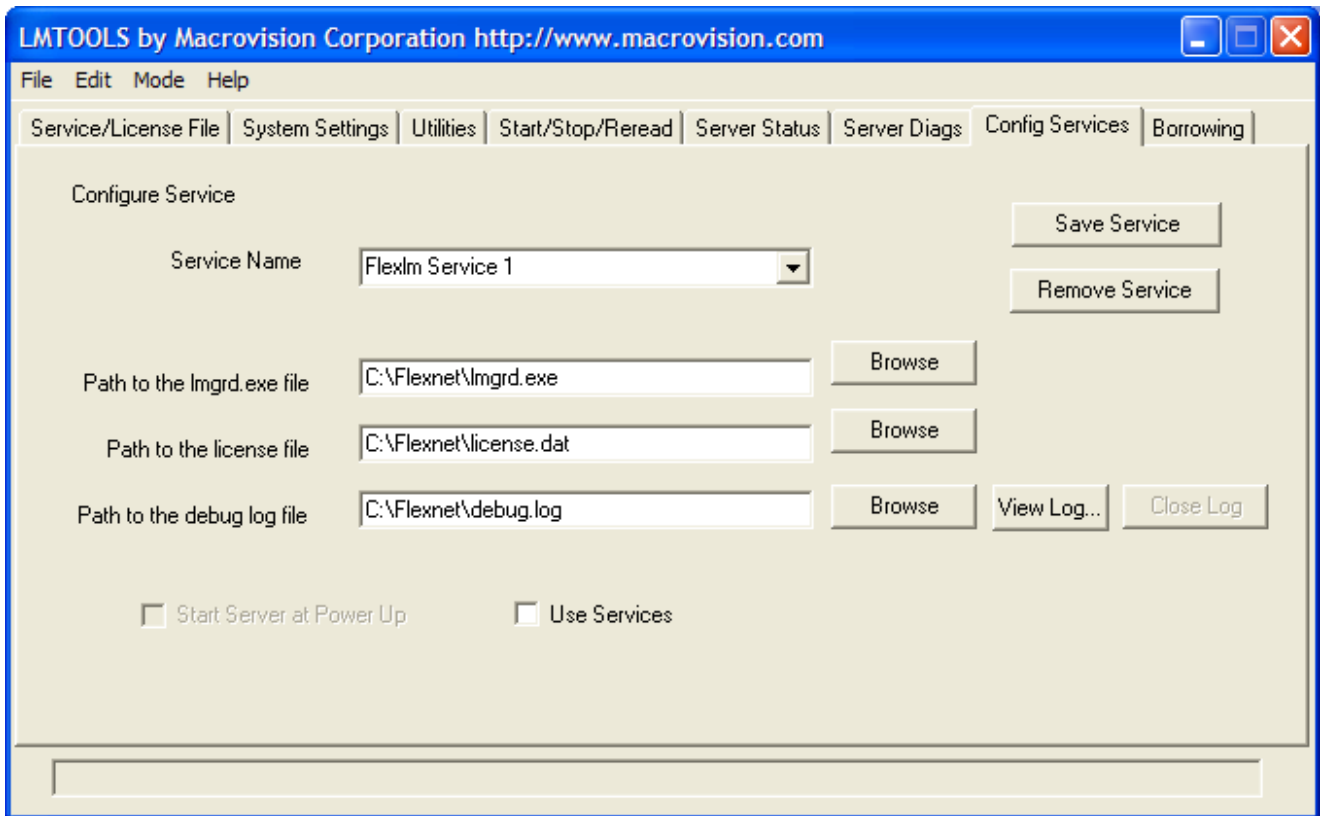


Figure 8-2 Server setup dialog

4. Enter the paths that specify the required files, or click the browse buttons to locate and select the files. You must specify paths for the following:
  - Executable file `lmgrd.exe`.
  - License file `license.dat`.
  - Log file, which is `debug.log` in this example.

————— **Note** —————

The server log file can become very large, so it might be necessary for you to control its size.

5. If you require the server software to start running automatically when the server computer is turned on:
  - a. Select the **Use Services** checkbox.
  - b. Select the **Start Server at Power Up** checkbox.
6. Click **Save Service**.
7. At the prompt, confirm saving the FlexNet License Manager service.
8. Click the **Start/Stop/Reread** tab, and ensure that the FlexNet License Manager service is selected.
9. Click **Start Server** to start running the floating license server software.
10. Check whether the floating license server started successfully:
  - a. Click the **Server Status** tab.
  - b. Click **Perform Status Enquiry**.
  - c. The status information shows that the floating license server and vendor daemon are both UP, followed by a licensed feature usage list. Any floating license server errors are shown in the status information.
11. After you have started the floating license server, you can use a text editor, such as Notepad, to inspect the server debug log file.

**Related concepts**

*8.10 About shutting down a floating license server on page 8-91.*

*8.2 About installing the server software on page 8-80.*

**Related tasks**

*7.11 How can I manage the size of the server debug log file? on page 7-74.*



## 8.8 Starting a Windows server from the command line

You can start a Windows server from the command line.

To start the floating license server on Windows using the command line:

### Prerequisites

If you are adding a new license to a system with a running floating license server, you must stop the server, point the server to the new license file, then restart the server. If this is not done, the new license information might not be recognized by the license daemons. ARM does not recommend the use of the `lmreread` command.

### Procedure

1. Go to the directory where the floating license server software is installed.
2. Enter `lmgrd -c license_file_name -l logfile_name`

where:

*license\_file\_name*

specifies the fully qualified path name of the license file

*logfile\_name*

specifies the fully qualified path name to a server debug log file.

————— **Note** —————

The server log file can become very large, so it might be necessary for you to control its size.

3. After you have started the floating license server, you can use a text editor, such as Notepad, to inspect the server debug log file.

### Related concepts

[8.10 About shutting down a floating license server on page 8-91.](#)

### Related tasks

[7.11 How can I manage the size of the server debug log file? on page 7-74.](#)

## 8.9 Starting a Unix/Linux server from the command line

You can start a Unix/Linux server from the command line.

To start the floating license server software on a Unix/Linux server, go to the directory containing the floating license server software and type:

```
nohup lmgrd -c license_file_name -l logfile_name
```

where:

*license\_file\_name*

Specifies the fully qualified path name of the license file.

*logfile\_name*

Specifies the fully qualified path name to a server debug log file. The debug log file must be on a local storage device to avoid network problems.

————— **Note** —————

The server log file can become very large, so it might be necessary for you to take action to control its size.

————— **Note** —————

`lmgrd` does not require root privileges, ARM recommends that you start the server as a non-privileged user, instead of giving it root privileges.

After you have started the floating license server, enter the following command to see the most recent output from the floating license server:

```
tail -f logfile_name
```

### Prerequisites

If you are adding a new license to a system with a running floating license server, you must stop the server, point the server to the new license file, then restart the server. If this is not done, the new license information might not be recognized by the license daemons. ARM does not recommend the use of the `lmrread` command.

### Related concepts

[8.10 About shutting down a floating license server on page 8-91.](#)

### Related tasks

[7.11 How can I manage the size of the server debug log file? on page 7-74.](#)

## 8.10 About shutting down a floating license server

You might have to shut down a floating license server to, for example, use a new license or to use a modified options file. How you stop a floating license server depends on the platform you are using.

### Related tasks

[8.11 Shutting down a Windows server](#) on page 8-92.

[8.12 Shutting down a Unix/Linux server](#) on page 8-93.

## 8.11 Shutting down a Windows server

Use the graphical user interface to stop a Windows floating license server.

To use the graphical user interface to stop a Windows floating license server:

### Prerequisites

Avoid using the Windows Task Manager to terminate the floating license server daemons because the processes might not end gracefully. This might cause problems on server restart. If you must forcibly shut down a server, always stop `lmgrd` before `arm1md`.

### Procedure

1. Start the `lmttools.exe` program.
2. Click the **Start/Stop/Reread** tab.
3. Click the **Stop Server** button to shut down the floating license server.
4. The server log, if you are using one, records that the server has been shut down.

### Related tasks

[8.4 Installing server software on Windows](#) on page 8-82.

## 8.12 Shutting down a Unix/Linux server

Use the command-line utility to stop a Unix/Linux floating license server.

To stop a Unix/Linux floating license server:

### Prerequisites

Avoid using `kill -9` to terminate the floating license server daemons because the processes might not end gracefully. This might cause problems on server restart. If you must forcibly shut down a server, always stop `lmgrd` before `arm1md`.

### Procedure

1. Change to the directory containing the floating license server software.
2. At the command prompt, enter:

```
lmutil lmdown -q -c license_file_name
```

where:

*license\_file\_name*

Specifies the fully qualified path name of the license file that you used when you started the server.

## Chapter 9

# ARM License Manager Diagnostic Utility

Describes the `armlmddiag` license manager diagnostic utility.

It contains the following sections:

- [9.1 About the `armlmddiag` utility on page 9-95.](#)
- [9.2 `armlmddiag` utility on page 9-96.](#)

## 9.1 About the armlmdiag utility

The ARM license manager diagnostic utility, `armlmdiag`, allows you to perform basic licensing operations and print useful debugging information that can help diagnose licensing problems.

You can use `armlmdiag` to:

- Checkout a floating or a nodelocked license.
- Hold a license for a specified amount of time.
- View the license information after a successful checkout:
  - Expiry date.
  - Number of seats.
  - Version number.
  - Previous version number, if upgraded.

The utility is located at `DS-5_install_directory\sw\FLEXnet_version\platform`, where:

*version*

is the currently supported version of FlexNet.

*platform*

is the name of the platform you are using to run the utility.

## 9.2 armlmdiag utility

The armlmdiag utility has options to perform basic licensing operations and print useful debugging information.

### Syntax

```
armlmdiag --feature name --version N.n [option]
```

Where:

--feature *name*, -f *name*

Is the name of the feature.

--version *N.n*, -v *N.n*

Is the version number of the feature.

*option*

Is any of the following:

--location *Location*, -L *Location*

Location of the license file, in the format port@host or path.

This option is not required if you set the ARMLMD\_LICENSE\_FILE environment variable.

--count *n*, -c *n*

Number of licenses to checkout.

--delay *seconds*, -d *seconds*

Number of seconds to delay the checkout.

--help, -h

Display the command help.

--linger *seconds*, -l *seconds*

The time in seconds that a license is to remain checked out.

--queue, -q

Wait until the license becomes available.

### Example 9-1 Examples

- To check out a license from the server located at 27008@localhost:

```
$ armlmdiag --location 27008@localhost --feature f1 --version 1.0
[14:34:06] ... Checking out f1 (1 licenses).
[14:34:06] ... Checkout done.
Feature Information
-----
Line Type:          INCREMENT
Feature:            f1
Version:            1.0
Expiry date:        permanent
Max number of users: 10
Issuer:             ARM Limited
Notice:             For support see http://www.arm.com/support/
Serial Number:      20987-98444-00000
[14:34:06] ... Exiting, all licenses will be checked in.
```

- To check out five licenses from the server located at 27008@localhost:

```
$ armlmdiag -L 27008@localhost -f f1 -v 1.0 --count 5
...
```

- To check out five licenses from the server located at 27008@localhost and hold them for one minute:

```
$ armlmdiag -L 27008@localhost -f f1 -v 1.0 -c 5 --delay 60
...
```



- To check out a license from the license file /home/user/license.dat:

```
$ armlmdiag -L /home/user/license.dat -f f1 -v 1.0  
...
```

- To set the ARMLMD\_LICENSE\_FILE environment variable to specify the license location, instead of using --location:

— On Linux:

```
$ export ARMLMD_LICENSE_FILE=27010@localhost  
$ armlmdiag -f f1 -v 3.0  
...
```

— On Windows:

```
> set ARMLMD_LICENSE_FILE=27010@localhost  
> armlmdiag -f f1 -v 3.0  
...
```

---

### Related concepts

[5.9 Do ARM licenses support queuing? on page 5-51.](#)