

# **ADOBE FRAMEMAKER 8**

## **INSTALLATION GUIDE FOR UNIX**



© 2007 Adobe Systems Incorporated and its licensors. All rights reserved.

#### Installing Adobe® FrameMaker® 8 for UNIX

If this guide is distributed with software that includes an end-user agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end-user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe, the Adobe logo, Acrobat, Distiller, Flash, FrameMaker, Illustrator, PageMaker, Photoshop, PostScript, Reader, Garamond, Kozuka Mincho, Kozuka Gothic, MinionPro, and MyriadPro are trademarks of Adobe Systems Incorporated.

Microsoft, Windows, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Solaris is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. UNIX is a trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. SVG is a trademark of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions MIT, INRIA, and Keio. All other trademarks are the property of their respective owners.

This product contains either BISAFE and/or TIPEM software by RSA Data Security, Inc.

This product contains color data and/or the Licensed Trademark of The Focoltone Colour System.

PANTONE® Colors displayed in the software application or in the user documentation may not match PANTONE-identified standards. Consult current PANTONE Color Publications for accurate color. PANTONE® and other Pantone, Inc. trademarks are property of Pantone, Inc. © Pantone, Inc. 2003. Pantone, Inc. is the copyright owner of color data and/or software which are licensed to Adobe Systems Incorporated to distribute for use only in combination with Adobe Illustrator. PANTONE Color Data and/or Software shall not be copied onto another disk or into memory unless as part of the execution of Adobe Illustrator software.

Software is produced under Dainippon Ink and Chemicals Inc.'s copyrights of color-data-base derived from Sample Books.

This product contains ImageStream® Graphics and Presentation Filters Copyright ©1991-1996 Inso Corporation and/or Outside In® Viewer Technology ©1992-1996 Inso Corporation. All Rights Reserved.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

Certain Spelling portions of this product is based on Proximity Linguistic Technology. ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. Legal Supplement ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1994 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1993 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Dr. Lluís de Yzaguirre I Maura ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Van Dale Lexicografie bv ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1995 Van Dale Lexicografie bv ©Copyright 1996 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 IDE a.s. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1992 Hachette/Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Text & Satz Datentechnik ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Bertelsmann Lexikon Verlag ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 MorphoLogic Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 William Collins Sons & Co. Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1993-95 Russicon Company Ltd. ©Copyright 1995 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 IDE a.s. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. The Hyphenation portion of this product is based on Proximity Linguistic Technology. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 William Collins Sons & Co. Ltd. ©Copyright 1988 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1997 Van Dale Lexicografie bv ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 Editions Fernand Nathan ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1983 S Fischer Verlag ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 Zanichelli ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc.

Burlington, New Jersey USA. ©Copyright 1989 IDE a.s. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Espasa-Calpe ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 C.A. Stromberg AB. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA.

Portions of Adobe Acrobat include technology used under license from Autonomy, and are copyrighted.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. government end users. The software and documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

# Contents

## **Chapter 1: Getting started**

About multiple-platform networks .....	1
System requirements .....	2

## **Chapter 2: Installing files**

Preparing to install FrameMaker .....	5
Setting up the FrameMaker directory .....	6
Selecting the files to install .....	7
Installing FrameMaker .....	10
Copying FrameMaker files .....	18
Where to go from here .....	18

## **Chapter 3: Setting up the user environment**

Using the fmusersetup script .....	20
Setting a user's environment without fmusersetup .....	22
Using aliases or scripts to avoid modifying paths .....	24
Using FMHOME .....	26

Specifying an XKeysymDB file .....	26
------------------------------------	----

#### **Chapter 4: Setting up licenses**

Licensing basics .....	29
Setting up shared licenses .....	32
Setting up personal licenses .....	36
Using the fmaddlicense program .....	37
Managing a license server .....	42
Starting the license server automatically .....	43
How licensing works .....	44
Generating license reports .....	48
Troubleshooting licensing .....	49
Moving a license server .....	55

#### **Chapter 5: Starting FrameMaker**

Starting FrameMaker .....	57
Running FrameMaker on a remote system .....	59
Using existing licenses files .....	61
Using existing setup files .....	62
Using two versions simultaneously .....	63
Using templates .....	64

Using fonts .....	64
<b>Index</b> .....	<b>67</b>





# Chapter 1: Getting started

This guide describes how a UNIX® system administrator can install and license Adobe® FrameMaker®. A software developer installing the Frame® Developer's Kit™ can use the commands provided in “Installing files” on page 5

*Note: If you currently use an earlier version of FrameMaker, see the Appendix.*

## About multiple-platform networks

You can install UNIX versions of FrameMaker on a single network of workstations that use the same UNIX file server. All users on Sun Microsystems™ can share an installation directory and setup files.

Documents created by FrameMaker are compatible across UNIX and Microsoft® Windows® platforms. For more information, see “*Working on Multiple Platforms*”, on the Adobe website: [www.adobe.com/support/documentation/](http://www.adobe.com/support/documentation/).

## System requirements

To use FrameMaker for UNIX, you need the following hardware and software.

### Platform and software requirements

The following table shows the supported system configuration. See also “Memory requirements” on page 3.

Platform	Workstation model	Operating system	Window environment	Window manager
Sun™	Sun™ SPARCstation®	Solaris 2.8, 2.9, and 2.10	X Window System™ X11R5 or X11R6	dtwm, olwm, or CDE

***Note:** Adobe tests and recommends the listed ICCCM-compliant window managers. If you don't have an ICCCM-compliant window manager, you might be unable to manipulate some FrameMaker windows.*

You also need the following:

- A CD-ROM drive
- For Japanese, Korean, and Chinese characters from the keyboard, a front-end processor software that uses the X Window System Input Manager (XIM) protocol is required.

## Memory requirements

The following table shows requirements for RAM, disk space, and swap space. Disk space requirements are 25 MB larger for Japanese functionality.

Platform	RAM (minimum)	Disk space	Swap space
Sun (Solaris)	512 MB	550 - 820 MB	512 - 1024 MB

## Other requirements for specific configurations

**Sun (KA-Share/ie)** To reliably transfer files using KA-Share on a Sun server with an ie (Intel® Ethernet) interface, you need Ethernet Jumbo Patch 102143-01, available from Sun Microsystems.



# Chapter 2: Installing files

This chapter describes how to install Adobe® FrameMaker® 8 files from a CD-ROM onto a workstation or server. After installing the files, you will set up user environments and licenses.

***Note:** If you have an earlier version of FrameMaker installed, you must install version 8 of the product in a different directory (see “Upgrading from an earlier version” on page 61).*

## Preparing to install FrameMaker

If Adobe® Acrobat® Reader® is not available on your network, you can install it before or after you install FrameMaker.

### Installing Adobe Acrobat Reader

Adobe Acrobat Reader is used to view files in Portable Document Format (PDF), such as the FrameMaker online manual that is saved as a PDF. The Reader installation is available on your FrameMaker CD.

## Setting up the FrameMaker directory

If you are installing for more than one user, you must put the files in a directory for which all users have read and execute permissions. The directory must also be mounted on all users' systems. To install the files, you must have write permission for the directory. Adobe recommends that you install FrameMaker files in a directory named `frame` in the directory where you normally store applications (for example, `/usr/frame` or `/var/frame`).

*Note: Adobe recommends that you make a separate directory for the FrameMaker files. Install only the files for version 8 in this directory.*

### Using hard mounts

The file systems containing your FrameMaker files should be hard-mounted. When you write to a hard-mounted file system, error-checking guards against truncated files. When you write to a soft-mounted file system, your files can be truncated without notification by an error message (for example, if the file system is full). If a FrameMaker process automatically writes to the installation directory, as with license configuration files, the file system containing your FrameMaker installation directory should also be hard-mounted.

### Creating the directory and setting permissions

Do the following to set up the installation directory.

**To set up the FrameMaker installation directory:**

- 1 Use the `mkdir` command to create the directory. For example, enter **`mkdir /usr/frame`**
- 2 Use the `chmod` command to ensure that users can run the products from this directory. For example, enter **`chmod 755 /usr/frame`**

*Note:* The remaining installation instructions refer to the directory you just created as `install_dir`. Type the full name of the new directory when you see `install_dir` in a command.

## Selecting the files to install

The installation script displays a series of menus that list installation options. Your choices from these menus determine which files the script installs into the *install\_dir* directory.

### Performing a basic or custom installation

The installation script gives you the choice of a basic or custom installation. A basic installation provides a quick way to install all the files you need for FrameMaker. If you choose a basic installation, the script installs the following:

- Base files for the product

- Executable files for the platform from which you are installing
- An English-language user interface, or a Japanese-language interface if the LANG environment variable is set to ja
- Type 1 fonts
- Multibyte fonts. For more information, see *Working with Multiple Platforms*, on the Adobe website: [www.adobe.com/support/documentation/](http://www.adobe.com/support/documentation/).
- All filters (for information, see *Using Filters*, on the Adobe website: [www.adobe.com/support/documentation/](http://www.adobe.com/support/documentation/))
- All international dictionary and thesaurus files, to allow viewing and editing of documents created in any supported language
- Demonstration documents and online manuals
- Adobe PDF export filter
- Canadian English Dictionaries (If you choose to install the Canadian English Dictionaries, they will replace the US English Dictionaries.)

If you choose custom installation, you can choose which features to install. For example, you can do the following:

- Install additional user-interface languages (if you have purchased licenses for international editions of FrameMaker).



- Select individual international dictionary and Thesaurus file sets to install, instead of installing all files. (This saves disk space, but may affect the formatting of text in the languages for which you have not installed dictionaries.)

## Installing fonts

If your previous version of FrameMaker installed Linotype fonts (Times, Helvetica, Helvetica Narrow, NewCenturySchlbk, and Palatino), the fonts can be copied into FrameMaker 8 installation. You can purchase additional fonts and add them to FrameMaker. For information, see *Working with Fonts in FrameMaker for UNIX*, available in this path: \$FMHOME/fminit/<UILanguage>/Documents/Unix\_Fonts.pdf.

## Installing sample documents and online manuals

A basic installation includes sample documents and online manuals. A custom installation lets you choose whether to install these files. You might choose to omit these files to save disk space; however, Adobe strongly recommends that you install the online manuals. These provide valuable information that is not available in the printed manuals.

## Choosing the user-interface language for international products

A custom installation also lets you choose which user-interface languages to install. The installation script installs the user-interface, dictionary, thesaurus, and demonstration files in the languages you choose, and optionally installs online manuals in those languages (some online manuals are only available in English). For information about changing the default user-interface language after installation. For information, see the online manual *Customizing FrameMaker* on the Adobe website: [www.adobe.com/devnet/frame-maker/pdfs/Customizing\\_Frame\\_Products.pdf](http://www.adobe.com/devnet/frame-maker/pdfs/Customizing_Frame_Products.pdf).

## FrameMaker modes

FrameMaker is capable of launching and running in different modes. If you want to launch in the FrameMaker mode, select Maker. If you want to launch in Structured FrameMaker mode, select Structured. If you want each user to determine which mode to run in, select User Choice.

# Installing FrameMaker

These instructions describe how to do the following:

- Check whether the CD-ROM drive is mounted

- Mount the CD-ROM drive, if necessary
- Run the CD-ROM installation script to install FrameMaker files
- Troubleshoot the installation

You can install from a CD-ROM drive on your local system or from a remote system. If the CD-ROM drive isn't mounted, you must log in as the root user to mount it. If you need to log in as root to mount the drive, log out after mounting it and log in under a different name before installing the files. Installing under your non-root user ID or under a special system administration name makes it easier to safely administer the installation.

### **Checking for a CD-ROM directory**

Normally, the CD-ROM is mounted to the /cdrom or /CDROM directory. Load the disc into the drive and check for this directory on your system.

If you don't find a CD-ROM directory, create one and then skip to “Mounting the CD-ROM drive on a local system” on page 12 or “Mounting the CD-ROM drive from a remote system” on page 14. If you do find a CD-ROM directory, check whether the drive is mounted as described in the following section.

### **Checking whether the CD-ROM drive is mounted**

If the CD-ROM drive is not mounted, you will have to mount it before you can install the files.

**To determine if the CD-ROM drive has already been mounted:**

Enter the following command:

```
/sbin/mount
```

You should see a list of mounted file systems. If the CD-ROM drive has already been mounted on your local system, a line similar to the following line appears in the list (where the value of *device\_name* depends on your system):

```
/cdrom/adobe_framemaker#1 on /vol/dev/dsk/c0t2d0/adobe_framemaker#1
```

- If the CD-ROM drive has been mounted from a remote system, a line similar to the following one appears in the list of mounted file systems (where *host* is the host name of the remote system):

```
host: /cdrom on /cdrom
```

If the drive has not been mounted, please read the following section, “Mounting the CD-ROM drive from a remote system” on page 14. If the drive has been mounted, skip to “Running the installation script” on page 16.

**Mounting the CD-ROM drive on a local system**

To mount a CD-ROM drive, you must know your system’s root password.

**To mount the drive on a local system:**

**1** Log in as the root user and insert the FrameMaker CD into the CD drive. Otherwise, the mount command may not work.

**2** Create the CD-ROM directory by entering

**3** Mount the drive by entering the following command:

The CD drive may be automatically mounted for you when you insert the CD. If it is not, type **`/sbin/mount -F hsfs -r /dev/sr0 /cdrom`**

The value of *device\_name* depends on your system. The -r option mounts the drive as read-only. Use this option to prevent media error messages from appearing.

**4** Verify that the drive was mounted correctly by entering the following command:

You should see a list of mounted file systems, including the name of the drive you just mounted.

**5** To exit root, enter **exit**

Now, skip to “Running the installation script” on page 16.

## Mounting the CD-ROM drive from a remote system

If the CD-ROM drive is attached to a remote system, check whether it is listed in the exports file on the remote system, and add it if necessary. The exports file specifies directories that can be exported to local systems. Its pathname and filename depend on your platform.

After checking, correcting, and loading the exports file, you can mount the CD-ROM drive.

### To check the exports file:

- 1 Log in to the remote system as the root user.
- 2 Change to the appropriate directory by entering the following command:

```
cd /etc/dfs
```

- 3 List the exports file by entering the following command:

```
ls dfstab
```

If you do not see the file listed, create the file and edit it as described in the following procedure.

- 4 Check the contents of the exports file for the following line:

```
share -F nfs -o ro /cdrom
```

If you see this line, skip to the procedure, “To mount the CD-ROM drive from a remote system:” on page 15. If you do not see this line, you need to add it to the file as described in the following procedure.

**To edit the exports file:**

- 1 If the file does not exist, use a text editor to create it.
- 2 Using a text editor, add the following line as the last line of the exports file:

```
share -F nfs -o ro /cdrom
```

After editing the exports file, you must load it.

**To load the exports file:**

The CD drive may be automatically mounted for you when you insert the CD. If it is not, type `/sbin/mount -F hsfs -r /dev/sr0 /cdrom`. Load the file by entering `shareall`

Now you can mount the CD-ROM drive.

***Note:** Your remote machine must be running NFS server daemon processes to permit other systems to access exported disk partitions.*

**To mount the CD-ROM drive from a remote system:**

- 1 Insert the FrameMaker CD into the CD drive and on the remote system, enter the following command:

```
/sbin/mount -F hfs -r /dev/sr0 /cdrom
```

The value of *device\_name* depends on your system. The -r option mounts the drive as read-only. Use this option to prevent media error messages from appearing.

2 Exit root on the remote system by entering **exit**

3 Log in as the root user on your local system. The system will request the root password.

4 On the local system, enter the following command (where *host* is the name of the remote system):

```
/sbin/mount host:/cdrom /cdrom
```

5 Exit root by entering **exit**

## Running the installation script

***Important:** You must run the installation script directly from the CD-ROM. Don't install the current version of FrameMaker in the same directory as earlier versions.*

### To run the installation script:

1 Change to the CD-ROM directory by entering the following command:

```
cd /cdrom/adobe_framemaker#1
```



2 Enter the following CD-ROM installation command:

```
./read.cd
```

3 Read the instructions provided by the CD-ROM installation script and answer the questions. The script asks you to specify the FrameMaker installation directory and which files to install. It then installs the appropriate files into *install\_dir*.

## Checking the installation

After the installation script copies the FrameMaker files to the product directory, you can use the `fmcheckinstall` script to check the installation.

### To check the installation:

- 1 Change to the installation directory (`install_dir`)
- 2 Run the `fmcheckinstall` script by entering `./bin/fmcheckinstall`

The script verifies the files you installed and reports missing or modified files. Some files will be modified according to options specified during installation.

## Copying FrameMaker files

If you need to copy the FrameMaker directory and its subdirectories from one location to another, use the `fmcopy` shell script located in `install_dir/bin`. To see instructions for using this script, change to `install_dir/bin` and enter `./fmcopy` in a UNIX window.

**Important:** *Do not use the UNIX `cp -r` command; only `fmcopy` will copy the correct file permissions and symbolic links.*

## Where to go from here

To complete the installation, set up each user's environment for FrameMaker, see , “Setting up the user environment.”

If you are installing a licensed version of FrameMaker, you must then set up licenses. See , “Setting up licenses.”

# Chapter 3: Setting up the user environment

You should change the PATH environment variable for each user so they can start Adobe® FrameMaker® from any directory. The easiest way to do this is to run the provided `fmusersetup` script. You also can manually change each user's path without using `fmusersetup`. If you choose not to change paths, users must supply full pathnames or change to the installation directory to start FrameMaker. To make this easier for users whose paths are not changed, you can create aliases or scripts.

This chapter describes how to run `fmusersetup` and explains the changes it makes. The chapter also describes the alternative methods, and explains how to set an environment variable that refers to a provided XKeysymDB file.

**Note:** Adobe also provides special setup files and programs to integrate FrameMaker with several desktop environments. Desktop integration lets you use standard procedures—for example, double-clicking a document icon—to start FrameMaker. For more information, see the list of README files for UNIX platforms in the online Release Notes.

## Using the `fmusersetup` script

The `fmusersetup` script alters the user's startup files to define the environment variable `FMHOME` (which specifies the directory in which you installed FrameMaker products). It also adds the FrameMaker bin directory to the `PATH` environment variable, to make it easier to launch FrameMaker, and adds execute permission to the user's home directory (for printing).

### Running `fmusersetup`

You can run the `fmusersetup` script for each user, or instruct users to run it themselves.

#### To set up a user's environment by using `fmusersetup`:

- 1 Change to the installation directory (*`install_dir`*). For example, if FrameMaker is installed in `/usr/frame`, enter
- 2 Run the `fmusersetup` script by entering
- 3 Answer the questions the script asks. The `fmusersetup` script changes the user's path and defines the `FMHOME` environment variable.
- 4 Log out and log in again, to put the changes into effect.

5 Test the script by starting FrameMaker from any directory. (For information, see “Starting FrameMaker” on page 57)

## Running `fmusersetup` as root

If you choose to run `fmusersetup` as root, make sure the `HOME` environment variable is set to the root directory of the machine on which FrameMaker is installed.

### To run `fmusersetup` as root:

- 1 Do one of the following:
  - If you use the C shell, enter `setenv HOME /`
  - If you use the Bourne shell or Korn shell, enter:  
`HOME=/; export HOME`
- 2 Run `fmusersetup` as described in the preceding section.

## Making corrections for absolute paths

The `fmusersetup` script changes a user’s path by editing the user’s `.login`, `.cshrc`, or `.profile` file. It assumes the path statement is defined by the `PATH` environment variable, and it redefines this variable.

If the path assignment statement is defined absolutely, you must edit the file after `fmusersetup` changes it. (The path is defined absolutely if the path statement doesn't have the environment variable `PATH` on the right side of the equal sign. It has pathnames instead.) To correct the file, you must move the lines that `fmusersetup` added so that they occur before the path assignment statement.

### To correct for an absolute path assignment:

Do one of the following (where *install\_dir* is the name of the directory in which you installed FrameMaker):

- If you use the C shell, make sure the set path line follows the `setenv FMHOME` line. For example:  
`setenv FMHOME install_dir`  
`set path=($FMHOME/bin $path)`
- If you use the Bourne or Korn shell, make sure the `PATH` line follows the `FMHOME` line. For example:  
`FMHOME=install_dir; export FMHOME`  
`PATH=$FMHOME/bin:$PATH; export PATH`

## Setting a user's environment without `fmusersetup`

You can set `FMHOME` to the FrameMaker installation directory in a user's `.login` or `.profile` file and then use `$FMHOME` to represent that directory when you change the user's path.

**To set up a user's environment without running fmusersetup:**

**1** Define FMHOME by doing one of the following (where *install\_dir* is the name of the directory in which you installed FrameMaker):

- If you use the C shell, add this line to the .login file:

*install\_dir*

- If you use the Bourne or Korn shell, add this line to the .profile file:

*install\_dir*

**2** Modify the user's path by doing one of the following (where *install\_dir* is the name of the directory in which you installed FrameMaker):

- If you use the C shell, add this line to the .login file:
  
- If you use the Bourne or Korn shell, add this line to the .profile file:

**3** Instruct the user to log out and log in again, to put the changes into effect.

## Using aliases or scripts to avoid modifying paths

If you don't want to modify paths, you can use aliases or scripts to make it easier for users to start FrameMaker. To do this, create an alias or a shell script that refers to the full pathname for the command to start a product. For information on the commands, see , “Starting FrameMaker.”

*Important:* Do not rename the actual product files.

### Creating an alias

If your UNIX shell supports aliases, you can use an alias to provide a shorter command that represents the full pathname for FrameMaker.

#### To create an alias that starts FrameMaker:

Do one of the following (where the examples create a new command, `framemaker`, for starting the US English version of FrameMaker, and where *install\_dir* is the name of the directory in which you installed FrameMaker):

- If you use the C shell, define the alias in the `.login` file. For example:

*install\_dir*



- If you use the Bourne shell, define the alias in the .profile file. For example:

```
{  
install_dir/bin/maker  
}
```

- If you use the Korn shell, define the alias in the .profile file. For example:  
**alias framemaker='install\_dir/bin/maker'**

## Creating a script

You could also place a script that starts FrameMaker in a directory that is in the user's path. Then the user can start the product by entering the name of the script file.

### To create a script that starts FrameMaker:

Place a script such as the following, with a filename that specifies the product, in a directory that is in the user's path. (This example starts the U.S. English version of FrameMaker; *install\_dir* is the name of the directory in which you installed FrameMaker.)

```
#!/bin/sh
case $# in
0) exec install_dir/bin/maker;;
*) exec install_dir/bin/maker "$@" ;;
esac
```

## Using FMHOME

The remaining instructions in this guide assume that you've defined FMHOME to represent the FrameMaker installation directory (*install\_dir*). If you haven't defined this variable, enter the installation directory's full pathname whenever you see \$FMHOME in a command.

## Specifying an XKeysymDB file

If you are using an older version of the XKeysymDB file, you may see messages such as the following when you start FrameMaker:

Subject: Warning: translation table syntax error: Unknown keysym name: osfPageLeft

You can correct this problem by setting the XKEYSYMDB environment variable to refer to an updated version of the XKeysymDB file, which is provided in \$FMHOME/fmunit.

**To use an updated XKeysymDB file:**

Set the XKEYSYMDB environment variable by doing one of the following:

- If you use the C shell, enter:  
**setenv XKEYSYMDB \$FMHOME/fmunit/XKeysymDB**
- If you use the Bourne or Korn shell, enter:  
**XKEYSYMDB=\$FMHOME/fmunit/XKeysymDB ; export XKEYSYMDB**

Instead of using an environment variable, you can copy the provided XKeysymDB file to overwrite your existing XKeysymDB file (this is usually located in /usr/lib/X11 or in /usr/openwin/lib/X11).



# Chapter 4: Setting up licenses

This chapter describes how to set up licenses, get reports on license usage, and troubleshoot licensing problems.

*Note: Please skip this chapter if you are installing a demonstration version of FrameMaker.*

## Licensing basics

When you start Adobe® FrameMaker®, the product searches for a license that allows you to run it. Without a license, you can run only a demonstration version. When you purchase FrameMaker, your Adobe sales representative helps you determine the appropriate number and type of licenses for your site.

### License types

FrameMaker supports shared and personal license types so you can choose the one that is most appropriate for your site. You can also add licenses at any time.

**Personal licenses** A personal license allows a single user to use FrameMaker. When you set up licensing, you must assign a user name to each personal license. A licensed user can use FrameMaker from any system on your network, but can only start the product on one system at a time.

**Shared licenses** Shared licenses allow multiple users to use FrameMaker. Each shared license allows a certain number of users simultaneously. As long as the number of users is not greater than the license limit, users automatically get licenses when they start the product.

### **Network requirements for shared licenses**

FrameMaker supports shared licenses in a local area network (LAN) environment. Licenses cannot be shared across a wide area network (WAN). FrameMaker distinguishes between a LAN and a WAN by requiring that all machines on a LAN reside within the same building. In addition, to support shared licensing, a LAN must maintain a reasonable level of performance at all times. As a guideline, if the LAN is used for the Sun Network File System (NFS), it is likely to meet the requirement for reasonable performance. If your network does not meet this definition of a LAN, you must use personal licenses.

### **Registration numbers**

Your FrameMaker package contains a registration card or a separate sheet that includes the registration numbers for licenses you have purchased. You use this information to set up licensing.

***Note:** Make copies of your licensing information and store them in a safe place. If your licensing files become damaged, you must create them again from the original information.*

## **Licenses file**

You set up licenses by using FrameMaker to record license registration numbers in the licenses file.

If you are setting up shared licenses, create the licenses file on a system that all users can use, for example, a file server. You can set up a single licenses file in \$FMHOME/fmunit, or you can create separate licenses files for work groups. See “Setting up workgroups for shared licenses” on page 32.

If you are setting up personal licenses, you can create a personal licenses file in \$FMHOME/fmunit, or you can create individual licenses files in an fmunit subdirectory of each user’s home directory (~/.fmunit). A central licenses file is easier to manage if you have many users. If you have just a few users, you might want to have them manage their own personal licenses files.

Different versions of FrameMaker (version 8 and earlier) may share the same licensing files.

## Setting up shared licenses

If you are installing shared licenses, you must designate a system as the license server host. The program that allows users to share licenses runs on this system. The license server host can be the system on which you installed FrameMaker or any other system on your network.

### Setting up workgroups for shared licenses

If you purchase several sets of group licenses (for example, one set of 10 licenses and another set of 20 licenses), you can choose to install each set of licenses as a separate workgroup. Each workgroup will have its own license server, licenses file, and other supporting licensing files.

There are many advantages to setting up multiple workgroups. You can control access to FrameMaker for individual persons or departments. (Set up one workgroup for an engineering department, another for finance, and a third for marketing and control access for each department.)

You can also set up several workgroups where all users have access. Doing this gives you several license servers instead of just one. The benefit is that if one of the license servers is problematic, users can get licenses from another server.

When you have multiple workgroups, you install the licensing information for each workgroup in a separate directory. Use standard UNIX permissions to control access to workgroup licensing files.



## Setting up license servers

After the installation script installs FrameMaker, it recommends that you install licenses. To do this, you run the `fmsetupfls` script, which sets up the license server. You can also run `fmsetupfls` at a later time to set up additional licenses.

**Important:** *You must run `fmsetupfls` on the license server host.*

### To set up the license server:

1 On the license server host, start the `fmsetupfls` script by doing one of the following:

- If you are setting up licensing centrally for an entire site (not separately for workgroups), enter:

The `fmsetupfls` script sets up licensing files in `$FMHOME/fminit`.

- If you are setting up licensing for a specific workgroup, enter:

***workgroup\_dir***

where ***workgroup\_dir*** is the full pathname of the workgroup directory where the licensing files will be stored. Specify a directory to which all members of the workgroup have read and execute permission.

- If you are setting up a shared license and you want to specify a different location for the licensing files, enter:

***<different\_location\_dir>***

where `<different_location_dir>` is the full pathname of the directory where the licensing files will be stored. Specify a directory to which all members of the workgroup have read and execute permission.

**2** Provide the information that the script requests. The script creates the scripts that control shared licenses on your network. It stores these scripts in the directory you specified when you started `fmsetupfls`. It also asks where you want to store log files that the license server process creates. For more information, see “How licensing works” on page 44.

**3** When the `fmsetupfls` script starts the `fmaddlicense` program, enter the licensing information. Use `fmaddlicense` to enter the information from your registration card (or other licensing sheet) into the licenses file. For instructions, see “Using the `fmaddlicense` program” on page 37.

**4** When you finish entering the licensing information, exit `fmaddlicense`. For instructions, see “Exiting `fmaddlicense`” on page 41. The `fmsetupfls` script continues.

## Setting up environments for a workgroup

After you set up licenses in workgroups, you need to adjust each user’s environment so the necessary environment variables are set correctly. If you set up a single, central licenses file in `$FMHOME/fmunit`, you can skip this step.

FrameMaker licensing variables are set by the `env.csh` script (for the C shell) or the `env.sh` script (for the Bourne or Korn shell) from the workgroup directory. You can set up these scripts to run when a user starts FrameMaker or when a user logs in.

**To set up licensing variables when a user starts FrameMaker:**

- 1 Create an fminit directory in the user's home directory.
- 2 Change to that directory.
- 3 Create a symbolic link to the workgroup's env.sh file, by entering this command (where workgroup\_dir is the workgroup directory you specified when you started fmsetupfls):

```
workgroup_dir/env.sh env.sh
```

**To set up licensing variables when a user logs in, do one of the following (where workgroup\_dir is the workgroup directory you specified when you started fmsetupfls):**

- If you use the C shell, add the following line to the user's .login file:  

```
workgroup_dir
```
- If you use the Bourne or Korn shell, add the following line to the user's .profile file:  

```
workgroup_dir
```

## Setting up personal licenses

You set up personal licenses by entering the information from your registration card (or other licensing sheet) into the licenses file. You can enter personal licensing information for multiple users in a central licenses file in \$FMHOME, or you can enter each user's licensing information in a personal licenses file in the user's home directory.

### **To set up a central licenses file for multiple users in \$FMHOME:**

- 1 Start the fmaddlicense program by entering:
  
- 2 Use fmaddlicense to enter the information from your registration card (or other licensing sheet) into the licenses file. For instructions, see "Using the fmaddlicense program" on page 37. The fmaddlicense program records licensing information in \$FMHOME/fminit/licenses. You can set up several personal licenses in this file.

### **To set up a personal licenses file in a user's home directory:**

- 1 Change to the user's home directory and then create a personal fminit directory by entering:

- 2 Start the `fmaddlicense` program by entering:
- 3 Use `fmaddlicense` to enter the information on your registration card (or other licensing sheet) into the licenses file. For instructions, see the following section, “Using the `fmaddlicense` program.” Enter licensing information for one user only.

## Using the `fmaddlicense` program

The `fmaddlicense` program is your tool for adding and administering licenses. This program runs automatically when you set up the license server (see “Setting up license servers” on page 33). You can also run the program at any later time. The first time you add a license, `fmaddlicense` asks you to enter the name of the license owner. Enter your company or organization name.

**Important:** *You cannot edit the licenses file in a text editor because this may corrupt the file. Instead, run `fmaddlicense` again, and back up your licenses file if you want to re-create the file.*

### To run `fmaddlicense`:

- 1 In a UNIX window, enter:

*license\_path*

where *license\_path* is the path to the licenses file. The default path is \$FMHOME/fmunit. The fmaddlicense prompt appears:

**fmaddlicense->**

**2** Use fmaddlicense by entering commands at the prompt and pressing Return. The rest of this section explains the fmaddlicense commands.

## Getting help

You can view a list of fmaddlicense commands. Parameters in brackets are optional.

### To view a list of fmaddlicense commands:

Enter one of the following commands:

- ?
- help

## Adding licenses

You can add shared and personal licenses.

**To add a shared license:**

Enter `registration_number`

where *registration\_number* is the registration number on your registration card (or other licensing sheet). If the registration number is not a valid shared license registration number, `fmaddlicense` warns you. Make sure that you entered the registration number correctly.

**To add a personal license:**

Enter `registration_number user_name`

where *registration\_number* is the registration number on your registration card (or other licensing sheet), and *user\_name* is the user name of the user who will own the license. The `fmaddlicense` program checks that you entered the registration number correctly and then records the information in the licenses file.

## Listing licenses

To verify the licenses you added, you can list them.

**To list licenses in the licenses file:**

Enter

For each license, `fmaddlicense` reports product type, license type, and part of the registration number. It also reports the user names assigned to personal licenses. For example:

```
License FrameMaker Personal   Adobe   11-1-11111
```

Assigned to: `user_name`

where `user_name` is the login ID of the user of the personal license.

In FrameMaker 8, Adobe serial numbers are locale-independent, and the product type is generic across locales. For example:

Locale	Product type in 7.2	Product type in 8
English	FrameMaker-USUK	FrameMaker
French	FrameMaker-E	FrameMaker
German	FrameMaker-E	FrameMaker
Japanese	FrameMaker-J	FrameMaker



## Removing and reassigning licenses

You must delete a license before moving it to a different licenses file. You cannot use the same license in two licenses files.

### To remove a personal license from your license file:

Enter `registration_number`

### To reassign a personal license to a different user:

- 1 Delete the license (see the preceding instruction).
- 2 Add the license to the file again, assigning it to the new user. (See “Adding licenses” on page 38.)

**Important:** You must enter the information from the original registration card (or other licensing sheet) when you reassign a license.

## Exiting fmaddlicense

When you are finished using fmaddlicense, exit the program.

### To exit the fmaddlicense program:

Enter

To have your changes take effect, you must reinitialize the license server process on the license server to load the new licensing information from the licenses file. Before exiting, `fmaddlicense` asks whether you want to reinitialize the license server process. If you choose not to reinitialize at this time, you can reinitialize the process later. For information, see “Reinitializing the license server” on page 43.

## Registering the product

When you exit, `fmaddlicense` asks if you want to register your new license(s) by:

- sending in the registration card

or

- visiting the Adobe website at [www.adobe.com/support](http://www.adobe.com/support), and then under “Next Steps” clicking Register Software.

## Managing a license server

You can use the `fmflsadm` program to manage the license server process on the license server. The `fmflsadm` commands have the following format:

*host command*

where *host* is the name of the license server host and *command* is an `fmflsadm` command. You can omit `-flshost` *host* to check the host specified by the `FM_FLS_HOST` environment variable.

## Checking the license server

Use the `fmflsadm ping` command to check whether the license server process is running on a license server host.

## Reinitializing the license server

Use the `fmflsadm reinit` command to reinitialize the license server by loading licensing information from the `licenses` file. Normally, the license server process reads licensing information only when it starts. If you make changes to the `licenses` file while the license server process is running, you must either reinitialize or exit and restart the license server process to put your changes into effect.

# Starting the license server automatically

You can start the license server process automatically whenever the license server host starts.

**To start the license server process automatically:**

Add the following line to the license server host's system startup file:

```
licenses_file > /dev/console 2>&1
```

where *licenses\_file* is the full pathname of the licenses file you want to use.

## How licensing works

This section gives more detailed information about licensing.

### Licensing files, scripts, programs, and variables

The `fmsetupfls` script creates the scripts that control shared licenses on your network. If you set up a single workgroup for shared licenses, these files are in `$FMHOME/fminit`. If you set up several workgroups, these files are in each workgroup directory.

**fm\_fls\_auto script** FrameMaker uses the `fm_fls_auto` script (the script named by the `FM_FLS_AUTO` environment variable) to start the `fm_fls` license server program on the license server host when it isn't already running.

**env.sh and env.csh scripts** The `fmsetupfls` program creates `env.sh` and `env.csh` scripts, or updates them if they already exist. These scripts set the values of the `FM_FLS_HOST` and `FM_FLS_AUTO` licensing environment variables. FrameMaker uses `env.sh` to look up the values of these variables. There are sample scripts in the files `$FMHOME/fmunit/env.sh.ex` and `$FMHOME/fmunit/env.csh.ex`.

**FM\_FLS\_HOST environment variable** `FM_FLS_HOST` contains the name of the license server host. When FrameMaker starts, it runs `$HOME/fmunit/env.sh` or `$FMHOME/fmunit/env.sh` to look up the values of `FM_FLS_HOST` and `FM_FLS_AUTO`.

**FM\_FLS\_AUTO environment variable** The `env.sh` or `env.csh` script automatically sets `FM_FLS_AUTO` to the pathname of the `fm_flc_auto` script that starts the license server process. FrameMaker runs `fm_flc_auto` when it can't communicate with the license server process on the license server host. If this variable is not set, FrameMaker cannot start the license server process automatically.

**fm\_flc program** The `fm_flc` program runs the license server process on the license server host. This process must be running for users to obtain shared licenses. It runs until you kill it or the license server reboots.

**fm\_flb program** When FrameMaker starts, it runs the `fm_flb` program on the host on which it was started. The `fm_flb` program periodically exchanges licensing information with the `fm_flc` program running on the license server host. If there is more than one copy of FrameMaker running on a host, all products share one `fm_flb` program. The `fm_flb` program remains running for up to one hour after FrameMaker exits.

***Note:** The `fm_flb` license broadcaster is designed to transmit information from each machine running FrameMaker to ensure that there are no duplicate licenses being used on the network. All network broadcasts communicate with all I/O ports, which could activate dial-on-demand connections. It is the duty of the network administrator to configure networks and routers responsibly.*

**fm\_fls.log file** The `fm_fls` program stores information about license usage in the `fm_fls.log` file. When you run `fmsetupfls` to set up license servers, you can specify the directory in which to store this file. Otherwise, it is stored in `$FMHOME/fm_init/tmp`. You can use the `fmreport` program to display the contents of the log file. For more information, see “Generating license reports” on page 48.

## What happens when a user starts FrameMaker

The following is a summary of what happens when a user starts FrameMaker.

- 1 When the user gives the command to start the product, a startup script looks for the `env.sh` script in `$HOME/fm_init` and then in `$FMHOME/fm_init`.
  - If the startup script finds the `env.sh` script, it sources `env.sh` (attempting to set the `FM_FLS_HOST` and `FM_FLS_AUTO` environment variables) and starts FrameMaker. The product continues with step 2.
  - If the startup script doesn't find the `env.sh` script, the product starts without attempting to set the environment variables.

**2** The product searches for a personal license in \$HOME/fmunit/licenses and then in \$FMHOME/fmunit/licenses.

- If it finds a personal license, the licensed version of the product starts.
- If a licenses file exists but the product doesn't find a personal license, the product continues with step 3.

**3** The product checks the licensing environment variable, FM\_FLS\_HOST, to determine whether shared licensing is installed.

- If FM\_FLS\_HOST is set, the product continues with step 4.
- If FM\_FLS\_HOST is not set, the product displays an alert that the user doesn't have a license. The user may be able to get a license if another license server process is running on a different host. For instructions, see the *Install\_Readme* file.
- If FM\_FLS\_HOST is not set and no licenses file exists, the product displays a message that licensing is not installed and asks whether the user wants to run a demonstration version. (With a demonstration version, the user can't save files or print files and can't obtain a license.)

**4** The product checks for a license server process, fm\_fls, running on the license server host named by FM\_FLS\_HOST.

- If fm\_fls is running, the product continues with step 5.

- If `fm_fl`s doesn't respond, the product runs the `fm_fl`s\_auto script (the script named by `FM_FLS_AUTO`) to start the license server process, and then continues with step 5. (It can't run `fm_fl`s\_auto if `FM_FLS_AUTO` is not set. In that case, the product displays an alert that the user doesn't have a license. The user may be able to get a license from a process running on a different host (see the *Install\_Readme* file).
- 5 The product requests a license from the license server process.
- If `fm_fl`s assigns a license, the licensed version of the product starts.
  - If `fm_fl`s doesn't assign a license, the product starts but displays an alert that the user doesn't have a license. The user may be able to get a license from a process running on a different host (see the *Install\_Readme* file).

## Generating license reports

The `fm_fl`s license server process keeps a log of license activity in the `fm_fl`s.log file (see “`fm_fl`s.log file” on page 46). You can use the `fmreport` program to retrieve licensing information from each of the license server hosts and format it into a report.

### To generate a license report:

Enter `product log_file`



The *log\_file* option specifies the pathname of the fm\_fls.log file from which you will generate the report. This pathname is required.

The fmreport program reports the following:

- The number of times the license server was started and the date and time of each start
- The number of times the license server detected illegal license use, the name of the illegal user, and the date and time
- The length of time and time of day of the shortest and longest license use

## Troubleshooting licensing

This section describes some licensing problems and suggests troubleshooting techniques and possible solutions.

### Displaying licensing messages

If users are having problems with licensing, you can have them instruct FrameMaker to display additional messages as it attempts to start. The messages may help you in identifying the source of the problems.

**To display additional messages at startup:**

Add the `-nlverbose` command-line option to the command that starts the product. For example: -

For a list of commands for starting FrameMaker, see , “Starting FrameMaker.”

## Specifying personal licenses

If both personal and shared licenses are installed on your network, users can specify the type of license they want FrameMaker to use. If FrameMaker cannot obtain that type of license, it starts a demonstration version.

**To specify a personal license:**

Add the `-personalLicenseOnly` command-line option to the command to start the product. For example:  
**`-personalLicenseOnly`**

**To specify a shared license:**

Add the `-sharedLicenseOnly` command-line option to the command to start the product. For example: **`maker -sharedLicenseOnly`**

## **FrameMaker reports that there are no licenses**

This problem occurs when the FrameMaker cannot find the licenses file in \$HOME/fmunit or in \$FMHOME/fmunit. It asks if you want to run a demonstration version.

You might not have set up the licenses file, or it might have been unintentionally removed. Restore the licenses file or use the information from your registration card (or other licensing sheet) and the fmaddlicense program to re-create the file.

## **FrameMaker reports that the license server failed**

If you changed the owner name in the licenses file, the product reports that the server failed or that it was unable to open the file. Do not change this name after entering it with fmaddlicense. For more information, see “Using the fmaddlicense program” on page 37.

## **FrameMaker cannot communicate with the license server**

If the message “Having trouble communicating with the license server” appears in a UNIX window, the license server process probably isn’t running on the host you specified. First, make sure the license server host specified in the License dialog box is correct. Then follow these instructions.

**To get a license when the product cannot communicate with the license server:**

1 Do one of the following:

- If your workstation is the license server host, change the name of the license server host in the License dialog box to localhost and try to get a license again.
- If the license server host is a remote host, or if the preceding action doesn't solve the problem, continue with the following steps.

2 Log in to the license server host (using the rlogin, rsh, or remsh command if the host is remote).

3 Make sure the Port Mapper is running by entering the following command:

- **ps -e | grep rpcbind**

If the Port Mapper is running, a message similar to the following appears:

- **51 ? 3:39 rpcbind**

4 If the Port Mapper isn't running, enter the following command:

- **/etc/rc2.d/S71rpc rpcstart**

You may need to be logged in as root to run the Port Mapper.

5 Check whether the license server process is running by entering:

**fmflsadm -flshost *host* ping**

where **host** is the name of the license server host. You can omit `-flshost host` to check the host specified by the `FM_FLS_HOST` environment variable. The `fmflsadm` program reports whether or not the license server process is running.

**6** If the process isn't running, restart the license server process by entering:

```
$FMHOME/bin/fmsetupfls workgroup_dir
```

where ***workgroup\_dir*** is the licensing directory for the workgroup. If you are changing licensing information for an entire site (not for a workgroup), omit this directory from the command.

Messages appear in a UNIX window indicating that the license server process is initializing. If the process still doesn't start, write down the error messages that appear and contact your technical support representative.

**7** Try to get a license again. If you still can't get a license, contact your technical support representative.

## **The license server process is running, but you cannot get a license**

Even though the license server process is running, you might be unable to obtain a license for the following reasons:

- The license server host name that you entered in the License dialog box was incorrect.

- The date on your workstation is different from the date on the license server host. (If the difference is 4 hours or less, you should be able to obtain a license. If the difference is between 4 and 8 hours, you may be able to obtain a license. If the difference is more than 8 hours, you won't be able to obtain a license.)
- The licenses file was updated after the license server process was started. If this is the case, you can instruct the process to reread the file.

**To instruct the license server process to reread the licenses file:**

Enter `-flshost host`

where *host* is the name of the license server host. You can omit `-flshost host` to use the host specified by the `FM_FLS_HOST` environment variable.

## Manually starting the license server

If the license server (`fm_fls`) doesn't start automatically, you can start it manually, and optionally specify a location for the licensing log file (`fm_fls.log`).

**To start the license server manually:**

Enter one of the following commands (where *licenses\_file* is the full pathname to the licenses file you want to use):

- To start the server and store the licensing log file in the standard location, enter:

*licenses\_file*

- To start the server and specify a nonstandard location for the licensing log file, enter:

*licenses\_file log\_file\_path*

where *log\_file\_path* is the full pathname to the directory in which you want to store the log file. For more information on the log file, see “fm\_fls.log file” on page 46.

## Moving a license server

Follow these steps to move the license server process to another system:

- 1 Instruct all users to exit FrameMaker.
- 2 On the current license server, search for the fm\_fls license server process by entering the following command:

-

If the process is running, a message reports the process number. For example:

- 3 Kill the license server process by entering - *process\_number*

For example: -

**4** On any systems that are still running FrameMaker, kill the `fm_flb` process. For each workstation, use the command shown in step 3 to determine the process number, and use the `kill` command to kill the process. If you skip this step, users might see an error message and have to provide the name of the new license server in order to continue.

**5** Use the `fmsetupfls` program to set up the new license server. For instructions, see “Setting up license servers” on page 33.



# Chapter 5: Starting FrameMaker

This chapter describes how to start licensed and demonstration versions of Adobe® FrameMaker® and how to run the product on a remote workstation or server. While running a demonstration version, you can use all features except the Save command.

## Starting FrameMaker

To start FrameMaker, enter in a UNIX window one of the commands in the following table, where *language* represents a user-interface language (for example, ukenglish, deutsch, francais, or japanese).

To start	For a licensed version, enter	For a demonstration version, enter
FrameMaker (default language)	maker	demomaker
FrameMaker International Edition	maker -l (lowercase L) <i>language</i>	demomaker -l (lowercase L) <i>language</i>

**Note:** *These commands work only if you have put the FrameMaker directory in your path, as described in , “Setting up the user environment.” If you have not done so, you must enter the full pathname (for example, /usr/frame/bin/maker).*

There are also many command-line options you can use to customize FrameMaker. For example, you can specify a file to open when the product starts, or change a resource for that session only. For information, see the online manual *Customizing FrameMaker* on the Adobe website: [www.adobe.com/devnet/frame-maker/pdfs/Customizing\\_Frame\\_Products.pdf](http://www.adobe.com/devnet/frame-maker/pdfs/Customizing_Frame_Products.pdf).

When you start a licensed version of FrameMaker, it searches for licensing information. If licenses are not set up, it asks if you want to start the demonstration version. For more information, see “What happens when a user starts FrameMaker” on page 46. For information on setting up licenses, see , “Setting up licenses.”

**Note:** *Be sure to warn users not to start the demonstration version if they want to use a license. They can't save their work or obtain a license from within the demonstration version.*

## Running FrameMaker on a remote system

FrameMaker can run locally or remotely. When running locally, FrameMaker runs as a client application on a user's workstation. When running remotely, FrameMaker runs as a client application on a remote system, and employs the user's workstation only as a display server, thus conserving workstation resources. When deciding whether to run FrameMaker locally or remotely, consider the following:

- If users run FrameMaker remotely, their workstations won't have the added overhead of running the product. This improves local system performance.
- If the network is running near capacity, the additional load required to transmit display information from the remote system can degrade network and workstation performance.

### To run FrameMaker remotely:

**1** Allow the remote host to use a workstation as a display server by doing one of the following (where *client\_host* is the name of the host on which you'll run FrameMaker):

- To specify a remote host, enter *client\_host*
- To allow any host to use the workstation as a display server, enter **xhost +**

**2** Log in to the host on which you'll run the product by entering **rlogin *client\_host***

**3** Designate a workstation as a display server by doing one of the following (where *myhost* is the host name of the workstation):

- If you are using the C shell, enter **setenv DISPLAY *myhost*:0.0**
- If you are using the Bourne shell or Korn shell, enter:  
**DISPLAY=*myhost*:0;export DISPLAY**
- If you don't want to change the DISPLAY variable, instruct users to add the -display hostname:0 option to the command they use to start FrameMaker. You will need to use this option each time they start the product.

**4** Start FrameMaker (see “Starting FrameMaker” on page 57).

# Appendix A: Upgrading from an earlier version

If you have an earlier version of Adobe® FrameMaker® installed, you might want to retain its setup files and font information. After you install the new FrameMaker files, follow the instructions in this appendix for copying files from the earlier version's installation directory.

For information on new features in version 8 of FrameMaker, see the *FrameMaker User Guide*.

## Using existing licenses files

You must install new licenses for FrameMaker 8 that you are installing. If you have existing licenses files from version 7.2 (or earlier) of FrameMaker, you can add the new licenses to these files. Copy the version 8 licenses file to the version 8 \$FMHOME/fmunit directory, and then run fmsetupfls from the version 8 installation directory. For instructions, see “Setting up license servers” on page 33 and “Using two versions simultaneously” on page 63.

## Using existing setup files

The resource files and other setup files in `$FMHOME/fmunit` and `$FMHOME/fmunit/UILanguage` directories let you customize FrameMaker for your installation. (*UILanguage* is a user-interface language directory, for example, `usenglish` or `francais`.) For more information about these setup files and about the structure of the `fmunit` directory, see the online manual *Customizing FrameMaker* on the Adobe website: [www.adobe.com/devnet/frame-maker/pdfs/Customizing\\_Frame\\_Products.pdf](http://www.adobe.com/devnet/frame-maker/pdfs/Customizing_Frame_Products.pdf).

Users can store their own custom FrameMaker settings in setup files within personal `fmunit` directories in their home directories and other directories. When FrameMaker starts, it searches for setup files in the user's personal `fmunit` directory before it uses `$FMHOME/fmunit`. Users switching to FrameMaker 8 should create new personal `fmunit` directories and follow instructions in this section for updating setup files.

### Using existing X Window System resource files

UNIX versions of FrameMaker store most setup information as X Window System resources. Existing resource files will work with FrameMaker 8 because the products ignore obsolete resources.

To customize the new features of FrameMaker, compare your existing resource files with the standard ones installed with this version. (If you open the files in FrameMaker, you can use the document comparison feature to do this.) Add the new resources to your existing resource files. For more information about resources, see the online manual *Customizing FrameMaker* on the Adobe website: [www.adobe.com/devnet/frame-maker/pdfs/Customizing\\_Frame\\_Products.pdf](http://www.adobe.com/devnet/frame-maker/pdfs/Customizing_Frame_Products.pdf).

## Using two versions simultaneously

You can run two versions of FrameMaker at the same time (for example, FrameMaker 8 and FrameMaker 7.2), but a single license server can only run one license server process at a time. To run two versions simultaneously, you must do one of the following:

- Use different systems as the license servers for the two versions.
- Add your version 7.2 licenses to your version 8 licenses file as described in “Using existing licenses files” on page 61. Then copy the following files from the version 7.2 \$FMHOME/fmunit directory (or your version 7.2 licensing directory, if you specified a different location) to the version 8 \$FMHOME/fmunit directory (or licensing directory): licenses, env.csh, env.sh, and fm\_fl\_s\_auto. (If the FM\_FLS\_AUTO environment variable is set, it must name the 8 version of the fm\_fl\_s\_auto file.)

***Note:** With version 8, you can open documents created with earlier versions of FrameMaker. However, you can open a FrameMaker 8 document in earlier versions only indirectly. First you must open the document in version 8 and save it as a FrameMaker 7.0 or Maker Interchange Format (MIF) 7.0 document. You can then open the FM or MIF file in the earlier version. New features in version 8 are ignored, and messages in a UNIX window will report MIF statements for these features as being “skipped.”*

## Using templates

FrameMaker 8 stores templates in this directory: `$FMHOME/fmunit/UILanguage/Templates`. You can use templates from version 7.2 or earlier by copying them from your old templates directory into the current directory. To convert the templates to the current format, open and save them with FrameMaker 8.

## Using fonts

FrameMaker provides font files for several Type 1 fonts. For information on adding fonts, see *Working with Fonts in FrameMaker for UNIX*, available in this path: `$FMHOME/fminit/<UILanguage>/Documents/Unix_Fonts.pdf`.



If you added any other Adobe Type 1 fonts to version 7.2 or earlier, follow these steps to add each font to the current version.

**To add fonts from version 7.2 or earlier to the current version:**

Copy all custom font files to the current font directory (\$FMHOME/fmunit/fontdir).

For a complete description of the fontlist format, see *Working with Fonts in FrameMaker for UNIX*, available in this path: \$FMHOME/fmunit/<UILanguage>/Documents/Unix\_Fonts.pdf.



# Index

## A

- add command 27
- Adobe Acrobat 7.0, installing 5
- Adobe Acrobat Reader, installing 5
- aliases for starting FrameMaker products 18

## C

- CD-ROM drive
  - checking directory for 9
  - mounting 9–12
- configuration files, using from an older release 41
- copying FrameMaker product files 14

## D

- delete command 28
- desktop environments, integrating with 15
- directory, FrameMaker product 5
- done command 28

## E

- env.csh script 30
- env.sh script 30
- environment variables
  - FM\_FLS\_AUTO 30
  - FM\_FLS\_HOST 30
  - FMHOME 15
  - LANG 6
  - XKEYSYMDB 19

## F

- files, setup, using from an older release 41
- fm\_flb program 30
- fm\_fls program 30
- fm\_fls.log file 31, 32
- FM\_FLS\_AUTO environment variable 30
- FM\_FLS\_HOST environment variable 30
- fmaddlicense program 26–28
  - add command 27
  - delete command 28
  - done command 28
  - exiting 28
  - help command 26
  - list command 27

fmcopyscript 14  
 fmflsadm program 29–??  
     ping command 29  
     reinit command 29  
 FMHOME environment variable  
     15  
 fmreport program 32  
 fmsetupfls script 23  
 fmusersetup script 15  
 fonts  
     installing 7  
     using, added in an older version  
         43  
 FrameMaker product directory,  
     creating 5  
 FrameMaker product files,  
     copying 14  
 FrameMaker product setup files,  
     using from an older release 41  
 FrameMaker products  
     running, on remote system 40  
     starting 39–40

**H**  
 help command 26  
  
**I**  
 installing 5–14  
     additional platforms later 13  
     basic or custom 6  
     creating a FrameMaker product  
         directory 5  
     fonts 7  
     fonts added in an earlier release  
         43  
     over an older release 41–??  
     running the installation script  
         12  
     sample documents and online  
         manuals 7  
     selecting files 6  
     setting users' environments  
         15–20  
     templates from an earlier release  
         42

    using setup files from an older  
         release 41

**J**  
 Japanese functionality  
     system requirements for 1  
     user interface installation 6  
  
**L**  
 LANG environment variable 6  
 language of user interface,  
     specifying 6, 39  
 license reports 32  
 license server host 23  
 licenses  
     adding to the licenses file 27  
     checking for license server  
         process 29  
     communicating with license  
         server, problems in 34  
     displaying messages 33

- finding, problems in 34
- listing 27
- network requirements 22
- obtaining, problems in 35
- personal 21
- process for getting 31
- reassigning 27
- registration numbers for 22
- removing 27
- reporting usage 32
- rereading the licenses file 29
- setting up 21-??
- starting the license server 36
- troubleshooting 33-??
- licenses file 22-31, 34, 41
- licensing components 30-31
- list command 27

**M**

- multiple-platform networks 1

**P**

- personal licenses 21
  - setting up 25
- ping command 29

**R**

- registration numbers 22
- reinit command 29
- requirements, system 1

**S**

- scripts
  - env.csh 30
  - env.sh 30
  - fmcopy 14
  - fmsetupfls 23
  - fmusersetup 15
  - for starting FrameMaker products 18
  - installation 12

- startup for FrameMaker products 31
- setup files, using from an older release 41
- shared licenses
  - network requirements 22
  - setting up 23
- starting FrameMaker products 39-40
  - licensing processes 31
- Sun systems, system requirements 1
- system requirements 1

**T**

- templates, using from an earlier release 42

**U**

- upgrading from earlier releases 41-??

## **V**

variables, environment

FM\_FLS\_AUTO 30

FM\_FLS\_HOST 30

FMHOME 15

LANG 6

XKEYSYMDB 19

## **X**

XKEYSYMDB environment  
variable 19

XkeysymDB file, using updated 19