

ClearDDTS Installation and Licensing Guide

Version 4.7

support@rational.com
<http://www.rational.com>

Rational
the e-development company™

IMPORTANT NOTICE

COPYRIGHT NOTICE

Copyright © 1988 — 2000 Rational Software Corporation. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, photocopying, recording or otherwise, without prior written consent of Rational . No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this book, Rational assumes no responsibility for errors or omissions. This publication and features described herein are subject to change without notice.

The program and information contained herein are licensed only pursuant to a license agreement that contains use, reverse engineering, disclosure and other restrictions; accordingly, it is “Unpublished — rights reserved under the copyright laws of the United States” for purposes of the FARs.

DISCLAIMER OF WARRANTY

Rational makes no representations or warranties, either express or implied, by or with respect to anything in this manual, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to the restrictions as set forth in subparagraph (c) (1) (a) of the Rights in Technical Data and Computer Software clause of the DFARs 252.227-7013 and FAR 52.227-19(c) and any successor rules or regulations.

TRADEMARKS

ClearDDTS, PureDDTS, ClearCase, Rational and the Rational logo are U. S. trademarks of Rational Software Corporation.

All other products or services mentioned in this manual are covered by the trademarks, service marks, or product names as designated by the companies who market those products.

Printed in the U.S.A. on recycled paper.

Part Number: 800-025151-000

Contents

Intended audience	v
Using this guide	v
Where to go for more information	vi
Questions or suggestions?	vii
1 Installing ClearDDTS for the First Time	
Before You Begin	1-2
Disk and Memory Requirements	1-2
CPU Requirements	1-3
Preparing the Environment	1-3
Adding aliases for the CGI scripts to your HTTP server configuration 1-5	
Example of how to modify the Apache server configuration ...	1-6
Extracting the software	1-7
Installing ClearDDTS	1-8
Starting an evaluation license manager	1-11
Running the webddts interface	1-12
Selecting an External Oracle Database	1-13
Checking the Installation and Informing New Users	1-13
Mounting the ClearDDTS Home Directory	1-14
Making ClearDDTS Available to Web Users	1-14
Making ClearDDTS Available to UNIX Users	1-15
Sending E-Mail to the User Community	1-16
2 Upgrading from a Previous Release	
Upgrading from releases prior to 4.1	2-2
Licensing with FLEXlm	2-2
Adding aliases for the CGI scripts to your HTTP server configuration	

2-2	
Example of how to modify the Apache server configuration . . .	2-4
Backing up your current installation	2-5
Extracting the software	2-5
Determining changes to implement	2-6
Upgrading ClearDDTS	2-6
Starting an evaluation license manager	2-11
Running the webddts interface	2-12
3 Managing Licenses	
How licenses work	3-1
What is licensed	3-2
The license manager	3-3
The license file	3-3
License manager commands	3-4
Imgrd	3-4
Imhostid	3-4
Imstat	3-5
Imdown	3-5
Imrread	3-5
Setting up permanent licensing	3-5
Obtaining licenses	3-6
Removing your evaluation license setup	3-6
Starting a new FLEXIm license manager	3-7
Using an existing FLEXIm license manager	3-8
Restarting Imgrd automatically	3-10
Adding additional licenses	3-10
Changing your server host	3-11
4 Using FastCGI	
Using Native FastCGI	4-1
Running without FastCGI	4-3

Preface

Welcome to ClearDDTS™, Rational's Distributed Defect Tracking system. ClearDDTS tracks and manages product feedback such as defects ("bugs") and enhancement requests throughout the product's life cycle.

Intended audience

The *ClearDDTS Installation and Licensing Guide* is intended for system administrators and managers responsible for installing and upgrading the product and maintaining licenses. This guide describes how to manage a ClearDDTS installation and configure licensing to suit your business needs. For information about administering or customizing the system, see the *ClearDDTS Administrator's Guide*.

Although basic installation is simple, it is helpful to be familiar with the following:

- UNIX operating system and common UNIX commands (or the POSIX shell if working in a POSIX-compliant environment)
- basic system administration commands for your platform
- requirements of your business environment
- web servers, client browsers, and your HTTP server configuration for use with the *webddts* interface

Using this guide

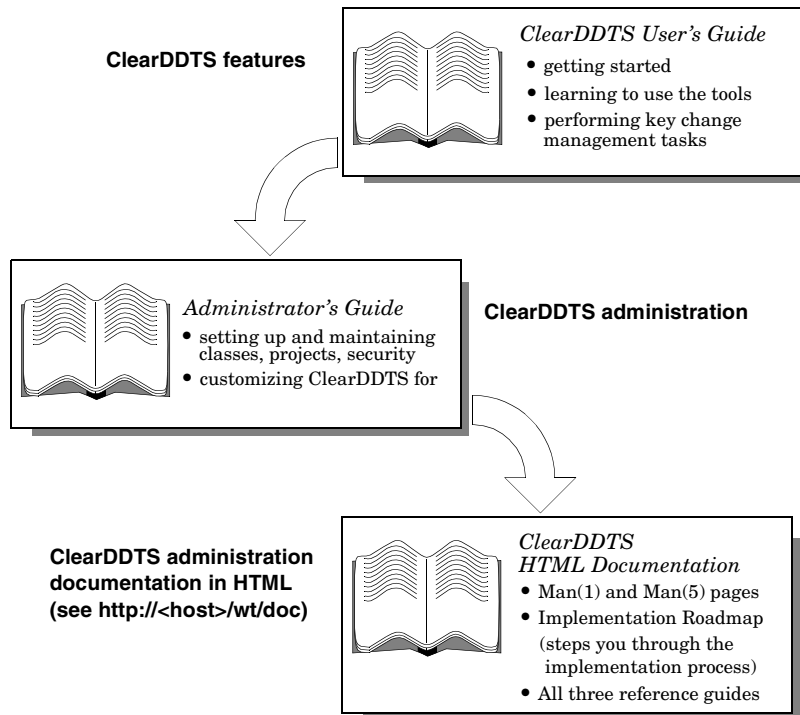
The following topics are covered in this guide:

- Chapter 1, *Installing ClearDDTS for the First Time*, provides steps for establishing the administrator user and installing ClearDDTS.

- Chapter 2, *Upgrading from a Previous Release*, describes the steps for upgrading from a previous release of ClearDDTS.
- Chapter 3, *Managing Licenses*, describes how to set up and manage licensing in ClearDDTS.
- Chapter 4, *Using FastCGI*, describes FastCGI technology and your options for using it with ClearDDTS.

Where to go for more information

For special notes about this release of ClearDDTS, see the *Release Notes* file, `[ddts_home]/RELEASE_NOTES`. In addition to this guide, the ClearDDTS documentation set includes several other sources of information. Use the following roadmap to guide you to the documents that best suit your needs:



Questions or suggestions?

For technical support or suggestions for improving ClearDDTS or this guide, contact the nearest Rational Technical Support Center.

Rational U.S.A.	Rational Europe	Rational K.K.
18880 Homestead Road Cupertino, CA 95014 U.S.A	Beechavenue 30 1119 PV Schiphol-Rijk The Netherlands	Kyowa Shinkawa Bldg. 2020-8 Shinkawa Chuo-ku, Tokyo 104 Japan
Tel: (800) 433-5444 or (408) 863-4000 Fax: (408) 863-4590 e-mail: info@rational.com support@rational.com	Tel:+31 (0)20 4546 200 Fax: +31 (0)20 4546 201 e-mail: info@europe.rational.com support@europe.rational.com	Tel:+81 3 3551 9370 Fax:+81 3 3551 9362 e-mail: info@japan.rational.com support@japan.rational.com
URL: http://www.rational.com		

If you have questions regarding the licensing of your Rational Software products, please contact the appropriate Licensing Support office.

North & Latin America	Europe, Israel, Africa	Japan
Phone : 800-728-1212 FAX : 781-676-2510 Email : lic_americas@rational.com	Phone : +-31 23 554 10 62 FAX : +-31 23 554 10 69 Email : lic_europe@rational.com	Phone : +-81 3 5423 3611 FAX : +-81 3 5423 3622 Email : lic_japan@rational.com

North Asia Pacific (Mainland China, Hong Kong, Taiwan)	North Asia Pacific (Korea)	South Asia Pacific (Australia, New Zealand, Malaysia, Singapore, Indonesia, Thailand, The Philippines, Vietnam, Guam and India)
Phone : +-852 2143 6382 FAX : +-852 2143 6018 Email : lic_apac@rational.com	Phone : +-82 2 556 9420 FAX : +-82 2 556 9426 Email : lic_apac@rational.com	Phone : +-612 9419 0100 FAX : +-612 9419 0160 Email : lic_apac@rational.com

1

Installing ClearDDTS for the First Time

This chapter describes the process for installing ClearDDTS for the first time.

Note: If you are upgrading from a previous release, see Chapter 2, *Upgrading from a Previous Release*.

The following steps summarize the installation procedure. Each section of this chapter describes a step in detail:

- 1 Check that your system meets the described requirements in "Before You Begin" on page 1-2.
- 2 Prepare your environment for installation by setting up the special administrative user *ddts* and the home directory where ClearDDTS will reside. See "Preparing the Environment" on page 1-3.
- 3 Add aliases for the ClearDDTS CGI scripts to your HTTP server configuration. See "Adding aliases for the CGI scripts to your HTTP server configuration" on page 1-5.
- 4 Extract the ClearDDTS files from the installation media. See "Extracting the software" on page 1-7.
- 5 Install ClearDDTS. See "Installing ClearDDTS" on page 1-8.
- 6 Start a FLEXlm license manager to use with your Startup license key. See "Starting an evaluation license manager" on page 1-11.
- 7 Run the *webddts* interface to complete the web configuration. See "Running the webddts interface" on page 1-12.
- 8 Optionally, define an external ORACLE database. See "Selecting an External Oracle Database" on page 1-13.

- 9 Inform ClearDDTS users how to access ClearDDTS. See "Checking the Installation and Informing New Users" on page 1-13.

Before You Begin

Before installing ClearDDTS, be sure you have the following:

- the product CD-ROM
- sufficient disk space and memory (see the following table)
- the temporary Startup license key certificate shipped with ClearDDTS

Note: The Startup license key is an evaluation license that works on any host and expires after approximately one month. It allows you to get ClearDDTS up and running while you configure the system. When you are ready to put ClearDDTS into production, you must obtain permanent licenses from Rational. You can request licenses through our web site at:

http://www.rational.com/products/clear_ddts/support

For more information on licensing see Chapter 3, Managing Licenses.

Disk and Memory Requirements

The first step is to find appropriate disk space for ClearDDTS. As shipped, ClearDDTS requires approximately 10 megabytes of disk space for the base system, approximately 25 megabytes for its architecture-dependent binaries, and additional space for your defect records and temporary files. Therefore, you should plan on the following disk space requirements:

Disk Space	What it is used for
10MB	Base system
25MB	Each architecture's binaries
10MB	Temporary files
6500 bytes	Each defect record

If, for example, your environment supports both Solaris and HP platforms, you should reserve 70 megabytes (10 base + 25 solaris + 25 hp + 10 temporary), plus 6500 bytes per defect.

You should also consider memory requirements. The following table provides guidelines for configuring your system:

Requirement	Recommendation
Server memory	32MB minimum
Swap space	100MB minimum

CPU Requirements

We suggest that you put ClearDDTS on a server that the entire development group is currently using and let it share existing CPU and disk resources. Keep in mind, however, that database performance is dependent on the speed of the server. Therefore, you may want to install ClearDDTS on your fastest machine.

Preparing the Environment

Follow these steps to prepare for the installation of ClearDDTS.

Note: In the following procedures and examples, `~ddts` is used to refer to the ClearDDTS home directory.

- 1 Locate disk space for installing ClearDDTS. To check disk space, use a UNIX command such as `du` or `df`. For example:

```
$ df
Filesystem kbytes used avail capacity Mounted on
/dev/sd0a 29491 3946 22596 15% /
/dev/sd0g 200982 160905 19979 89% /usr
/dev/sd0h 233838 9512 200943 5% /home
```

You should put ClearDDTS on a file system this is accessible to all users via NFS (Network File System).

- 2 Log in as root.

```
$ su - root
Password: <root password>
```

- 3 Create a new group for ClearDDTS according to the procedures for your operating system. For example, edit the */etc/group* file to include a line similar to the following:

```
ddts:*:400:ddts
```

You now have a group called *ddts* with no group password, a group ID of 400, and only the user *ddts* as a member.

- 4 Create the *ddts* login account. Every ClearDDTS installation is owned by a special administrative user called *ddts*. ClearDDTS uses the home directory of this user to determine where the product is installed.

Using the proper procedures for your operating system, create the *ddts* user. For example, edit the */etc/passwd* file to include a line similar to the following:

```
ddts::400:400:ClearDDTS Owner:/usr/home/ddts:/bin/csh
```

For security reasons, the *ddts* user should have a password. You can use the *passwd* command to assign *ddts* a password now or after installation.

Note: On Solaris systems you also need to run the *pwconv* command.

- 5 Create a home directory for ClearDDTS. This is where the product files will reside. For example:

```
$ mkdir /usr/home/ddts
```

For this directory:

- Make *ddts* the owner and group associated with this directory.

```
$ chown ddts /usr/home/ddts
$ chgrp ddts /usr/home/ddts
```

- Make sure that all of the directories in the path to this directory are readable and executable (searchable) by everyone.

```
$ chmod 755 /usr/home/ddts
```

- Put any default files you normally give users in this directory. For example, most installations have generic *.cshrc*, *.login*, *.exrc*,

and *.profile* files. Be sure that these files belong to the user and group *ddts*.

Note: Do not add such files to any other ClearDDTS directories, as they may confuse daemons that search these directories.

- 6 Make sure the *ddts* user does not have an existing crontab entry. (A crontab entry will be created during installation.) To see if there is a crontab, log in as *ddts* and enter:

```
$ crontab -l
```

To remove a crontab, enter:

```
$ crontab -r
```

Adding aliases for the CGI scripts to your HTTP server configuration

The ClearDDTS installation will ask you to specify in which directory you want to install ClearDDTS. Within that directory will be the `www` directory and the `allbinaries` directory. The `www` directory contains the *webddts* CGI scripts, images, and HTML documentation, and the `allbinaries` directory will contain binary files users associate with defects. Because your HTTP server cannot run ClearDDTS unless it knows the location of these files, you must explicitly identify their location for your HTTP server.

The procedures for defining the location of scripts, images, HTML documentation, and binary attachments are different for different HTTP servers, so you need to check your server documentation for exact procedures. Most HTTP servers require you to make a separate entry for the location of scripts and a separate entry for the location of images and HTML files, even if all of these objects are located in the same directory. For each of these entries you need to specify the directory that contains the files and an alias or name for the directory.

We recommend you use the aliases listed in the following table for the location of the directories containing the ClearDDTS scripts, images, HTML documentation, and binary attachments. You can

use different aliases, but if you do, you must change *ddts_main.perl* to match.

Recommended Alias or ScriptAlias	Directory
<code>/ddts/</code>	ScriptAlias for the <code>www</code> directory containing the ClearDDTS scripts.
<code>/wt/</code>	Alias for the <code>www</code> directory containing the ClearDDTS images and HTML documentation.
<code>/all-bin/</code>	Alias for the <code>allbinaries</code> directory containing binary attachments. (For more information on using binary attachments see Chapter 6 in the ClearDDTS User's Guide.)

Below is an example of how to define the location of the ClearDDTS files for the *Apache* server. If you are using a different HTTP server or want more information on how to configure the HTTP server, see your server documentation.

Example of how to modify the Apache server configuration

To define the location of the ClearDDTS images, HTML documentation, and binary attachments area, add **Alias** directives to your *srm.conf* file. For example, if your *www* directory is defined as `/usr/home/ddts/www`, add the following directives to the *srm.conf* file:

```
Alias /wt/ /usr/home/ddts/www/  
Alias /all-bin/ /usr/home/ddts/allbinaries/
```

To define the location of the ClearDDTS scripts, add a **ScriptAlias** directive to your *srm.conf* file. For example, if your *www* directory is defined as `/usr/home/ddts/www` and you want to assign the alias `/ddts/` to this directory, add the following directive to the *srm.conf* file.

```
ScriptAlias /ddts/ /usr/home/ddts/www/
```

Note: If you choose to run native FastCGI you still need to create an alias for the ClearDDTS `www` and `allbinaries` directories, but

you do not have to create the `ScriptAlias` for the ClearDDTS CGI scripts. For more information on FastCGI see Chapter 4, Using FastCGI.

If you are using ClearDDTS with multiple connected sites that are not in the same domain, be sure to define your web server name using the domain. This allows users to access the URL to binary attachments stored on sites in other domains. To define the server name on an Apache server, set the `ServerName` variable in the `http.conf` file. For example:

```
ServerName <host>.<some_domain>.com
```

To apply the changes you just made apply the `kill -HUP` command to the current `httpd` process. This will force a reread of the configuration files and apply the changes just made.

Note: If you do not have HTTP connectivity between your connected sites (i.e., if there is a firewall), users at remote sites will not be able to access attachments created at the local site.

Extracting the software

- 1 Load the CD-ROM. Depending on your platform, you may need to mount the CD-ROM before continuing.

- 2 Switch to the user `ddts`:

```
$ su - ddts
```

- 3 Go to the ClearDDTS directory you created earlier. For example:

```
$ cd /usr/home/ddts
```

- 4 Extract the ClearDDTS files for your architecture(s) from the `tar` file. For example:

```
$ tar xvf /cdrom/<architecture_subdirectory>/clearddts.tar
```

If you are using multiple architectures, you should run the `tar` command separately for each architecture to extract all of the necessary tar files from the appropriate architecture subdirectories (for example, `solaris2` and `hpux`).

Within the current directory, all ClearDDTS product files should now reside in a directory called *NEWCLEARDDTS*.

Installing ClearDDTS

- 1 While logged in as *ddts*, go to the ClearDDTS home directory.

```
$ cd
```

- 2 Run the *ddtsinstall* installation script:

```
$ NEWCLEARDDTS/ddtsinstall
```

The *ddtsinstall* script installs ClearDDTS in the directory from which the script is run. Make sure you run the command from the ClearDDTS home directory (as shown in the example) and not by changing to the *NEWCLEARDDTS* directory.

Note: If your mail spool is running on a System V machine, you must run *ddtsinstall* as root. This allows ClearDDTS to add and correctly set permissions for the *mail.in* program. If you are running ClearDDTS on a System V machine, but NFS mount a mail spool not on a System V machine, you can run *ddtsinstall* as the *ddts* user.

In the following example, script output is shown in **Courier** font and sample responses are shown in **Bold Courier**. Some prompts have default responses (in brackets []). You can simply press RETURN with no input to accept the default.

```
*****
*
*           C l e a r D D T S   I n s t a l l a t i o n           *
*
*   (This installation script may be safely rerun at any time.)   *
*
*****

Installing ClearDDTS in /usr/home/ddts.

Is this correct? (y/n) [y]: y

Moving the files and directories in the installation directory
into their correct locations.

Do you want to install the ClearDDTS man pages in a common
directory? (y/n) [n]: n
```


Setting up appropriate file permissions. This will take a while...

Running adminbug inst to finish the installation...

Installing ClearDDTS on host <machine>.
This machine will be used as the ClearDDTS server.

Is this correct? (y/n): **y**

What are the mail address(es) of the ClearDDTS administrator(s)?

Mail address(es): **fred chris@piper**

What is the five character site identifier used to uniquely identify this site within the ClearDDTS network?

Site identifier (XXXyy): **SFOqa**

This is the user-defined site identifier used to uniquely identify the machine on the ClearDDTS network. The first three characters must be upper case alphabetic and the last two must be lower case alphabetic.

What is the name of your organization or division? This name is used in the Submitter Information section of all records submitted from this site.

Organization/division name: **Software Business Division**

What is the mail command to use on this machine? If a '%s' is supplied as part of the command, DDTs will substitute the subject of the mail in its place. Be sure to quote the '%s' so spaces in the subject will be handled correctly.

Mail command: **/usr/bin/mailx -s '%s'**

Because ClearDDTS networking depends on the mail system, you need to inform ClearDDTS how to send mail on your system. The default mail command varies depending on your architecture. Enter the full path name of the mailer program you use. If necessary, ask your system administrator for the appropriate mail program and invocation parameters to use.

Updating the site ID's in all of the projects...
Setting up the project help files...
Installation complete.

Running adminbug alic to enter your license...

Are you entering an evaluation license? (y/n): **Y**

Enter the information EXACTLY as it appears on your license certificate.

```
Hostname           : <machine>
Hostid             : ANY
Quantity          : 100
Expiration Date    : 31-Mar-1999
Password          : <password>
```

Adding the license...

Enter the Startup license key information shipped with ClearDDTS. You do not enter permanent licenses here. If there is an error in the license information or you enter N, ClearDDTS will warn you that entering the license was not successful, but will continue with the installation.

If you need to reenter the Startup license key information, run the *adminbug alic* command after the installation completes. Since a *license.dat* file will already exist (created during the first attempt), ClearDDTS will save the newly entered license information in a file called *ddts.lic*. You can delete the incorrect *license.dat* file and rename the *ddts.lic* file to *license.dat*.

If you have already received your permanent licenses, complete the installation, then follow the procedure for setting up permanent licensing in Chapter 3, Managing Licenses.

```
Running ddtsdbbuild to build the ClearDDTS database...
Going into maintenance mode...
Killing the ClearDDTS daemons...
ClearDDTS is in maintenance mode.
Building the ClearDDTS database on the internal database...
Creating table defects...done.
Creating table enclosures...done.
Creating table change history...done.
Loading defect records from /usr/home/ddts/allbugs/00...
.
.
.
Loading defect records from /usr/home/ddts/allbugs/99...
Exiting maintenance mode...
Starting the ClearDDTS daemons...
```

Rebuild of the ClearDDTS pure database completed successfully.

```
Setting up cron to run ddtsd and ddtsclean...
warning: commands will be executed using /usr/bin/sh
```

The *ddtsd* and *ddtsclean* programs must be running for ClearDDTS to operate properly. The installation script automatically puts them in the *ddts* user's *crontab* file, with *ddtsd*

set to run every 15 minutes and *ddtsclean* once a day. Removal of these programs will result in incorrect operation of your ClearDDTS system.

The *ddts* user's *crontab* file must be on the server where the ClearDDTS binaries are installed (the host whose name is in *~ddts/conf/hostname*). Do not put the *crontab* file on another NFS attached system. An online example is supplied in *~ddts/etc/crontab.sample*.

Configuring the web interface in:

```
/usr/home/ddts/www
```

You are now ready to continue the installation steps as described in the installation guide. Don't forget to:

1. start the FLEXlm license manager,
2. add the appropriate aliases to your HTTP config files for scripts, images, and binary attachments, and
3. log into the web interface at `http://<webhost>/ddts/ddts_main` to finish its configuration automatically.

Please refer to the installation guide for details.

- 3 After running the installation script, you can unmount and eject the CD-ROM using the commands appropriate to your platform.

Note: Include the path to the ClearDDTS binaries, *~ddts/bin*, in the *.cshrc* file (or appropriate login script) for user *ddts*. Specific instructions for editing the path statement are available in *Making ClearDDTS Available to UNIX Users* on page 15.

Starting an evaluation license manager

After running *ddtsinstall* and entering the Startup license key information you must start a temporary license manager. To manage licenses, ClearDDTS uses FLEXlm 6.0a, from Globetrotter Software, Inc. The Startup license information you entered during installation is written to the file *license.dat* in the

ClearDDTS home directory. You use this file to start your FLEXlm license manager.

Note: See Chapter 3, *Managing Licenses*, for information on setting up permanent licenses.

To start a new license manager do the following:

- 1 Switch to the user *ddts*:

```
$ su - ddts
```

- 2 Enter the following commands:

```
$ bin/lmgrd -c ./license.dat -l <log_file> &
```

The license manager daemon (*lmgrd*) reads the *license.dat* file to get start-up information about the vendor daemon for ClearDDTS and starts the vendor daemon.

Specifying a log file for *lmgrd* is optional but recommended since it helps Rational technical support to debug license errors. The log file keeps debug messages from appearing on screen, and tracks the checkin and checkout of each license. You can keep the log file anywhere, but we suggest you keep it at the top level of your ClearDDTS home directory.

Running the webddts interface

After adding the location of the ClearDDTS CGI scripts to your HTTP server configuration, running the installation, and starting a license manager you are ready to launch the *webddts* interface. To do this point your browser to:

```
http://<host>/ddts/ddts_main
```

where *<host>* is your HTTP server.

ClearDDTS automatically completes the web configuration. Just follow the directions on the screen to reload the page.

If you want to run your web server on a different machine than your ClearDDTS server, make sure that:

- Your web server machine has access to the ddts home directory on the ClearDDTS server machine (for example, via NFS)
- You installed on your ClearDDTS server the ClearDDTS binaries for the operating system running on your web server. For example, if your ClearDDTS server is running HP-UX and your web server is running Solaris, your ClearDDTS server must contain binaries for both architectures.
- The ddts user is listed in the password file of both the web server machine and the ddts server machine and the user ID and group ID of the two entries match.

If you meet these requirements, follow these instructions to configure your web server:

- 1 Log into the web server machine as user ddts.
- 2 Change to the ddts home directory.
- 3 Run *bin/webconfigure*.

After you run *webconfigure*, the environment of your web server will match the paths on the ClearDDTS server.

Selecting an External Oracle Database

Although ClearDDTS includes its own internal SQL database, you have the option to use an Oracle database. Before deploying ClearDDTS to your users, decide which database you want to use. For information on using ClearDDTS with an external Oracle database see Appendix G, Using an Oracle Database, of the *ClearDDTS Administrator's Guide*.

Note: You can switch databases after you have completed the installation procedure. However, you will have to take ClearDDTS offline while switching databases if you change after deployment.

Checking the Installation and Informing New Users

After installation you may need to export the ClearDDTS home directory so that others can access ClearDDTS. If your ClearDDTS home directory is not already exported, use the procedures for your environment to export the directory. For example, on BSD systems, enter the full directory path in the */etc/exports* file, then export it by using the *exportfs(8)* command:

```
$ exportfs -a
```

On SVR4, see the *share(1M)* man page; on SVR3, see *exportfs(1M)*.

Note that NFS administration is different on different vendors' systems, so you may need the help of your local system administrator to perform this step.

Mounting the ClearDDTS Home Directory

After installation, users must mount the ClearDDTS home directory to make sure that their client workstations can access the system. For example, if you have a BSD-style system, add an entry to the */etc/fstab* file in the form of:

```
<hostname>:/usr/home/ddts /usr/home/ddts nfs rw,bg,soft 0 0
```

Then mount the system by using *mount(8)*, like this:

```
$ mount /usr/home/ddts
```

Note: Be sure that the ClearDDTS file system is *not* mounted with the *nosuid* option; several ClearDDTS programs will not work when this option is used. For information about how to check your mount options, see your system administrator or the *mount(8)* man page.

Once you have verified that client workstations can mount the exported directory on their local systems, you are ready to make ClearDDTS available to the user community.

Making ClearDDTS Available to Web Users

To access the web interface, have users point their web browsers to:

```
http://<host>/ddts/ddts_main
```

where <host> is your HTTP server.

Making ClearDDTS Available to UNIX Users

Each person using one of the ClearDDTS UNIX interfaces (*xddts*, *findbug*, etc.) needs to set some variables in his or her own login scripts to gain access to ClearDDTS. To do this, users need to do the following:

- 1 Check whether you can access the ClearDDTS home directory. For example:

```
$ cd /usr/home/ddts
```

- 2 Add *~ddts/bin* to your path. If you plan to use ClearDDTS with more than one architecture, you need to determine which architecture you are running and then set up the PATH accordingly. For example, if you have Solaris and HP-UX and use the *C shell* you would add the following in your *.cshrc* file:

```
set my_bin = `~ddts/bin/ddts_archdir.sh`  
set path = ($path ~ddts/$my_bin)  
rehash
```

In the *Bourne* or *Korn* shell, you would add the following to the *.profile* file:

```
my_bin=`/usr/home/ddts/bin/ddts_archdir.sh`  
PATH=$PATH:/usr/home/ddts/$my_bin; export PATH
```

This allows you to log in to any system and have the PATH environment variable configured properly. If you are using ClearDDTS with only one hardware architecture, the procedure is somewhat simpler. For the C-shell, you can set the path by adding *~ddts/bin* to your current path:

```
$ set path=($path ~ddts/bin )  
$ rehash
```

If you are using the Bourne or Korn shell, you would specify something like this:

```
$ PATH=PATH:/usr/home/ddts/bin
$ export PATH
```

- 3 Set up the MANPATH environment variable. If ClearDDTS man pages were installed with the installation script, you can skip this section. Otherwise, add `~ddts/doc/man` to the MANPATH variable.

In the C-shell, add the following to your `.cshrc` file:

```
setenv MANPATH ~ddts/doc/man:$MANPATH
```

In the Bourne shell, add the following to your `.profile` file:

```
MANPATH=$MANPATH:/usr/home/ddts/doc/man; export MANPATH
```

Sending E-Mail to the User Community

Now that you understand how to configure ClearDDTS for client work stations, you should send e-mail to your user community describing how to:

- Set the PATH environment variable.
- Mount the ClearDDTS home directory.
- Configure the MANPATH environment variable.

2

Upgrading from a Previous Release

If you are upgrading from a previous release of ClearDDTS (release 3.1 or later), the procedure is slightly different from the procedure used for a new installation. Before you begin the upgrade, read the Release Notes and the Changes file to become familiar with what has changed in this release.

The following steps summarize the upgrade procedure. Each section of this chapter describes a step in detail:

- 1 If applicable review the information in "Upgrading from releases prior to 4.1" on page 2-2
- 2 Back up your current installation of ClearDDTS. See "Backing up your current installation" on page 2-5.
- 3 Extract the ClearDDTS files from the upgrade media. See "Extracting the software" on page 2-5.
- 4 Read the Changes file and Release Notes. See "Starting an evaluation license manager" on page 2-11.
- 5 Run the upgrade script. See "Upgrading ClearDDTS" on page 2-6.
- 6 If applicable, start a FLEXlm license manager to use with your Startup license key. See "Starting an evaluation license manager" on page 2-11.
- 7 Run the *webddts* interface to complete the web configuration. See "Running the webddts interface" on page 2-12.

Upgrading from releases prior to 4.1

Before upgrading from releases prior to 4.1 you must be aware of the following:

- releases 4.1 and beyond require a new ClearDDTS license using FLEXlm
- your HTTP server configuration must be modified to locate the new ClearDDTS CGI scripts

Licensing with FLEXlm

As of release 4.1, ClearDDTS uses FLEXlm to manage its licenses. This replaces the PureLA license advisor. In order to use the FLEXlm license manager you must have a new license password and you must set up the FLEXlm license manager.

If you are upgrading from a release prior to 4.1 you should have received a Startup license key with your upgrade packet. The Startup license key is an evaluation license that works on any host and expires after approximately one month. It allows you to get ClearDDTS up and running while you configure the system.

You can obtain permanent licenses by contacting Rational with the hostname and hostid of the machine that will be your license server. Rational will then generate a password specifically for that hostid and the number of licenses you purchased. You can request licenses through our web site at:

http://www.rational.com/products/clear_ddts/support

For information on setting up permanent ClearDDTS licensing with FLEXlm see Chapter 3, Managing Licenses.

Adding aliases for the CGI scripts to your HTTP server configuration

After upgrading, the ClearDDTS directory will contain the `www` directory and the `allbinaries` directory. The `www` directory contains the `webddts` CGI scripts, images, and HTML

documentation, and the `allbinaries` directory will contain binary files users associate with defects. Because your HTTP server cannot run ClearDDTS unless it knows the location of these files, you must explicitly identify their location for your HTTP server.

The procedures for defining the location of scripts, images, HTML documentation, and binary attachments are different for different HTTP servers, so you need to check your server documentation for exact procedures. Most HTTP servers require you to make a separate entry for the location of scripts and a separate entry for the location of images and HTML files, even if all of these objects are located in the same directory. For each of these entries you need to specify the directory that contains the files and an alias or name for the directory.

We recommend you use the aliases listed in the following table for the location of the directories containing the ClearDDTS scripts, images, HTML documentation, and binary attachments. You can use different aliases, but if you do, you must change `ddts_main.perl` to match.

Recommended Alias or ScriptAlias	Directory
<code>/ddts/</code>	ScriptAlias for the <code>www</code> directory containing the ClearDDTS scripts.
<code>/wt/</code>	Alias for the <code>www</code> directory containing the ClearDDTS images and HTML documentation.
<code>/all-bin/</code>	Alias for the <code>allbinaries</code> directory containing binary attachments. (For more information on using binary attachments see Chapter 6 in the ClearDDTS User's Guide.)

Below is an example of how to define the location of the ClearDDTS files for the *Apache* server. If you are using a different HTTP server or want more information on how to configure the HTTP server, see your server documentation.

Example of how to modify the Apache server configuration

To define the location of the ClearDDTS images, HTML documentation, and binary attachments area, add `Alias` directives to your `srm.conf` file. For example, if your `www` directory is defined as `/usr/home/ddts/www`, add the following directives to the `srm.conf` file:

```
Alias /wt/ /usr/home/ddts/www/  
Alias /all-bin/ /usr/home/ddts/allbinaries/
```

To define the location of the ClearDDTS scripts, add a `ScriptAlias` directive to your `srm.conf` file. For example, if your `www` directory is defined as `/usr/home/ddts/www` and you want to assign the alias `/ddts/` to this directory, add the following directive to the `srm.conf` file.

```
ScriptAlias /ddts/ /usr/home/ddts/www/
```

Note: If you choose to run native FastCGI you still need to create an alias for the ClearDDTS `www` and `allbinaries` directories, but you do not have to create the `ScriptAlias` for the ClearDDTS CGI scripts. For more information on FastCGI see Chapter 4, Using FastCGI.

If you are using ClearDDTS with multiple connected sites that are not in the same domain, be sure to define your web server name using the domain. This allows users to access the URL to binary attachments stored on sites in other domains. To define the server name on an Apache server, set the `ServerName` variable in the `http.conf` file. For example:

```
ServerName <host>.<some_domain>.com
```

To apply the changes you just made apply the `kill -HUP` command to the current `httpd` process. This will force a reread of the configuration files and apply the changes just made.

Backing up your current installation

- 1 Log in as *ddts*.
- 2 Back up your current ClearDDTS installation. For example, use *tar* to back up your installation to tape.

```
$ cd /usr/home/ddts
$ tar cvf /dev/xxx (where xxx is the name of the tape device)
```

You should also check that the backup tape is readable before continuing. You can do this by running *tar -tvf /dev/xxx*.

Extracting the software

Follow these steps to prepare for the ClearDDTS upgrade.

Note: In the following procedures and examples, *~ddts* is used to refer to the ClearDDTS home directory.

- 1 Log in as *ddts*.
- 2 In the *ddts* home directory, make sure there is not an existing *NEWCLEARDDTS* directory. If there is, either rename or delete it.
- 3 While still in the *ddts* home directory, extract the new release from the installation media. For example:

```
$ tar xvf /cdrom/<OS_subdirectory>/clearddts.tar
```

If you are using multiple architectures, you should run the *tar* command separately for each architecture to extract all of the necessary *tar* files from the appropriate architecture subdirectories (for example, *solaris2* and *hpux*).

Your ClearDDTS home directory should now contain the directory *NEWCLEARDDTS*.

Determining changes to implement

Most customization of ClearDDTS occurs in the class directories. To preserve your customizations, ClearDDTS does not automatically overwrite or add anything to the class directories during an upgrade. Therefore, ClearDDTS cannot always install new or upgraded versions of some features.

In order to learn what the new features and updates are, you must read the `RELEASE_NOTES` and `CHANGES` files located in the top level of the ClearDDTS home directory. The `CHANGES` file lists all changes and gives code examples of some new features and customizations. You can use that information to merge the changes into your ClearDDTS installation.

Some existing features may behave differently if you do not implement certain changes, therefore it is important to understand what has changed in the current release. You can then determine if you want to take advantage of some of the new or updated features.

Upgrading ClearDDTS

In some cases, performing the upgrade requires a rebuild of the database (done automatically as part of the upgrade). Be sure to plan for a database rebuild when determining the best time to perform the upgrade.

- 1 Log in as *ddts*.
- 2 Run the *ddtsupgrade* script to install the product files. From the ClearDDTS home directory enter:

```
$ NEWCLEARDDTS/ddtsupgrade
```

The *ddtsupgrade* script updates ClearDDTS in the directory from which the script is run. Make sure you run the command from the ClearDDTS home directory (as shown in the example) and not by changing to the *NEWCLEARDDTS* directory.

In the following example, script output is shown in Courier font and sample responses are shown in Bold Courier. Some prompts have default responses (in brackets []). You can simply press RETURN with no input to accept the default.

```
*****
*
*           C l e a r D D T S   U p g r a d e           *
*
*****
This script is about to upgrade your current ClearDDTS system. For the
most part this will simply replace outdated files with the newer versions
However, because both the customer and Rational Software may have made
changes to certain configurable files, part of the upgrade will be to
compare the current version with the new one. If any differences are
detected then you will be prompted to replace the current version with
the new version. Note that you may need to merge our changes into your
customized files in some cases. This will require manual merging after
the upgrade script has completed.

You will be asked a series of questions during this process. The script
is interactive and fairly self-explanatory. If you choose to keep the
current version of a file, the new version will be retained in
NEWCLEARDDTS/for later reference. If you choose to replace with the new
version, a copy of the old version will be stored in OLDCLEARDDTS/. Both
directories are kept in the current directory.

Although this script will automatically keep backups for you, some
files will simply be replaced. These include files in doc, and all
compiled executables in *_bin. It is STRONGLY recommended that you make
a tape backup of your current ClearDDTS installation before proceeding.

Upgrading ClearDDTS in /usr/home/ddts.

Is this the correct directory? (y/n) [y]: y

Upgrading ClearDDTS will overwrite your existing installation of DDTS.

Have you made a backup copy of your current installation? (y/n) [n]: y

Upgrading 4.0 to 4.5.

Is this correct? (y/n) [y]: y

Performing the upgrade...

Going into maintenance mode...
Killing the ClearDDTS daemons...
Waiting for ddtsd[14509] to terminate...Done.
ClearDDTS is in maintenance mode.

Backing up the current bin directories that aren't going to
be upgraded...
Upgrading .ddtsrc...
  Nothing to upgrade.
Upgrading BUGFIXES...
  Done.
```

```
Upgrading README...
  Done.
```

```
.
.
.
Upgrading doc...
  Done.
```

```
  Do you want to install the new ClearDDTS man pages in a common
  directory? (y/n) [n]:N
Upgrading etc...
```

For each file that has changed you can choose to view the differences, replace the current file with the new file, or keep the current version. For example:

```
etc/Browser is different in the new version.
Do you want to:
  V)iew the changes (diff -w -c ...)
  R)eplace the current version with the new version
  K)eep the current version
Choice (v/r/k): r
```

```
.
.
.
```

In most cases, you will want to replace the current version, but if you have customized your installation, you should review the differences carefully before deciding whether to keep or replace your current version. When you replace a file, the old version is stored temporarily in the *OLDCLEARDDTS* directory.

```
.
.
.
Upgrading local...
  Nothing to upgrade.
Upgrading proj.info...
  Nothing to upgrade.
Upgrading projects...
  Nothing to upgrade.
.
.
.
```

During the upgrade saved queries are automatically converted to the current format. If a saved query does not convert, you will see a warning message. If a saved query does not convert, it will have an asterisk next to its name on the Saved Queries page. Queries that did not successfully convert can still be run, but they cannot be edited.


```

Cleaning up sol2x_bin...
Cleaning up spool...
Updating file permissions. This may take a while...

Running adminbug alic to enter your license...

Are you entering an evaluation license? (y/n): Y

Enter the information EXACTLY as it appears on your license certificate.
  Hostname           : <machine>
  Hostid             : ANY
  Quantity           : 100
  Expiration Date    : 31-Mar-1999
  Password           : <password>

Adding the license...

```

If you are upgrading from a release prior to 4.1 you are prompted for license information. Otherwise, ClearDDTS recognizes your existing FLEXlm licenses, and no licensing prompts are necessary.

Enter the Startup license key information shipped with ClearDDTS. You do not enter permanent licenses here. If there is an error in the license information or you enter N, ClearDDTS will warn you that entering the license was not successful, but will continue with the upgrade.

If you need to reenter the Startup license key information, run the *adminbug alic* command after the installation completes. Since a *license.dat* file will already exist (created during the first attempt), ClearDDTS will save the newly entered license information in a file called *ddts.lic*. You can delete the incorrect *license.dat* file and rename the *ddts.lic* file to *license.dat*.

If you have already received your permanent licenses, complete the upgrade, then follow the procedure for setting up permanent licensing in Chapter 3, Managing Licenses.

```

Configuring the web interface in:
    /usr/home/ddts

Running ddtsdbbuild to rebuild the ClearDDTS database...
Going into maintenance mode...
Killing the clearDDTS daemons...
ClearDDTS is in maintenance mode.
Building the ClearDDTS database on the internal database...
Creating table defects...done.
Creating table enclosures...done.
Creating table change_history...done.
Loading defect records from /usr/home/ddts/allbugs/00...

```

```

.
.
.
Loading defect records from /usr/home/ddts/allbugs/99...
Exiting maintenance mode...
Starting the ClearDDTS daemons...

Rebuild of the ClearDDTS pure database completed successfully.

Upgrade script complete.

You are now ready to continue the upgrade steps as described in the
installation guide. Don't forget to:

1. start the FLEXlm license manager,

2. add the appropriate aliases to your HTTP config files for scripts,
   images, and binary attachments, and

3. log into the web interface at http://<webhost>/ddts/ddts_main
   to finish its configuration automatically.

Please refer to the installation guide for details.

***** Reminder*****
*Once everything is working correctly you will want to remove the install*
*directory NEWCLEARDDTS and the backup directory OLDCLEARDDTS to      *
*conserve disk space.                                                *
*                                                                       *
*Please refer to the RELEASE_NOTES for changes to include in your new  *
*release, specifically to your master.tpl file.                        *
*****

```

- 3 Review the files in the *OLDCLEARDDTS* and *NEWCLEARDDTS* directories and merge your customizations into the new system as appropriate.

The files remaining in the *NEWCLEARDDTS* directory are those normally highly customized by users (for example, class specific *master.tpl* files). We do not move these files during the upgrade to avoid overwriting your customizations. You should review these files for differences and recommendations from Rational Software, and manually add these to your installation as necessary. See the *CHANGES* file in the top level of the ClearDDTS home directory for information on what has changed in the release.

- 4 Set up ClearDDTS licensing using FLEXlm. See "Starting an evaluation license manager" on page 2-11, for information on setting up an evaluation license manager. See Chapter 3, *Managing Licenses*, for information on setting up permanent licensing.

- 5 Test the new version of ClearDDTS to make sure it works as expected. Once you are satisfied that the system is working properly, you can delete the *OLDCLEARDDTS* and *NEWCLEARDDTS* directories to reclaim disk space.

Starting an evaluation license manager

If you are upgrading from release 4.1 or higher your licensing should already be set up, therefore you can skip this section. ClearDDTS will use your existing FLEXlm license server during the upgrade. If you are upgrading from a release prior to 4.1, continue with this section.

After running *ddtsupgrade* and entering the Startup license key information you must start a temporary license manager. To manage licenses, ClearDDTS uses FLEXlm 6.0a, from Globetrotter Software, Inc. The Startup license information you entered during the upgrade is written to the file *license.dat* in the ClearDDTS home directory. You use this file to start your FLEXlm license manager.

Note: See Chapter 3, Managing Licenses, for information on setting up permanent licensing.

To start a new license manager do the following:

- 1 Switch to the user *ddts*:

```
$ su - ddts
```

- 2 Enter the following commands:

```
$ bin/lmgrd -c ./license.dat -l <log_file> &
```

The license manager daemon (*lmgrd*) reads the *license.dat* file to get start-up information about the vendor daemon for ClearDDTS and starts the vendor daemon.

Specifying a log file for *lmgrd* is optional but recommended since it helps Rational technical support to debug license errors. The log file keeps debug messages from appearing on screen, and tracks the checkin and checkout of each license. You can keep the log file

anywhere, but we suggest you keep it at the top level of your ClearDDTS home directory.

Running the webddts interface

After adding the location of the ClearDDTS CGI scripts to your HTTP server configuration and running the upgrade, you are ready to launch the *webddts* interface. To do this point your browser to:

```
http://<host>/ddts/ddts_main
```

where <host> is your HTTP server.

ClearDDTS automatically completes the web configuration. Just follow the directions on the screen to reload the page.

3

Managing Licenses

This chapter contains a summary of the license manager features and procedures for setting up permanent licensing. Topics covered include:

- How licenses work
- The license manager
- The license file
- License manager commands
- Setting up permanent licensing

How licenses work

ClearDDTS uses two types of licenses:

- *floating licenses* (also called concurrent or network): Floating licenses allow anyone on your network to use any ClearDDTS interface as long as a license is available. The number of licenses you purchase determines the maximum number of users who can use ClearDDTS simultaneously. You must have floating licenses to use the *webddts* interface.
- *node-locked licenses*: Node-locked licenses are tied to a specific machine and host ID, and are used by the *xddts* and *bugs* interfaces only. Node-locked licenses are not part of the pool of floating licenses.

ClearDDTS checks out a license when you perform a licensed activity (see *What is licensed* on page 3-2). You retain a license for 20 minutes after the beginning of your last licensed action. If you do not perform another licensed activity within 20 minutes the license is returned to the license manager and becomes available for another user.

When you buy ClearDDTS, you purchase some number of floating and/or node licenses. To manage licenses, ClearDDTS uses FLEXlm 6.0a, from Globetrotter Software, Inc. The FLEXlm license manager daemon runs on a

previously determined machine to track and make available ClearDDTS licenses. This machine is called the license server.

What is licensed

A license is required by any ClearDDTS user who modifies defects, runs queries, or generates management reports using any of the ClearDDTS interfaces. A user does not take up a license when submitting a defect, or running the command line utilities.

Checks out a license	Does not check out a license
<ul style="list-style-type: none">o Modifying defects<ul style="list-style-type: none">- changing states- adding, editing, or deleting enclosures- creating links- cloningo Running querieso Generating Reports	<ul style="list-style-type: none">o Submitting defectso Viewing a single defecto Command line utilities:<ul style="list-style-type: none">- adminbug - dumpbug- ddtssql - findbug- batchbug - bugval

When you start *webddts*, you are initially unlicensed. You do not check out a license until you perform one of the licensed activities listed above. If a license is available, the license manager gives you a license. If no license is available, you are unable to use licensed features until a license is returned by another user.

When you start *xddts*, the first activity performed is a query to load the index, therefore you do need a license. To start *xddts* without initially using a license, run one of the following options:

- `xddts -c`
Starts *xddts* and immediately enters submit mode.
- `xddts -i <defect_ID>`
Starts *xddts* and immediately displays the defect whose ID you enter on the command line.

Floating licenses are tracked based on username only. If a user runs ClearDDTS from two separate locations, or two different interfaces, no additional license is required.

The license manager

The license manager includes the following components:

- A vendor daemon named *rational* that dispenses the licenses.
The *rational* vendor daemon is used for all of Rational's licensed products. If you have other products from other vendors that also use FLEXlm, they will include their own vendor daemons.
- A license manager daemon named *lmgrd*.
The same license manager daemon can be used by all licensed products from all vendors using FLEXlm (you can have a different FLEXlm installation for each if you choose). The *lmgrd* daemon does not process license requests on its own, but forwards requests to the appropriate vendor daemon.
- A license file, *license.dat*.
The license file specifies your license servers, vendor daemons, and product restrictions. The *license.dat* file is sent to you by Rational when you order permanent licenses.

The license file

ClearDDTS looks for the *license.dat* file in the ClearDDTS home directory. If it does not find it there, it looks in the location defined in the environment variable LM_LICENSE_FILE. If a *license.dat* file is present in the ClearDDTS home directory, that file is used regardless of how LM_LICENSE_FILE is set.

The *license.dat* file is a text file you can edit with any text editor. In general, it contains one to three server lines (beginning with the keyword SERVER), followed by one or more vendor daemon lines (beginning with the either keyword VENDOR or DAEMON), followed by one or more product feature lines (beginning with the keyword FEATURE or INCREMENT).

To merge the ClearDDTS license file with an existing license file you copy the VENDOR/DAEMON and FEATURE/INCREMENT lines to your current master *license.dat* file and put a symbolic link to that file in the ClearDDTS home directory. See *Setting up permanent licensing* on page 3-5, for more information on setting up FLEXlm with ClearDDTS.

License manager commands

The license manager supports several system administration commands you can use to maintain licenses:

Command	Description
<code>lmgrd</code>	The main daemon program for FLEXlm
<code>lmhostid</code>	Displays the host ID of the machine on which it is run
<code>lmstat</code>	Reports status on daemons and feature usage
<code>lmdown</code>	Shuts down the license and vendor daemons
<code>lmreread</code>	Rereads the license file and starts new vendor daemons

These utilities are located in the `~ddts/bin` directory for each platform you have installed. For more information you can refer to the man pages for each utility.

lmgrd

This is the licensing daemon. Use this command to initialize the FLEXlm system. The licensing daemon reads the `license.dat` file to get start-up information about the vendor daemons, then starts each of them in turn. On UNIX systems, it is strongly recommended that the process be run as a non-privileged user, instead of root. For example:

```
$ lmgrd -c ./license.dat -l log_file &
```

Adding a log file is optional but recommended. The log file keeps debug messages from appearing on screen, and tracks each checkin and checkout of a license. This is a useful tool for technical support. You can keep the log file anywhere, but we suggest you keep it at the top level of your ClearDDTS home directory.

lmhostid

This command displays the host ID of the machine on which it is run. Run this command on the machine that will be your license server. For example:

```
$ lmhostid
```

The output looks like:


```
lmhostid - Copyright (C) 1989-1997 Globetrotter Software, Inc.
The FLEXlm host ID of this machine is "1234c5cc"
```

Provide this host ID to Rational when ordering licenses.

lmstat

This command reports on the status of the license server, vendor daemons, feature usage, and current users. For example:

```
$ lmstat -c license.dat -f ddts
```

The output looks like:

```
lmstat - Copyright (C) 1989-1997 Globetrotter Software, Inc.
Flexible License Manager status on Wed 12/31/1997 11:33
Users of ddts: (Total of 10 licenses available)
    "ddts" v4.1, vendor: demo
        floating license
chris banana /dev/tty (v4.1) (pumpkin/27000 385), start Wed
12/31 3:15 (linger: 1800)
fred eclipse /dev/pts/1 (v4.1) (pumpkin/27000 678), start Wed
12/31 11:08 (linger: 1800)
```

lmdown

This process causes a graceful shutdown of all of the daemons (both *lmgrd* and all of the individual vendor daemons) on all nodes. For example:

```
$ lmdown -c license.dat
```

Using the UNIX `kill` command is not recommended as this shuts down *lmgrd* but leaves the vendor daemons running.

lmreread

This process is used to tell FLEXlm to reread and reevaluate the license file. This is usually used when changes are made to the license file while the FLEXlm system is running. Any changes you make in the file do not go into effect until the reread command is run. For example,

```
$ lmreread -c license.dat
```

Setting up permanent licensing

This section describes how to obtain your licenses and set up the license manager.

Note: The following procedures are for standard FLEXlm configurations. If you have specific questions about FLEXlm or want to implement a more complex configuration we encourage you to visit the Globetrotter Software, Inc. web site at:

<http://www.globetrotter.com/home.htm>

Obtaining licenses

Before requesting permanent licenses you must do the following:

- purchase the licenses through your Rational sales representative
- determine the machine you will use as the license server
- use the *lmhostid* command to obtain the host ID for that machine (The license manager daemon will run only on the machine whose host ID you provide to Rational.)

When you have all the required information, use Rational's license key generation web site at:

http://www.rational.com/products/clear_ddts/support

Complete the information requested being sure to correctly enter the host ID. Rational uses this information to generate your permanent license key and create a *license.dat* file. The *license.dat* file is sent to you in an e-mail message along with directions within 24 hours.

You can use this *license.dat* file to start your license manager. If you already have a FLEXlm license manager for other applications, you can choose to merge the *license.dat* files and run one license manager daemon for all applications licensed with FLEXlm. See below for details on setting up these configurations.

Removing your evaluation license setup

Before installing permanent licenses you must shutdown the evaluation license manager and remove the *license.dat* file. To do this:

- 1 Switch to the user *ddts*:

```
$ su - ddts
```

- 2 In the ClearDDTS home directory enter the following command to shut down the license server:

```
$ bin/lmdown -c ./license.dat
```

- 3 Delete the evaluation *license.dat* file:

```
$ rm license.dat
```

Once the license manager is shut down and the *license.dat* file is moved you are ready to set up permanent licensing.

Starting a new FLEXlm license manager

The following steps for starting the license manager assume that the license server platform is the same as the platform on which you have installed ClearDDTS. If this is not the case at your site, please substitute the bin directory for the appropriate platform when following the example below.

To start a new license manager, have available the e-mail message from Rational containing the *license.dat* information, and do the following:

- 1 Log into your license server host machine.

Note: The manufacturer of FLEXlm, (Globetrotter Software, Inc.) recommends that the process be run as a non-privileged user, instead of root.

- 2 Change to the ClearDDTS home directory:

```
$ cd ~ddts
```

- 3 Create a *license.dat* file using your preferred editor, and copy the SERVER, VENDOR or DAEMON, and FEATURE or INCREMENT lines from the e-mail message you received from Rational. Follow any additional instructions contained in the e-mail from Rational on how to create your *license.dat* file.

- 4 Save the file.

- 5 Enter the following command to start the license manager:

```
$ bin/lmgrd -c ./license.dat -l <log_file> &
```

The license manager daemon (*lmgrd*) reads the *license.dat* file to get start-up information about the vendor daemon for ClearDDTS and starts the vendor daemon.

Specifying a log file for *lmgrd* is optional but recommended since it helps Rational technical support to debug license errors. The log file keeps debug messages from appearing on screen, and tracks the checkin and checkout of each license. You can keep the log file anywhere, but we suggest you keep it at the top level of your ClearDDTS home directory.

Using an existing FLEXlm license manager

If you want to use the FLEXlm license manager daemon currently running on your existing license server to manage ClearDDTS licenses, you need to merge the information in your ClearDDTS *license.dat* with that in the master license file used by your existing license manager.

To use your existing FLEXlm license manager:

- 1 Log on to your license server.
- 2 Determine the version of your license manager daemon by entering the following command:

```
$ lmgrd -v
```

The output looks like:

```
lmgrd v6.0a - Copyright 1988-1997, Globetrotter Software, Inc.
```

When combining license files from multiple vendors you **must** use the latest version of *lmgrd*. FLEXlm correctly supports older versions of vendor daemons and applications, therefore, it is always safe to use the latest version of *lmgrd*.

When comparing version numbers you only need to compare the main version number (for example, 6.0), and not the letters that may follow it.

- 3 If you already have a Rational product installed, your master *license.dat* file will already have a Rational VENDOR or DAEMON line. Determine the version of your existing Rational vendor daemon by entering the following command:

```
$rational -v
```

The output looks like:

```
lmgrd v6.0a - Copyright 1988-1997, Globetrotter Software, Inc.
```

All Rational products use the same vendor daemon. When combining license files, you **must** use the latest version of the Rational vendor daemon.

- 4 Merge the ClearDDTS license information into your master *license.dat* file.

Copy the `VENDOR/DAEMON` and `FEATURE/INCREMENT` lines (but not the `SERVER` line) from the ClearDDTS *license.dat* file (contained in an e-mail from Rational) into your current master *license.dat* file. Follow any additional instructions contained in the e-mail from Rational on how to merge your *license.dat* file.

If you already have a Rational vendor daemon that is equal to or newer than the ClearDDTS version you do not need to copy the `VENDOR/DAEMON` line. If your current Rational vendor daemon is older than the version shipped with ClearDDTS, replace your existing `VENDOR/DAEMON` line with the newer version.

- 5 If you have a ClearDDTS *license.dat* file in the ClearDDTS home directory, delete it.
- 6 In the ClearDDTS home directory, use the `ln -s` command to create a symbolic link to the master *license.dat* file.

The *webddts* interface runs as the user *http*. Since the *http* user does not have the `LM_LICENSE_FILE` environment variable set, you must link to the *license.dat* file from the ClearDDTS home directory to enable ClearDDTS to find the file.

- 7 Reread the license file using the *lmreread* command. For example:

```
$ cd ~ddts
$ bin/lmreread -c license.dat
```

Note: If the license manager daemon shipped with ClearDDTS is newer than your existing version, you must shut down your current *lmgrd* (use *lmdown*) and start up again with the new version.

Restarting lmgrd automatically

You can use a UNIX system startup file to automatically restart the FLEXlm license manager daemon, *lmgrd*, after a reboot. Startup files vary from UNIX architecture to architecture; consult your system's documentation for details (on Solaris, see the man page for *init.d*).

The following example shows how you could create a startup file on Solaris.

- 1 Create the file *DDTS* in */etc/init.d/* consisting of:

```
#!/bin/sh
# /etc/init.d/DDTS - Start up the ClearDDTS license daemon

# First, get the DDTS home directory (this bit copied from ddtsclean)...
DDTSHOME=`/bin/echo `echo ~ddts` | /bin/csh -sf `
if test $? != 0; then
    echo "ERROR: Can't start lmgrd; cannot locate home directory..."
    exit 1
fi

# Start the license manager...
if [ -f $DDTSHOME/bin/lmgrd ]; then
    echo "Starting FLEXlm daemon for DDTS..."
    su ddts -c "$DDTSHOME/bin/lmgrd -c $DDTSHOME/license.dat -l
$DDTSHOME/license.log"
fi
```

- 2 Create a symbolic link to this file in */etc/rc3.d*:

```
cd /etc/rc3.d
ln -s /etc/init.d/DDTS S80DDTS
```

Upon reboot the system runs this script.

Adding additional licenses

To add additional licenses:

- 1 Access the Rational license key generation web page at:
http://www.rational.com/products/clear_ddts/support
- 2 Copy and paste the contents of your existing *license.dat* file in the text box supplied.
- 3 Fill out all other required information.

Rational will evaluate your *license.dat* file and either:

- issue an INCREMENT key to add to your existing *license.dat* file, or
- issue a new license key for the combined number of licenses as a replacement for your current license key.

The *license.dat* file can contain only one ClearDDTS FEATURE line. Additional FEATURE lines are ignored. The *license.dat* file can have multiple INCREMENT lines, since the license manager reads these as additions to existing FEATURE and INCREMENT lines.

Follow the directions in the e-mail from Rational to install the new key.

Changing your server host

If you want to move your license manager to a different machine you must obtain a new *license.dat* file from Rational by doing the following:

- 1** Access the Rational license key generation web page at:
`http://www.rational.com/products/clear_ddts/support`
- 2** Copy and paste the contents of your existing *license.dat* file in the text box supplied.
- 3** Enter the host ID for your new license server (use *lmhostid* to determine the host ID).
- 4** Fill out all other required information.

Rational will issue a new license key for the new license server and e-mail it to you within 24 hours.

Follow the directions in the e-mail from Rational to install the new key.

4

Using FastCGI

Note: This chapter applies to the *webddts* interface. If you plan to use the *xddts* or *tty* interfaces only, you can skip this section.

FastCGI is an open extension to CGI that improves performance of applications without introducing the overhead and complexity of proprietary Application Programming Interfaces (APIs).

As shipped, ClearDDTS uses a CGI program to bridge to FastCGI protocols. The FastCGI bridge improves ClearDDTS performance, and does not require you to make any changes to your Web server configuration or make any selections while installing or upgrading ClearDDTS.

You also have the option to upgrade your Web server to run native FastCGI or run without any form of FastCGI. These options and how to switch to them are detailed in this chapter.

Using Native FastCGI

Native FastCGI gives ClearDDTS the biggest performance improvement, but requires heavy modifications to your Web server and its configuration. Modules, plugins, and installation procedures for various servers can be obtained free from:

<http://www.fastcgi.com/servers/>

Warning: Before attempting to upgrade to native FastCGI, make a backup copy of your current Web server configuration. Upgrading to native FastCGI requires *expert* Web server administration knowledge. If you have concerns about upgrading your Web server, consider using the default FastCGI bridge instead.

To use native FastCGI with ClearDDTS:

- 1 Make a backup of your current web configuration.
- 2 Shut down your web server.
- 3 Upgrade your current web server to a version that supports FastCGI as described on the FastCGI web site and configure it for use with *webddts*.
- 4 If you have defined a CGI alias (sometimes called a ScriptAlias) for the location of the ClearDDTS scripts, remove that alias from your web server configuration.
- 5 Log in as the user *ddts*.

- 6 Run the *webconfigure* program.

This script loads your current configuration and asks you to select the mode in which you want to run the web interface.

- 7 Select Option 3, **Using native FastCGI protocols**.
- 8 Use the *ln -s* command to create a symbolic link between the ClearDDTS home directory and the *www* subdirectory. For example:

```
$ cd ~ddts
$ ln -s www ddts
```

This allows you to use the same URL, http://<host>/ddts/ddts_main, that you would use without FastCGI.

Note: Your web server must support following symbolic links.

- 9 Bring up your web server.
- 10 Log into the web interface to complete the configuration.

Running without FastCGI

You can run ClearDDTS without any type of FastCGI, however, running without FastCGI does not deliver optimum performance.

To switch to running without FastCGI:

- 1 Log in as the user *ddts*.
- 2 Run the *webconfigure* program.
The script loads your current configuration and asks you to select the mode in which you want to run the web interface.
- 3 Select Option 1, **Using standard CGI protocols**.
- 4 Log into the web interface to complete the configuration.

Index

A

- adminbug alic 1-10, 2-9
- aliases
 - scripts, images, attachments 1-5, 2-2
- Apache HTTP
 - Alias 1-6, 2-4
 - ScriptAlias 1-6, 2-4

B

- backup, performing 2-5

C

- CGI scripts
 - directory 1-5, 2-2
- changes, implementing 2-6
- ClearDDTS
 - ddtsclean 1-10
 - ddtsd daemon 1-10
 - SQL database 1-13
- ClearDDTS directory 1-4
- CPU requirements 1-3
- crontab 1-10
 - location of sample 1-11
 - removing 1-5
- customization
 - merging with new release 2-6, 2-10

D

- DAEMON line 3-3
- database
 - internal vs. external 1-13
- ddts administrative user 1-4
- ddts group
 - creating 1-4
- ddts user
 - creating 1-4
 - password 1-4

- removing a crontab 1-5
- ddts.lic 1-10, 2-9
- ddtsclean 1-10
- ddtsd 1-10
- ddtsinstall 1-8
- ddtsupgrade 2-6
- directory
 - creating for ClearDDTS 1-4
- disk space
 - locating 1-3
- documentation set vi

E

- evaluation license, see Startup key

F

- FastCGI 4-1
 - bridge 4-1
 - native 4-1
- FEATURE line 3-3, 3-11
- features, implementing 2-6
- FLEXlm 2-2, 3-6, 3-8
 - adding additional licenses 3-10
 - changing server host 3-11
 - DAEMON line 3-3
 - FEATURE line 3-3, 3-11
 - floating licenses 3-1
 - how licenses work 3-1
 - INCREMENT line 3-3, 3-11
 - license file 3-3
 - license manager 3-3
 - commands 3-4
 - license manager daemon(lmgrd) 3-3
 - linking to the license.dat file 3-9
 - LM_LICENSE_FILE variable 3-9
 - lmdown 3-5
 - lmgrd 3-4, 3-8
 - version 3-8
 - lmhostid 3-4

- lmreread 3-5
- lmstat 3-5
- log file 3-8
- managing licenses 3-1
- merging license.dat files 3-9
- node-locked licenses 3-1
- removing evaluation license 3-6
- restarting lmgrd automatically 3-10
- SERVER line 3-3
- setting up licensing 3-5
- starting evaluation license 1-11, 2-11
- starting new 3-7
- tracking floating licenses 3-2
- using existing lmgrd 3-8
- vendor daemon 3-3
- VENDOR line 3-3
- version 1-11, 2-11, 3-1
- what is licensed 3-2

G

- GlobeTrotter Software, website 3-6

H

- home directory
 - exporting 1-13
 - mounting from clients 1-14
- home directory, ClearDDTS 1-4
- HTTP Alias 1-5, 2-2
- HTTP ScriptAlias 1-5, 2-2
- HTTP server configuration 1-5, 2-2

I

- images
 - directory 1-5, 2-2
- INCREMENT line 3-3, 3-11
- installation
 - backing up 2-5
 - exporting home directory 1-13
 - locating disk space 1-3
 - man pages 1-15
 - mounting from clients 1-14
 - multiple architectures 1-7, 1-15, 2-5
 - path variable 1-15
 - running ddtsinstall script 1-8
 - upgrading previous release 2-1

- user access 1-15

installing

- modifying the HTTP configuration 1-5, 2-2

L

- license file 3-3
- license.dat file
 - linking to 3-9
 - merging 3-9
- licensing
 - adding additional 3-10
 - changing server host 3-11
 - DAEMON line 3-3
 - FEATURE line 3-3, 3-11
 - floating 3-1
 - how licenses work 3-1
 - INCREMENT line 3-3, 3-11
 - license file 3-3
 - license manager 3-3
 - commands 3-4
 - license manager daemon (lmgrd) 3-3
 - linking to the license.dat file 3-9
 - LM_LICENSE_FILE variable 3-9
 - lmdown 3-5
 - lmgrd 3-4, 3-8
 - version 3-8
 - lmhostid 3-4
 - lmreread 3-5
 - lmstat 3-5
 - log file 3-8
 - managing 3-1
 - merging license.dat files 3-9
 - node-locked 3-1
 - obtaining 3-6
 - removing evaluation license 3-6
 - restarting lmgrd automatically 3-10
 - SERVER line 3-3
 - setting up 3-5
 - starting evaluation license 1-11, 2-11
 - starting new 3-7
 - Startup key 1-2
 - entering 1-10, 2-9
 - tracking floating 3-2
 - using existing lmgrd 3-8
 - vendor daemon 3-3, 3-8
 - VENDOR line 3-3

- what is licensed 3-2
- LM_LICENSE_FILE variable 3-9
- lmdown 3-5
- lmgrd 3-3, 3-4, 3-8
 - restarting automatically 3-10
 - version 3-8
- lmhostid 3-4
- lmreread 3-5
- lmstat 3-5

M

- manual pages 1-15
- memory requirements 1-3
- multiple architectures
 - installing 1-7, 2-5

N

- Network File System (NFS)
 - using ClearDDTS on 1-3
- NEWCLEARDDTS directory 1-7, 2-5, 2-10

O

- obtaining licenses 3-6
- OLDCLEARDDTS directory 2-8, 2-10
- Oracle database
 - switching to 1-13
- Oracle database, using 1-13

P

- PATH environment variable 1-15
- previous release
 - upgrading 2-1
- PureLA licenses 2-2

R

- requirements, space 1-2

S

- server host
 - changing 3-11
- SERVER line 3-3
- site identifier 1-9
- software

- extracting 1-7, 2-5
- special user (ddts) 1-4
- SQL
 - database 1-13
- Startup license key 1-2
 - entering 1-10, 2-9
- swap space 1-3

T

- tar file 1-7
- technical support vii

U

- upgrade 2-1
 - script 2-6
- URL
 - for web interface 1-12, 2-12
- user login scripts 1-15

V

- vendor daemon 3-3
 - version 3-8
- VENDOR line 3-3

W

- web interface
 - running 1-12, 2-12
- webddts
 - URL 1-12, 2-12
 - using FastCGI 4-1
 - bridge 4-1
 - native 4-1

