Eloquence Installation and Configuration B.07.00

Edition E1004

Legal Notices

The information contained in this document is subject to change without notice.

MARXMEIER SOFTWARE AG MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Marxmeier Software AG shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013. Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19 (c) (1,2).

Acknowledgments

Marxmeier Software AG
Besenbruchstrasse 9
42285 Wuppertal
Germany

Eloquence is a trademark of Marxmeier Software AG in the US and other countries.

This software and documentation are based in part on HP software and documentation under license from Hewlett-Packard Company. HP is a trademark of Hewlett-Packard Company.

Printing History

The manual printing date indicates its current edition. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. New editions are complete revisions of the manual.The dates on the title page change only when a new edition or a new update is published.

Manual updates may be issued between editions to correct errors or document product changes. Manuals that are published on the Eloquence website (www.hp-eloquence.com/doc) may be updated more often, please visit this website periodically for the most recent versions. To ensure that you receive the updated or new editions, you should also subscribe to the appropriate product support service.

The software code printed alongside the date indicates the version level of the software product at the time the manual or update was issued. Many product updates and fixes do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one to one correspondence between product updates and manual updates.

First Edition	Apr 1990	A.01.00
Second Edition	July 1991	A.03.00
Third Edition	1995	A.05.00
Fourth Edition	July 1997	A.06.00
Fifth Edition	October 1997	A.06.00
Sixth Edition (E1202)	December 2002	B.06.32
Seventh Edition (E1004)	October 2004	B.07.00

Printed in the Federal Republic of Germany.

Printing History

Table of Contents

Contents

Contents

Contents

1

Installing Eloquence

This chapter provides an overview on the installation and configuration of the Eloquence software.

Introduction

This chapter provides an overview on installing and configuring the Eloquence B.07.00 software on your system.

- Eloquence Introduction
- Eloquence media
- Eloquence documentation
- The Eloquence web site
- The Eloquence license scheme

Since installation and system configuration is platform dependend, separate instructions for each supported platform are provided in subsequent chapters.

The information in this manual is directed to the system administrator for the Eloquence software. Some basic knowledge about operating the system is assumed.

What is Eloquence

Eloquence is a highly integrated applications development and runtime environment which significantly improves productivity. Developers can design and implement prototypes or customize existing programs easily. Since Eloquence was released in 1990, thousands of installations have confirmed this.

Recognized by software houses internationally, it is considered a powerful, reliable and flexible base for cost effective development and maintenance of commercial applications. A host of features, not found in such a combination in other programs, makes Eloquence unique and well suited to most customers needs.

Eloquence Media

Eloquence isavailable for download or on CD-ROM media.

CD-ROM media contents

The Eloquence CD-ROM media provides the following components:

- ❒ The Eloquence B.07.00 product for the HP-UX, Windows and Linux platform.
- ❒ The Eloquence on-line documentation in PDF, HTML and Postscript format.
- ❒ Contributed software. Please note, that this is not part of the Eloquence product.
  - The SQL/R product from Marxmeier Software AG. SQL/R provides SQL and ODBC access to the Eloquence database

Eloquence documentation

Eloquence no longer includes printed documentation by default (it is available as a separate option). Instead all documentation is available on-line in different formats.

PDF — You can view or print the documentation using the Acrobat Reader software which is included on the CD-ROM media or available free for almost all platforms on the Adobe web site (http://www.adobe.com/prodindex/acrobat/readstep.html). The Acrobat Reader does also include a plugin for the Netscape Navigator and Microsoft Internet Explorer web browsers so you can use a web browser to view the PDF documentation.

HTML — You can use your web browser to read the documentation.

Postscript — The documentation is included in Postscript format so it can be easily printed on a Postscript printer. To print on a non Poscript printer you can use the Acrobat Reader on Windows to print the PDF documenentation. Otherwise you may use the free ghostscript to convert the Postscript files into a printer specific format.

The PDF documentation provides the manuals in the same format as the printed documentation (its generated from the print files). The HTML documentation provides the same content but has been converted to on-line format. We recommend to install the Eloquence documentation on your web server, so it is available to all users instead of installing it locally on each computer.

The Eloquence web site

Eloquence is present on the Internet and has its own web site. If you are connected to the Internet, please point your browser at the URL

`http://www.marxmeier.com/eloquence/`

The Eloquence web site provides complete information on Eloquence ranging from marketing information, documentation, upcoming events to patches and support notes.

The Eloquence mailing List

The Eloquence mailing list was formed to support Eloquence programmers and users in their daily work. It is intended as an *enhancement* to the Eloquence support channel. Eloquence support is responsible for helping you in case of trouble and to solve your immediate problem. This mailing is intended to share knowledge and ideas with other Eloquence users.

So if you have a question like "*How could i ...*" or "*Why doesn't this work ...*" this list is the place to ask. Other welcome topics include the dicussion about Eloquence features and how you would like Eloquence to evolve in the future.

Please refer to the URL

`http://www.marxmeier.com/eloquence/mlist/`

for more information.

Eloquence license scheme

In order to use the Eloquence software, you need to provide a license key on each system where you intend to run Eloquence server software on. With Eloquence B.07.00 it is *not* required to install a license key on each system running Eloquence unless you intend to run Eloquence server software.

When you install Eloquence on a system for the first time, a temporary license key is generated which will expire after 30 days. This option is currently not available on Linux.

In order to request your permanent license key, plase fill out the "Permanent License Request" form included with the software and send it to the address below:

Fax +49 202 2431420

Mail Marxmeier Software AG
Attn. Eloquence Support
Besenbruchstrasse 9
D-42285 Wuppertal
Germany

You may also request a license key on-line fro the Eloquence web site. Please refer to the URL

`http://www.marxmeier.com/eloquence/license/`

After you received your permanent license key, you must add it to your license file in order to activate it. The location of the Eloquence license file depends on operating system revision:

- **HP-UX and Linux**
  `/etc/opt/eloquence6/license`
- **Windows**
  The file **`license`** is located in the **`etc`** subdirectory of the Eloquence installation path.

The license file is a plain text file which contains all licences which apply to the Eloquence product. The utility **`/opt/eloquence6/etc/chklic`** may be used to check the licence file.

Example license file contents

```
ELOQ B.07.00 base 0-0-0 0-0-0 0 1 19eb2b9d3e23517f261e2f56cd4ea5e2
```

```
ELOQ B.07.00 hp3k 0-0-0 0-0-0 0 2 0ce6387e66337bace426a097fd472c31
```

Please refer to /opt/eloquence6/newconfig/config/license for a detailed description of the license file format.

Eloquence Personal Edition License

The Eloquence "Personal Edition" provides a free, perpetual license for personal or evaluation usage. It is limited to two users and a maximum database size of 50MB.

This is different from a temporary license key, as the Eloquence "Personal Edition" does not expire and can be used on any number of systems.

The Eloquence Personal Edition Software is subject to the Eloquence license agreement and comes "AS IS" with NO WARRANTY whatsoever. Be sure to read and agree to the license BEFORE you use the software.

Please note, that there is no formal support for the Eloquence "Personal Edition". Support is only available through the Internet.

The license key for the Eloquence Personal Edition license is installed by default. It is also included in the license template file (/opt/eloquence6/newconfig/config/license on HP-UX and Linux or license.sam on Windows).

2

Installing Eloquence on the HP-UX platform

This chapter covers the installation of Eloquence on the HP-UX platform

- Software and patch installation
- Configuring the HP-UX operating system
- Configuring Eloquence

Installation Overview

This chapter describes the installation of the Eloquence product on the HP-UX platform, the configuration of the HP-UX operating system and the configuration of the Eloquence software on HP-UX.

Please read the Eloquence release notes *before* installing or upgrating Eloquence as the release notes may include additional or more recent information.

Eloquence B.07.00 on the HP-UX platform requires the following HP-UX versions:

- HP-UX 10.20 on PA-RISC based systems (HP9000 server)
- HP-UX 11.23 in Itanium based systems (HP Integrity server)

Eloquence B.07.00 provides different versions for HP-UX based systems:

- PA-RISC 1.1 (PA11) based systems. This build is compatible with all PA-RISC based systems and should be used for older HP9000 systems.
- PA-RISC 2.0 (PA20) based systems. This build should be used with all recent PA-RISC based HP9000 systems.
- Itanium2 (IA64) based systems

NOTE:

You may use SAM to identify the CPU in your HP-UX system. Please select the Performance Monitors -> System Properties function. If the "CPU version" specifies PA 8000 or greater you have a PA-RISC 2.0 based system. A PA 7XXX CPU designates a PA-RISC 1.1 based system.

Updating from previous Eloquence versions

If you are updating from a previous Eloquence revision, there are some special considerations which should be taken into.

- The Eloquence configuration files may have changed. You may want to consider comparing the Eloquence configuration files with the template configuration files.
- When using the Eloquence Personal Edition license you may need to copy the template license file manually.
- Please note, that the HP-UX kernel configuration requirements have changed from previous Eloquence revisions.

Installation on HP-UX

Eloquence B.07.00 on the HP-UX platform requires HP-UX release 10.20 (HP9000, PA-RISC) or HP-UX 11.23 (HP Integrity, Itanium) or later and is installed in the **`/opt/eloquence6`** directory.

Eloquence is installed and updated using the swinstall operating system utility.

The Eloquence installation archives may be obtained from a CD-ROM media or downloaded from the Internet. The installation procedure with both versions is the same once swinstall is started.

Installing from CD-ROM media

As “superuser” follow the steps below to install the Eloquence software.

- Mount the CD-ROM media. For HP-UX 11.11 and beyond a command line like below may be used

  ```
  mount -F cdfs -o rr /dev/dsk/c0t0d0 /cdrom
  ```

  For previous HP-UX versions that don’t support the "Rock Ridge" extensions, a command line like below may be used

  ```
  mount -F cdfs -o cdcase /dev/dsk/c0t0d0 /cdrom
  ```

  where */dev/dsk/c0t0d0* is the device file associated with the CD-ROM drive and */cdrom* is the directory where the CD-ROM should be mounted. Please refer to the mount_cdfs(1M) man page for more information.

  The Eloquence software for the HP-UX platform is in the **`B0700/hpux`** subdirectory.

- Choose the appropriate installation archive for your HP-UX system
  - **Eloquence-B0700-1.depot** - PA-RISC 1.1 based systems
  - **Eloquence-B0700-1-pa20.depot** - PA-RISC 2.0 based systems
  - **Eloquence-B0700-1-ia64.depot** - Itanium based systems
- Start swinstall by typing:

  ```
  /usr/sbin/swinstall -s /cdrom/B0700/hpux/Eloquence-B0700-1.depot
  ```

  where */cdrom* is the directory where the cdrom is mounted.

NOTE: Because HP-UX versions before 11.11 do not support the Rock-Ridge CD-ROM extensions all file names are lower case (if mounted with the cdcase option) and files may have an extra dot (.) at the end. However file names in this document are listed in mixed case.

Download Installation

Eloquence can be downloaded from the Eloquence web site at:
http://www.marxmeier.com/eloquence/download

Download the appropriate version of Eloquence for your HP-UX system

- **Eloquence-B0700-1.depot.gz** - PA-RISC 1.1 based systems
- **Eloquence-B0700-1-pa20.depot.gz** - PA-RISC 2.0 based systems
- **Eloquence-B0700-1-ia64.depot.gz** - Itanium based systems

Eloquence installation files downloaded from the Eloquence ftp server are compressed with gzip and must be uncompressed before installation.

As “superuser” follow the steps below to install the Eloquence software.

- Unpack the archive at a temporary location.

  ```
  gzip -d Eloquence-B0700-1-pa20.depot.gz
  ```

  This unpacks Eloquence-B0700-1-pa20.depot.gz to Eloquence-B0700-1-pa20.depot. The compressed archive file is deleted.

- Start “swinstall” by typing:

  ```
  /usr/sbin/swinstall -s /tmp/Eloquence-B0700-1-pa20.depot
  ```

  where */tmp/Eloquence-B0700-1-pa20.depot* is the the absolute path of the Eloquence-B0700-1-pa20.depot file.

Using swinstall

By default, swinstall should display the "top" level options ("Bundles and Products"). The following options are available:

Eloquence_Std	Installs the base product
Eloquence_Tmplic	Installs the base product and a temporary license key which is valid for a period of four weeks. Please note that a temporary license key can only be installed once per system
Eloquence_All	Installs all components (including temporary license key)

Alterntively, the "View - Change Software View - Start with Products" menu option selects the "Products" view in swinstall which allows to select single Eloquence product options.

The Eloquence product supports the following options:

Base	The Eloquence base product
Tmplic	Install a temporary license key

To install the Eloquence software, follow the directions below:

- In the Software Selection Window, select/highlight the product options you would like to install.
- Then choose the “Mark for Install” item from the Actions Menu. The “Marked?” column will be set to “Yes”.
- Select the “Install (analysis...)” item from the Actions Menu. When the analysis is finished with no error, i.e. Status:Ready, click OK.
- Click Yes in the Confirmation window to begin the actual installation.
- When the installation is completed, a dialog is displayed to notify you that the install task is completed. You may exit then.

HP-UX Patches

The following HP-UX patches are required to use Eloquence. Please install these patches (or successors) on your HP-UX system.

PHCO_30693	For HP-UX 11iv2 (HP-UX 11.23) installation of patch PHCO_30693 is suggested (xcurses patch). Please refer to the patch documentation PHCO_30693.text for more information.
PHKL_23995	For HP-UX 11i (HP-UX 11.11) installation of patch PHKL_23995 is suggested (fixes erroneous EFAULT return). Please refer to the patch documentation PHKL_23995.text for more information.
PHKL_24005	If you are on HP-UX 11.0 we recommend the HP-UX patch PHKL_24005 (fixes erroneous EFAULT return). Please refer to the patch documentation PHKL_24005.text for more information.
PHCO_10947	For HP-UX 10.20 installation of patch PHCO_10947 is required (curses patch). Please refer to the patch documentation PHCO_10947.text for more information.

You may need additional HP-UX patches but these patches are known to fix problems encountered with Eloquence.

HP-UX patches may be obtained from the Internet at the **HP IT Resource Center** web site at **http://itrc.hp.com**

For your convenience the HP-UX patch files are also included on the CD-ROM media (in the **/B0700/hpux** directory) and available for download from the Eloquence web site.

Current information on HP-UX patches (required for Eloquence) is available on the Eloquence web site at:
http://www.marxmeier.com/eloquence/support/B07/patch.html

Installing Eloquence Patches

Eloquence patches are provided to either fix a defect or limitation that was found after release and/or to add new functionality.

For proper operation of the Eloquence software, it is important that the recent patches are installed. For this purpose, a patch bundle is provided which can be used to conveniently install all recommended patches at once.

The **/B0700/patch/bundle** directory on the CD-ROM media contains the current Eloquence patch bundle at the time of compilation. The individual Eloquence patch files are contained in the **/B0700/patch** directory.

To find out about the most recent patches, please visit the Eloquence web site at: **http://www.marxmeier.com/eloquence/support/B07/patch.html**

NOTE: Installation of the "recommended" set of Eloquence patches is strongly recommended and may be required for correct product function.

Installation instructions

To install a patch or a patch bundle you need to uncompress and unpack it with gzip and tar. The gzip utility is included with HP-UX 10.x and above. Installation requires root privileges.

```
cd /opt/eloquence6
gzip -dc /path/to/PE70-<id>-hpux-<arch>.tar.gz | tar xf -
```

where <id> is the patch id (such as 0304041) and <arch> is the patch architecture (such as pa20). The architecture should be equivalent to the Eloquence distribution that was installed initially.

The following architectures are supported with Eloquence B.07.00:

- **pa11** - PA-RISC 1.1
- **pa20** - PA-RISC 2.0 (32 and 64 bit HP-UX)
- **ia64** - Itanium

HP-UX patches include a README file that is placed in the **share/doc** subdirectory for later reference.

Configuring the HP-UX system

After installing the Eloquence software, configuration of your HP-UX system is required:

- Configure Kernel limits
- Configure or verify service names and port numbers
- Install the license key
- Configure automatic startup/shutdow
- Create new users and groups for use with Eloquence

Configure Kernel limits

Eloquence requires kernel resources such as files, locks, semaphores and shared memory segments. The HP-UX kernel can be configured to the requirements of the applications. This is done by tuning kernel parameters and building a new kernel with SAM.

Eloquence resource requirements depend on configuration and usage:

- eloqsd needs a shared memory segment and a semaphore set with two semaphores.
- eloqdb6 requires a semaphore set with two semaphores per active session if IPC transport is enabled in the configuration (EnableIPC=2) and optional additional shared memory segment per active connection (EnableIPC=1).
- Each eloqcore process uses up to 60 open files and 20 file locks. When using Eloquence DLLs, a separate shared memory segment is used for each active DLL.
- When eloq is used, a pty is required per task. When you login over the network (for example using Telnet), an additional pty is required.

In order to estimate the required kernel resources, you should estimate the number of active users, the number of running eloqcore processes and the number of active data base sessions (probably the same as the number of eloqcore processes) and use the data provided above to calculate the total requirements for Eloquence.

- Number of concurrent processes
- Number of concurrent processes per user
- Number of ptys
- Number of files
- Number of locks
- Number of shared memory segments
- Number of semaphore sets
- Number of semaphores

The next step is to use those values to tune the kernel parameters. Please keep in mind that other processes use kernel resources as well, so be generous. Please refer to the HP-UX administration documentation for a complete reference and more information on kernel parameters.

Citing the SAM online documentation on kernel parameters:

> Certain kernel operating parameters can be configured to fit specific system needs, resulting in better performance or more effective allocation of resources. The ideal value for each parameter is often determined by the system's particular hardware configuration, the specific mix of applications the system runs, and the trustworthiness of system users; factors that vary widely from system to system.
>
> HP attempts to provide reasonable default parameter settings, but you may find it necessary or beneficial to modify these settings to better suit the needs of your particular system's users. Use the list below to obtain detailed information about each configurable kernel parameter.
>
> *WARNING*: Changing kernel parameters to improper or inappropriate values or combinations of values can cause data loss, system panics, or other operating anomalies, depending on which parameters are set to what values. Before altering the value of any configurable kernel parameter, be sure you know the implications of making the change. Never set any system parameter to a value outside the allowable range for that parameter (SAM refuses to store values outside of the allowable range). Many parameters interact, and their values must be selected in a balanced way.

Note that individual parameters usually pertain to a specific subsystem; some are independent, but others are interrelated or interact with each other. The following subsections are grouped according to subsystem.

Configuring the number of processes

The *nproc* kernel parameter specifies the maximum total number of processes that can exist simultaneously in the system. When the total number of processes in the system is larger than nproc, the system issues these messages:

At the system console:

```
proc: table is full
```

Also, if a user tries to start a new process from a shell, the following message prints on their terminal:

```
no more processes
```

Configuring the number of user processes

The *maxuprc* kernel parameter specifies the maximum number of simultaneous processes available to each user on the system. The HP-UX default is 50 which should usually be sufficient. A user is identified by the user ID number, not by the login instances.

If a user attempts to start a new process that would cause the total number of processes for that user to exceed maxuprc, the system issues an error message to the user:

```
no more processes
```

The maxuprc should be much smaller than the nproc parameter, otherwise a single user could use up all system resources (for example due to a program failure).

Configuring the number of ptys (pseudo ttys)

The *npty* kernel parameter specifies the maximum number of pseudo-ttys available on the system. The default value for HP-UX is 60.

After rasing this value you also need to create the new pty slave and master side device files in the /dev/pty and /dev/ptym directories. This can be done with insf(1m). Please note, that for npty values above 60 you have to specify the -n argument to insf.

```
/sbin/insf -n 60 -d pty
```

Configuring the number of files

The *nfile* parameter specifies the maximum number of files that can be open simultaneously on the system at any given time. Be generous with this number because the required memory is minimal, and not having enough restricts system processing capacity.

The *ninode* parameter specifies the maximum number of open inodes that can be in memory. Each unique open file has an open inode associated with it. Therefore, the larger the number of unique open files, the larger ninode should be.

Also the *maxfile_lim* specifies the max. number of files that could be opened by a single process.

Configuring the number of locks

The *nflocks* parameter specifies the number of file locks that are available system-wide. The HP-UX default value is 200.

Each DATA file opened within eloqcore needs a lock for each process. So this value should be adapted to the anticipated usage. Each eloqcore process can currently open up to 20 DATA files concurrently.

Configuring the number of shared memory segments

The *shmmni* parameter specifies the maximum number of shared memory segments allowed to exist simultaneously, system-wide. The HP-UX default is 200 which should be sufficient unless you have a big number of active users.

Setting *shmmni* to an arbitrarily large number wastes memory and can degrade system performance. Setting the value too high on systems with small memory configuration may consume enough memory space that the system cannot boot. Select a value that is close to actual system requirements for optimum memory usage. A value not exceeding 1024 is recommended unless system requirements dictate otherwise.

The *shmseg* parameter specifies maximum number of shared memory segments that can be simultaneously attached to a single process. The HP-UX default is 12.

If using IPC transport for the eloqdb6 database server with separate shared memory segments (EnableIPC=1) the *shmseg* value should be adapted to the max. number of concurrent local processes accessing a single server.

Configuring the number of semaphore sets

The *semmni* parameter defines the maximum number of semaphore sets that can exist simultaneously on the system. The HP-UX default is 64.

If using IPC transport for the eloqdb5 and eloqdb6 servers, it may be necessry to adapt this value according to the anticipated max. number of concurrent local eloqcore processes which access the eloqdb5 or eloqdb6 server.

The *semmap* parameter specifies the size of the free-space resource map used for allocating new System V IPC semaphores. The HP-UX default is semmni+2.

If semaphore usage is heavy and a request for a semaphore set cannot be accommodated, the following message appears:

```
danger: mfree map overflow
```

You should then configure a new kernel with a larger value for semmap.

The *semmns* parameter specifies the system-wide maximum number of individual semaphores that can be allocated. The HP-UX default is 128 which should usually be sufficient.

The *semmnu* parameter defines the maximum number of processes that can have undo operations pending on semaphores. The HP-UX default is 30.

If using IPC transport for the eloqdb5 and eloqdb6 servers, it may be necessary to adapt this value according to the anticipated max. number of concurrent local eloqcore processes which access the eloqdb5 or eloqdb6 server. A semaphore undo entry is required for each process accessing the eloqdb5 or eloqdb6 servers with using the IPC transport.

The *semume* parameter defines the maximum number of semaphores that a given process can have undo operations pending on. The HP-UX default is 10.

An undo is a special, optional, flag in a semaphore operation which causes that operation to be undone if the process which invoked it terminates. semume specifies the maximum number of semaphores that any given process can have undos pending on.

If using IPC transport for the eloqdb5 and eloqdb6 servers, it may be necessary to adapt this value according to the anticipated max. number of concurrent local eloqcore processes which access the eloqdb5 or eloqdb6 server. A semaphore undo entry is required for each process accessing the eloqdb5 or eloqdb6 servers with using the IPC transport.

Configuration examples

The following configuration examples explain the kernel resourceswith some example configurations:

- Using two instances of the Eloquence database server with a max. of 200 (instance 1) and 20 (instance 2) concurrent connections (EnableIPC=2):
  - semmni should be set to at least 224, semmap to semmni+2
    The eloqdb6 server needs a sepahore set for each connection and another two semaphore sets for internal purposes for each eloqdb6 instance.
  - semmns should be set to at least 230
    The eloqdb6 server needs a sepaphore for each connection and in addition a few semaphores internally.
  - semume should be set to at least 200
    The eloqdb6 server uses a semaphore undo operation for each local connection. This should cover the largest eloqdb6 instance on the system.
  - maxfiles_lim needs to set to at least 250
    The eloqdb6 server opens a file (a socket) for each connection and in addition one for each volume file and log file.
- The same configuration but using separate shared memory segments (EnableIPC=1). The listed kernel resources are required in addition:
  - shmmni be set to at least 220, shmmap shmmni+2
    Each eloqdb6 instance needs a separete shared memory segment for each connec-

tion.

  - shmseg should be set to at least 200
    The highest number of shared memory segments that can be attached to a single process. As eloqdb6 needs a separate shared memory segment per connection this should cover the largest eloqdb6 instance on the system.

- Running 50 eloq processes and 100 eloqcore processes requires the following kernel resources:
  - npty should be set to at least 100
    eloq uses a separate pty for each eloqcore process.
  - nfile should be set to at least 6000
    Each eloqcore process may open up to 60 files.
  - nflocks should be set to at least 2000
    Each eloqcore may obtain up to 20 file locks.

Configure or verify service names and port numbers

You may want to verify the service names and port numbers added to your **`/etc/services`** file during installation are appropriate.

In your **`/etc/services`** file should be entries like below. Please make sure the specified port numbers are available on your system:

```
eloqsd    8100/tcp    # Eloquence A.06.00 eloqsd server
eloqdb    8102/tcp    # Eloquence A.06.00 eloqdb6 server
eloqdb5   8104/tcp    # Eloquence A.06.00 eloqdb5 server
runsrv    8010/tcp    # Eloquence RunSRV (Windows)
```

where the first column specifies the service name (eg. eloqsd) and the second column the associated port number and protocol (eg. 8100/tcp). The selected port numbers may not already be in use by another programs.

You may want to define the Eloquence specific service names in your **`/etc/services`** file (if you are using NIS it is probably required to do this on the NIS master server). This is optional, as you can specify the port number directly instead of a service name.

Install the license key

Add the supplied license keys to the **`/etc/opt/eloquence6/license`** file. We recommend that you comment out any existing license keys.

If a temporary license was created during installation this step may be skipped and the permanent license added later.

If the Personal Edition license should be used this step is not necessary as the PE license key is included in the default license template file.

The **`/opt/eloquence6/etc/chklic`** utility may be used to verify the license file.

Configure automatic startup/shutdown

The Eloquence server processes may be configured to start automatically by default if runlevel 2 is entered (after a reboot or single user mode) and shut down automatically on reboot.

The automatic server startup is configured in the Eloquence startup configuration file **`/etc/rc.config.d/eloquence6`**. Please refer to the next section for details

Create new users and groups for use with Eloquence

It is strongly recommended to run the eloqdb6 server and the eloqsd server with a dedicated user and group.

- **eloqsd** - a dedicated user and group should be created (default eloqsd/eloqsd) and configured in the eloqsd.cfg file. Please refer to the ssection on configuring the eloqsd process for details.
- **eloqdb6** - a dedicated user and group should be created (default eloqdb/eloqdb) and configured in the eloqdb6 configuration file. Please refer to the section on configuring the eloqdb6 process for details.

Eloquence startup/shutdown script

Manually starting and stopping the server processes

The Eloquence startup script may be used to manually start/stop/restart server processes or verify a server process is active.

Please note that using the startup/shutdown script to control server processes need root capabilities. Status inquiries do not require root capabilities.

The Eloquence startup/shutdown script supports the following options:

start [arg list ...]
Start the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or database server instances) are started. If the argument list is present, only the specified services (or instances) are started.

Please note: A database server instance is not be started automatically if the ELOQDB6_START[0] configuration option is set to 0.

stop [arg list ...]
Stop the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are stopped. If the argument list is present, only the specified services (or instances) are stopped.

Please note that the Eloquence start/stop script only manages configured services. Previous versions of the shutdown script stopped unconfigured Eloquence services as well.

restart [arg list ...]
Restart the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are restarted.

status [arg list ...]
Display the status of Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are returned.

info [arg list ...]
Display the startup configuration of Eloquence services. If the optional argument list is not provided, the configuration of all Eloquence services (or instances) is returned.

An optional argument list may be used to specify which Eloquence service(s) should be managed. An argument specifies either an Eloquence service or a database instance which is configued in the startup configuration file.

eloqsd	Specfies the Eloquence eloqsd service
eloqdb6	Specfies all configured Eloquence eloqdb6 instances

For example:

```
Start all configured services
# /sbin/init.d/eloq6 start

Restart eloqsd service
# /sbin/init.d/eloq6 restart eloqsd

Stop all configured eloqdb6 instances
# /sbin/init.d/eloq6 stop eloqdb6

Start eloqdb6 instances prod and test
# /sbin/init.d/eloq6 start prod test

# /sbin/init.d/eloq6 start
Starting eloqsd daemon                              [OK]
Starting eloqdb6[eloqdb] daemon                     [OK]

# /sbin/init.d/eloq6 status
eloqsd process is active (pid 16645)                [running]
eloqdb6[eloqdb] process is active (pid 16652)       [running]

# /sbin/init.d/eloq6 info eloqdb
 eloqdb6 instance id = eloqdb
  configuration file = /etc/opt/eloquence6/eloqdb6.cfg
  service = eloqdb
  args =
  run prefix =
  automatic start = 1
```

Location of the Eloquence startup/shutdown script

The location of the Eloquence startup/shutdown script depends on the operating system:

HP-UX	/sbin/init.d/eloq6
SuSE Linux	/etc/init.d/eloq6 (SuSE Linux 7.1 and above) /sbin/init.d/eloq6 (SuSE Linux before 7.1)
Red Hat Linux	/etc/init.d/eloq6 or /etc/rc.d/init.d/eloq6
LSB compliant Linux	/etc/init.d/eloq6

Location of the Eloquence startup/shutdown configuration file

The Eloquence startup/shutdown script uses a new configuration template file. It is backwards compatible to previous Eloquence versions. The location of the startup configuration file depends on the operating system:

HP-UX	/etc/rc.config.d/eloquence6
SuSE Linux	/etc/sysconfig/eloquence6 (SuSE 8.0 Linux and above) /etc/rc.config.d/eloquence6 (SuSE Linux 7.3 and before)
Red Hat Linux	/etc/sysconfig/eloquence6
LSB compliant Linux	/etc/sysconfig/eloquence6

During the update from a previous Eloquence installation a previous configuration file is moved to the new location. However, we recommend to start with the configuration template file if updating from a previous HP Eloquence version.

The example command shown below is specific to HP-UX, please change the target location depending on the operating system.

```
cp /opt/eloquence6/newconfig/startup/eloquence.rc \
   /etc/rc.config.d/eloquence6
```

Eloquence startup/shutdown configuration options

The Eloquence startup/shutdown configuration file specifies the operation of the startup/shutdown script. The following global configuration options are available:

START_ELOQ	If START_ELOQ is set to anything besides 1 it disables the automatic Eloquence startup entirely. To retain compatibility to previous Eloquence releases on Linux "yes" is accepted as well. The default is START_ELOQ=1
START_ELOQSD	If START_ELOQSD is set to 1 the eloqsd daemon is started automatically. The default is START_ELOQSD=1
ELOQSD_ARGS	The ELOQSD_ARGS allows specifying eloqsd command line arguments. The default is ELOQSD_ARGS=""
ELOQSD_RUNPFX	The ELOQSD_RUNPFX variable allows specifying a command which is then expected to start eloqsd. The default is ELOQSD_RUNPFX="" For example: ELOQSD_RUNPFX="/usr/bin/nice -n10" starts eloqsd with a nice value of 10.
START_ELOQDB6	If START_ELOQDB6 is set to 1 then the eloqdb6 daemon is started automatically. The default is

START_ELOQDB6=1

ELOQDB6_DEFAULT_ARGS The ELOQDB6_DEFAULT_ARGS specifies the default command line arguments which are used with all eloqdb6 instances unless defined specifically. The default is ELOQDB6_DEFAULT_ARGS=""

The Eloquence startup/shutdown configuration file may be used to configure multiple eloqdb6 instances.

Unless specified here, the Eloquence startup/shutdown script only supports the default database server instance. This section allows specifying database server instances which are to be maintained by the eloquence startup script. Each database server instance must be specified with a distinct index, starting with the index 0.

The following options may be specified for each eloqdb6 instance:

ELOQDB6_CFG[0] This option specifies the configuratin file which is used with this database server instance (required). This file can be specified with an absolute path or relative to the directory /etc/opt/eloquence6
The default is ELOQDB6_CFG[0]=eloqdb6.cfg

ELOQDB6_ID[0] This configuration option may be used to specify a symbolic name to the database server instance (optional). If not specified, this defaults to the service name or port number, specified in the config file or "eloqdb". The instance id may be used as an optional argument with the Eloquence startup/shutdown script to specify a database server instance.

ELOQDB6_START[0] This configuration option may be used to specify if the Eloquence startup script should start this service automatically. If set to 1, the database server instance is started automatically. If set to 0, the startup script will ignore this entry for the start and restart option and the database server instance must be managed manually by providing the instance id name on the script command line.
The default is ELOQDB6_START[0]=1

ELOQDB6_ARGS[0] The ELOQDB6_ARGS[0] option may be used to specify eloqdb6 command line arguments for this database instance. If not specified this defaults to ELOQDB6_DEFAULT_ARGS.

ELOQDB6_RUNPFX[0] This configuration option may be used to specify a command which is then expected to start eloqdb6. The default is ELOQDB6_RUNPFX[0]=""

For example:
ELOQDB6_RUNPFX[0]="/usr/bin/setmemwindow"
starts the eloqdb6 process in a separate memory window.

For example:

```
START_ELOQSD=0
ELOQDB6_CFG[0]=eloqdb6_prod.cfg
ELOQDB6_START[0]=1
ELOQDB6_ID[0]=prod
ELOQDB6_CFG[1]=eloqdb6_test.cfg
ELOQDB6_START[0]=1
ELOQDB6_ID[0]=test
```

This example configuration file specifies to not start the eloqsd service and defines two eloqdb6 instances, prod and test.

Eloquence startup/shutdown configuration file template

```
# @(#) eloquence.rc - B.07.00
# Eloquence automatic startup configuration
#
# Eloquence startup confguration file.
# This file is sourced by the startup/shutdown script.
#
# The location of this file depends on the operating system:
# HP-UX: /etc/rc.config.d/eloquence
# RedHat Linux, SuSE Linux 8.0+: /etc/sysconfig/eloquence
# SuSE Linux 7.0 to 7.3 /etc/rc.config.d/eloquence

### global settings

# If START_ELOQ is set to anything besides 1 it disables the
# automatic Eloquence startup entirely. To retain compatibility
# to previous Eloquence releases on Linux "yes" is accepted
# as well.
#START_ELOQ=1

# If START_ELOQSD is set to 1 the eloqsd daemon is started
# automatically. The ELOQSD_ARGS allows specifying eloqsd command
# line arguments. The ELOQSD_RUNPFX variable allows specifying
# a command which is then expected to start eloqsd (e.g. nice).
#START_ELOQSD=1
#ELOQSD_ARGS=""
#ELOQSD_RUNPFX=""

# If START_ELOQDB6 is set to 1 then the eloqdb6 daemon is started
# automatically. The ELOQDB6_DEFAULT_ARGS specifies the default
# command line arguments, which are used with all eloqdb6
# instances unless defined specifically.
#START_ELOQDB6=1
#ELOQDB6_DEFAULT_ARGS=""

### eloqdb6 instances ###

# Unless specified here, the Eloquence startup/shutdown script
# only supports the default database server instance. This
# section allows to specify database server instances which are
```

```
# to be maintained by the eloquence startup script.
# Each database server instance must be specified with a distinct
# index, starting with the index 0.
#
# ELOQDB6_CFG[0] =
#  Configuration file used with this database instance (required).
#  This can be an absolute path or relative to /etc/opt/eloquence6
#
# ELOQDB6_ID[0] =
#  The instance id is used to specify an alias to a eloqdb6 instance
#  (optional). If not specified, this defaults to the service name
#  or port number, specified in the config file (defaults to eloqdb).
#  The instance id may be used as a startup script argument to
#  specify a database server instance.
#
# ELOQDB6_ARGS[0]=""
#  Command line arguments for this instance (optional). If not
#  specified (eg. commented out), the ELOQDB6_DEFAULT_ARGS is used.
#
# ELOQDB6_START[0]={0|1}
#  Specifies if the Eloquence startup/shutdown script should make
#  use of this entry (optional). This setting defaults to 1 which
#  specifies a database instance is used with the startup script.
#  If set to 0, the startup script will ignore this entry for the
#  start and restart option.
#
# ELOQDB6_RUNPFX[0]=""
#  Allows to define a command which is then expected to start
#  the eloqdb6 instance (e.g. nice). On HP-UX this may be used
#  to define a memory window used by the database server instance.
#
#ELOQDB6_CFG[0]=eloqdb6.cfg
#ELOQDB6_START[0]=1
#ELOQDB6_ARGS[0]=""
#ELOQDB6_ID[0]=""
#ELOQDB6_RUNPFX[0]=""
```

Configuring the User Environment

There are two files where the user environment can be configured:

- The environment defined in **`/etc/profile`** affects all users.
- The file **`.profile`** located in a user's home directory configures the user-specific environment.

Perform the following steps:

1 Make the Eloquence executables accessible:

- This step is already done during software installation. The installation script automatically adds **`/opt/eloquence6/bin`** to the global path configuration file **`/etc/PATH`**.

2 Make sure that the terminal type is configured properly for all users. Normally, **`/etc/profile`** provides a mechanism that automatically detects the terminal type and sets the **`TERM=`** environment variable appropriately.

3 Make your configuration changes active. This is done by logging off and back on again.

Configuring the eloqsd server

Eloquence A.06.00 no longer has an eloqd server. Beginning with Eloquence A.06.00, the former eloqd has been renamed to eloqsd. This has been done to retain the interoperability of Eloquence A.06.00 with previous releases on the same system.

The eloqsd server is an important part of Eloquence. It is responsible for the following tasks:

- Eloqsd coordinates the TASKID values.
- Eloqsd provides file sharing capabilities for the new graphical Eloquence development environment.
- Eloqsd is used to start eloqcore processes in the background.
- Eloqsd is used to count active users and does validate it against available user licenses.
- Eloqsd optionally provides a HTTP interface so server status information can be queried with a web browser.

NOTE: The eloqsd server is not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you don't need to configure and run the eloqsd server.

Eloquence implements some limited file sharing capabilities for the new graphical development environment through the eloqsd server. This makes it independent of the availability of specific network file systems (NFS/ SMB) and overcomes inappropriate limitations.

Running an eloqsd server on your system is not mandantory unless you are using eloq (providing virtual terminal capabilities). However, when no eloqsd process is active, TASKID values are no longer unique and are set to 1 by default.

In order to run the eloqsd server it is required to adapt your system configuration. This involves the following steps:

- Configuring the eloqsd TCP service
- Configuring the default eloqsd account and group
- Configuring the eloqsd server startup
- Configuring the eloqsd server

Configuring the eloqsd TCP service

You may want to define the eloqsd service names in your /etc/services file. This is optional, as you can specify the port number directly in the eloqsd.cfg configuration file.

Please add lines like below to your **/etc/services** file:

```
eloqsd     8100/tcp    # Eloquence A.06.00 eloqsd server
```

the first column specifies the service name (eg. eloqsd) and the second column the associated port number and protocol (eg. 8100/tcp). The selected port numbers may not already be in use by another programs.

NOTE: All systems must use the same port numbers in order to communicate.

Configuring the default eloqsd account and group

The **eloqsd** server requires you to specify an account and group name in the configuration file. Whenever **eloqsd** is started with root capabilities it will switch to the specified account/group instead. This is required, because for one it is generally not a good idea to run programs with root capabilities unless necessary, on the other hand this is used as the default account and group for users accessing files through the eloqsd or starting a background process.

While you can specify any user or group account in the configuration file, we recommend to create a specific user account and group for Eloquence which is used by the eloqsd server.

We recommend to create the user account *eloqsd* and the group *eloqsd* which should have the account *eloqsd* as a member. You can configure this with SAM. The account should be marked "disabled" in SAM (which will result in an asterisk in the password) to prevent logins using the eloqsd account.

Configuring the eloqsd server startup

To specify thet the eloqsd server is started at system boot time, set the **START_ELOQSD** variable to **1** in the Eloquence startup/shutdown configuration file **/etc/rc.config.d/eloquence6**. Change the line defining the **START_ELOQSD** variable like this:

```
START_ELOQSD=1
```

Configuring the eloqsd server

The eloqsd server is configured by editing the **eloqsd.cfg**, **eloqsd.user** and **eloqsd.share** configuration files. All configuration files provide complete inline documentation and are included at the end of this section for your reference.

The configuration files are located in the **/opt/eloquence6/etc** directory. Each configuration file is responsible for a specific part of the eloqsd configuration:

eloqsd.cfg This is used for the general configuration of the server.

eloqsd.user Eloqsd provides its own user configuration. This makes it possible to define eloqsd users without the need to have a system account for each individual user. Instead eloqsd users are associated with system accounts and groups.

As passwords are defined in this file we consider it good practice to make this file unreadable for regular users. You should chown it to root and chmod id to 400.

eloqsd.share This configuration file is used to define resources which can be accessed through the eloqsd server.

The eloqsd command line options

The eloqsd server supports the following command line options which can be used to temporarily override configured settings in the **`eloqsd.cfg`** configuration file.

```
usage: eloqsd [options]
options:
 -help       = show usage (this list)
 -c name     = configuration file
 -d flags    = debug mode
 -l name     = log file name (or console/syslog/default)
 -f          = run in foreground
 -s name     = service name (tcp/ip transport)
 -F facility = syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
 -I ident    = syslog identifier
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-f	Run in foreground. This is used for debugging the eloqsd server.	
-s name	The service name (as defined in /etc/services) or the port number where the server should listen for requests. The default value is eloqdb6.	Service

Option	Description	Equiv.*
-F facility	When logging to the syslog daemon, you can define a syslog facility (USER/DAEMON LOCAL0..LOCAL7)	SysFacility
-I ident	When logging to the syslog daemon, you can define a syslog identifier. The default is eloqsd	SysIdent

*Equivalent configuration file directive.

The eloqsd HTTP status display

When the **ServiceHttp** is defined in the **`eloqsd.cfg`** configuration file, you can use a web browser such as Netscape or Mozilla to view the configuration and state of the eloqsd process in your network.

To access the eloqsd server, you need to provide an URL like below:

```
http://server:port/
```

where server is the host name or IP number of the system running the eloqsd server and port is the port number used for **ServiceHttp** in the **`eloqsd.cfg`** file.

Default eloqsd.cfg file

```
# eloqsd.cfg
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define the eloqsd properties.
# The location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.cfg
#   Linux: /etc/opt/eloquence6/eloqsd.cfg
#
# This file is read once at eloqsd startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
# Default values are provided commented out.

### Server configuration

[Config]

# Service       The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for requests. The default value is eloqsd.
#
# ServiceHttp   The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#
# UseKeepAlive  Numeric flag if the KEEP ALIVE socket option
#               should be used. Valid values are 1/0.
#               The default value is 1.
#               If this option is active, the server will check
#               after a system defined period of inactivity, if the
#               client is still alive.
#

#Service = eloqsd
#ServiceHttp =
#UseKeepAlive = 1

# panic         This option defines what should happen if a fatal
#               error is encountered.
#
#               The following options are valid:
#               exit    Terminate the process. This is the default.
#              dump    Terminate the process and create a core dump.
#
#            This is a problem tracking option. Unless you know what
#            you need the coredump for you probably want to stay with
#               the default

#panic = exit
```

```
# HttpFrame     Numeric flag if the links should be omitted in
#               HTTP status. The default value is 0.

# HttpFrame = 0

# Lang          This configuration option defines the locale, the
#               server should use. The default value is "C".
#               The only locale currently supported is "C".
#
# Messages      This configuration option defines the language
#               for server messages. This value defaults to Lang.
#               The only locale currently supported is "C".
#
# Charset       This defines the character set encoding, the server
#               should use internally.
#
#               Valid setings are:
#                  HPROMAN8 - HP Roman8
#                  ISO8859/1 - ISO 8859/1
#               The default value for HP-UX is HPROMAN8, all other
#               platforms default to ISO8859/1.
#
#               This setting is used by the server to translate
#               client strings like user or file names.

#Lang = C
#Messages = C
#Charset = HPROMAN8

# AuthPolicy    This entry specifies, how user names and passwords
#               are validated. The following entries are valid:
#
#               server  - The server will validate passwords
#                         and user names using eloqsd.user
#
#               The default value is "server".
#
# userFile      The path/name of the eloqsd.user file.
#               The default value depends on your operating system:
#               HP-UX: /etc/opt/eloquence6/eloqsd.user
#               Linux: /etc/opt/eloquence6/eloqsd.user
#
# shareFile     The path/name of the eloqsd.share file.
#               The default value depends on your operating system:
#               HP-UX: /etc/opt/eloquence6/eloqsd.share
#               Linux: /etc/opt/eloquence6/eloqsd.share

#AuthPolicy = server
#userFile = /etc/opt/eloquence6/eloqsd.user
#shareFile = /etc/opt/eloquence6/eloqsd.share

# DefaultUID   The default name (or numeric id) of the system account
#             to run client processes as, unless a different setting
#               is provided for the user.
#
# DefaultGID    The default name (or numeric id) of the system group
#             to run client processes as, unless a different setting
#               is provided for the user.

DefaultUID = eloqsd
```

```
DefaultGID = eloqsd

# LogFile       This defines where log messages are written to.
#              This configuration value either specifies a path/file
#               or one of the keywords below:
#
#               console  - log messages are written to the console
#               syslog   - log messages will be sent to the
#                          syslog daemon
#
#               The default value is "syslog".

LogFile = syslog

# SysIdent      When logging to the syslog daemon, you can define
#               a syslog identifier. Default is eloqsd.
#               See syslogd(1M) for more information
#
# SysFacility   When logging to the syslog daemon, you can define
#               a syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
#               The default setting is "USER".
#               See syslogd(1M) for more information

#SysIdent = eloqsd
#SysFacility = USER

# LogFlags      Each log message has an associated origin and
#               severity. The log flags define, which messages will
#             be logged. The "*" origin matches all message origins,
#               so it can be used to setup a default which can be
#              overriden for a specific message origin (eg. "*1N0"):
#               Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  C = Configuration subsystem
#                  N = Network transport
#                  P = Protocol handling
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#            Enabling log messages with L_DEBUG or L_VDEBUG severity
#               may result in a huge number of log messages.
#            To enable only fatal messages, you would want to set the
#              LogFlags to "*0", to enable regular log messages you
#               would want to set the LogFlags to "*1"

LogFlags = *0

# Configuration items below are the more traditional eloqsd
```

```
# settings.
#
# MaxUsers      Maximum number of eloqcore processes on the local
#               system. The default value is 40.
#
# MaxTasks     Maximum number of TASKIDs to reserve for "secondary"
#            eloqcore processes. If you don't know what this is good
#               for, you probably don't need it :-)
#               The default value is 20
#

MaxUsers = 40
MaxTasks = 20
```

Default eloqsd.user file

```
# eloqsd.user
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define all users which are known to
# Eloquence. The location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.user
#   Linux: /etc/opt/eloquence6/eloqsd.user
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# This makes it possible to define Eloquence users without the
# need to have a system account for each individual user.
# As passwords are defined in this file we consider it good practice
# to make this file unreadable for regular users. You should chown
# it to the administrator (probably root) and chmod id to 400.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each user definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [user_id]
# Name       The full user name (currently unused)
# Email      Email address of the user (currently unused)
# Password   The user password. This is currently clear text.
# uid        System account to execute client processes
# gid        System group to execute client processes
# Profile    Template user entry. User defaults will be taken from
#            this section.
# Home      Home path. Defaults to the home directory associated to
#            the UID by the system.
#
# There are two predefined sections:
#
# [public] is used, if a client does does not provide a user id. This
# can only happen, if an eloqcore has been started locally and
# requests a remote operation. (currently unused)
#
# [default] is used as the default user profile.

[public]
Name = Anonymous

[default]
Name = Default user profile

[demo]
Name = Jon Doe
```

```
Password = secret
```

Default eloqsd.share file

```
# eloqsd.share
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define all disk resources which are
# known to Eloquence. The location depends on the operating
# system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.share
#   Linux: /etc/opt/eloquence6/eloqsd.share
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# Eloquence A.06.00 provides its own file sharing capabilities.
# This will make you independent of the availability of specific
# network file systems (NFS/SMB) and overcomes possible file system
# limitations.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each share definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [share_id]
# Path      Absolute path
# Comment   Share description. This is displayed by the client.
# Users    Comma separated list of individual users or user profiles
#           (currently unused)

[example]
Path = /opt/eloquence6/share
Comment = Eloquence shared files
```

Configuring the eloqdb6 server

The eloqdb6 is the Eloquence database server. Eloquence uses a client/server database approach. Previous Eloquence A.05.xx databases can be accessed using the eloqdb5 server.

In order to run the eloqdb6 server it is required to adapt your system configuration. This involves the following steps:

- Configuring the eloqdb TCP service
- Configuring the default eloq account and group
- Configuring the eloqdb6 server startup
- Configuring the eloqdb6 server
- Creating the database environment

Configuring the eloqdb TCP service

It is recommended, that you define the eloqdb service names in your /etc/services file. This is optional, as you can specify the port number directly in the eloqsd.cfg configuration file.

Please add lines like below to your **`/etc/services`** file:

```
eloqdb      8102/tcp    # Eloquence A.06.00 eloqdb6 server
```

the first column specifies the service name (eg. eloqdb6) and the second column the associated port number and protocol (eg. 8102/tcp). The selected port numbers may not already be in use by another programs.

NOTE: All systems must use the same port numbers for the same service in order to communicate.

You can have more than one instance of the eloqdb6 server running on a single system, however they must use different services/port numbers.

NOTE: Multiple eloqdb6 instances are covered in detail in the section *Setting up multiple eloqdb6 instances* below.

Configuring the default eloqdb account and group

The eloqdb6 server requires you to specify an account and group name in the configuration file. Whenever it is started with root capabilities it will switch to the specified account/group instead. This is required, because for one it is generally

not a good idea to run programs with root capabilities unless necessary. In addition, all data base volumes are owned by this user and are thus protected from illegal access from other users.

While you can specify any user or group account in the configuration file, we recommend to create a specific user account and group for Eloquence which is used by the eloqdb6 server.

We recommend to create the user account *eloqdb* and the group *eloqdb* which should have the account *eloqdb* as a member. You can configure this with SAM. The account should be marked "disabled" in SAM (which will result in an asterisk in the password) to prevent logins using the eloqdb account.

Configuring the eloqdb6 server startup

If you want to specify that the eloqdb6 server is started at system boot time, set the **`START_ELOQDB6`** variable to **`1`** in the Eloquence startup/shutdown configuration file **`/etc/rc.config.d/eloquence6`**. Change the line defining the **`START_ELOQDB6`** variable like this:

```
START_ELOQDB6=1
```

Configuring the eloqdb6 server

The eloqsd server is configured by editing the **`eloqdb6.cfg`** configuration file. It is located in the **`/opt/eloquence6/etc`** directory. The eloqdb6.cfg configuration file provides complete inline documentation. The default configuration file is included at the end of this section for your reference.

The default configuration is not optimized for performance and does not handle a large number of concurrent connections. Therefore, you should adjust the following parameters in the eloqdb6.cfg configuration file to your requirements:

- Section **[Server]**, parameters **UID** and **GID**

  UID and GID must be set to the default eloqdb account as explained in the previous section *Configuring the default eloqdb account and group*.

- Section **[Server]**, parameter **EnableIPC**

  If the EnableIPC configuration item is set, eloqdb6 is enabled to make use of shared memory for communication between client and server. This results in a 25% performance increase but may require configuring the HP-UX kernel to provide additional resources. For details, please refer to the previous section *Configuring the HP-UX system*.

  The recommended setting is **EnableIPC = 2**.

- Section **[Config]**, parameter **Threads**

This parameter specifies how many connections to the database server can be established concurrently. The default is 40 which means that a maximum of 40 users can access the database at the same time. Multiple programs accessing the database count as multiple connections, while accessing multiple databases from within the same program counts as one single connection.

- Section **[Config]**, parameter **BufferCache**

  The recommended minimum value is 64 (megabytes). A higher value is recommended. The current limit is approx. 1 gigabyte. The default and minimum buffer cache size is 5 Megabytes which may lead to bad performance.

- Section **[Config]**, parameter **CheckPtSize**

  Whenever the transaction journal exceeds this size (in megabytes), eloqdb6 performs an internal checkpoint operation to recycle the journal. The default size is 10 megabytes which probably causes the checkpoint to happen too often, causing a performance impact. It is recommended to set this parameter to 20 or higher.

NOTE: If the configuration of a running eloqdb6 server is changed, it must be restarted to activate the new configuration.

Creating the database environment

A database environment consists of at least a primary data volume and a transaction log volume. These must be created on a local disk before the eloqdb6 can be started for the first time.

On HP-UX, the size of a single volume file is currently limited to 2 gigabytes.

Additional data volumes can be created at any time to extend the available database storage. Additional transaction log volumes can be added as well, but it is unlikely that these will ever be used because the purpose of a log volume is to hold temporary data only.

The maximum number of volume files in a database environment is 255 which is equivalent to a maximum database storage size of 510 gigabytes (254 data volumes and 1 log volume).

It is recommended to choose a dedicated directory to hold all volume files belonging to a particular database environment. The following example assumes that the initial volumes are created in the /data/db directory:

1. Change to the directory where the volumes shall be created:

   ```
   cd /data/db
   ```

2. Create the initial data volume:

   ```
   dbvolcreate data.vol
   ```

3 Create the initial transaction log volume:

```
dbvolextend -t log log.vol
```

Both files are created with their initial minimum size which is 2.5 megabytes. They will grow on demand until they reach their maximum size. The minimum and maximum size and the amount by which the files shall grow can be configured either at creation time or afterwards with the **dbvolchange** utility.

To view the available command line options, use:

```
dbvolcreate -help
dbvolextend -help
dbvolchange -help
```

The **dbvolcreate** and **dbvolextend** utilities automatically maintain the list of volume files in the **[Volumes]** section of the eloqdb6.cfg configuration file.

The eloqdb6 command line options

The eloqdb6 server supports the following command line options which can be used to temporarily override configured settings in the configuration file.

```
usage: eloqdb6 [options]
options:
 -help       = show usage (this list)
 -c name     = configuration file
 -d flags    = debug mode
 -l name     = log file name (or console/syslog/default)
 -f          = run in foreground
 -s name     = service name (tcp/ip transport)
 -F facility = syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
 -I ident    = syslog identifier
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-f	Run in foreground. This is used for debugging the eloqdb6 server.	
-s name	The service name (as defined in /etc/services) or the port number where the server should listen for requests. The default value is eloqdb6.	Service

Option	Description	Equiv.*
-F facility	When logging to the syslog daemon, you can define a syslog facility (USER/DAEMON LOCAL0..LOCAL7)	SysFacility
-I ident	When logging to the syslog daemon, you can define a syslog syslog identifier. The default is eloqdb6	SysIdent

*Equivalent configuration file directive.

The eloqdb6 HTTP status display

When the **ServiceHttp** is defined in the **`eloqdb6.cfg`** configuration file, you can use a web browser such as Netscape or Mozilla to view the configuration and state of the eloqdb6 server in your network.

To access the eloqdb6 server, you need to provide a URL like below:

```
http://server:port/
```

where server is the host name or IP number of the system running the eloqdb6 server and port is the port number used for **ServiceHttp** in the **`eloqdb6.cfg`** file.

Setting up multiple eloqdb6 instances

Multiple instances of the eloqdb6 server can coexist on the same system. This makes sense if multiple database environments should be kept separate from each other, either to keep the databases in separate namespaces or simply to maintain discrete database storage.

The following steps are required to setup an additional eloqdb6 server instance:

1 Each eloqdb6 instance needs a separate configuration file. You can copy either the configuration of any existing instance or the **eloqdb6.cfg** template file located in the **newconfig/config** subdirectory of your Eloquence installation.

In this example, we create a new configuration file named eloqdb6.instance.2.cfg in the default /etc/opt/eloquence6 configuration directory. This is not mandatory, the configuration file may have any name and can be located anywhere.

Please note that the **`cp`** command below must be entered as a single line:

```
cd /etc/opt/eloquence6
cp /opt/eloquence6/newconfig/config/eloqdb6.cfg
 eloqdb6.instance.2.cfg
```

Next, the new configuration file must be edited. Besides the parameters mentioned in the previous section *Configuring the eloqdb6 server*, the following parameters must be adjusted:

- Section **[Server]**, parameter **Title**

  The server title is displayed when you list the eloqdb6 processes with **`ps`**. You should specify an unique title for each eloqdb6 instance to be able to distinguish the different instances in the process listing.

- Section **[Server]**, parameter **Service**

  Each instance requires an unique TCP service name or port number. You can either configure a dedicated service name in your local SERVICES file (please refer to the previous section *Configuring the eloqdb6 TCP service* for details) or simply enter an unique port number.

- Section **[Server]**, parameter **ServiceHttp**

  If you use the eloqdb6 HTTP status display, you are required to configure an unique TCP service name or port number for each eloqdb6 instance.

- Section **[Server]**, parameter **LogFile** (optional)

  By default, all eloqdb6 log messages are written to the syslog. Each log message is labeled with the process id of the originating eloqdb6 instance. However, it might be more convenient to configure a separate log file for each instance. This log file could be located within the dedicated, instance-specific directory (see below), for example:

  ```
  LogFile = /data/db/instance.2/eloqdb6.log
  ```

NOTE: If you copied the configuration of an existing instance, it is required that you manually delete all volume references at the end of the file below the **[Volumes]** section header.

2 Create the instance-specific database environment. You do this according to the previous section *Creating the database environment*, but you use the **`-c`** command line option to refer to the instance-specific configuration file:

- Create the instance-specific directory. We recommend to create a dedicated directory for each instance where the instance-specific files are located:

  ```
  cd /data/db
  mkdir instance.2
  cd instance.2
  ```

- Create the initial data volume (the following command must be entered as a single line):

  ```
  dbvolcreate
   -c /etc/opt/eloquence6/eloqdb6.instance.2.cfg data.vol
  ```

- Create the initial transaction log volume (the following command must be entered as a single line):

```
dbvolextend
 -c /etc/opt/eloquence6/eloqdb6.instance.2.cfg -t log log.vol
```

In this example, the argument to the **-c** command line option refers to the instance-specific configuration file located in the current directory.

3 Create an entry for the new eloqdb6 instance at the end of the **/etc/rc.config.d/eloquence6** startup/shutdown configuration file:

```
ELOQDB6_CFG[1]=eloqdb6.instance.2.cfg
ELOQDB6_START[1]=1
ELOQDB6_ID[1]=instance.2
```

Use the next available index if there is already an instance defined which uses the index **[1]**. ELOQDB6_CFG refers to the instance-specific configuration file. If this is located outside the default /etc/opt/eloquence6 directory, the absolute path must be provided. If you don't want this instance to be started at system boot time, set the ELOQDB6_START variable to 0.

The new eloqdb6 instance is now configured. You can control it using the **/sbin/init.d/eloq6** startup/shutdown script as follows:

```
/sbin/init.d/eloq6 start instance.2
/sbin/init.d/eloq6 status instance.2
/sbin/init.d/eloq6 restart instance.2
/sbin/init.d/eloq6 stop instance.2
```

To access a database in an additional instance, specify the instance-specific service name or port number in addition to the database name. Those utilities which do not expect a database name instead provide the **-h** command line option. Also, the EQ_DBSERVER environment variable can be set to specify the default instance.

The following examples assume that an additional instance was configured to use the port number 8201. They illustrate different ways to address the same instance:

```
schema -h :8201 db.schema.txt
dbcreate :8201/db
```

```
EQ_DBSERVER=:8201
export EQ_DBSERVER
schema db.schema.txt
dbcreate db
```

Default eloqdb6.cfg file

```
# eloqdb6.cfg
# @(#) $Revision: B.07.00.2 $
#
# This file defines the eloqdb6 configuration and the database
# environment.
# The default location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqdb6.cfg
#   Linux: /etc/opt/eloquence6/eloqdb6.cfg
#
# This file is read once at eloqdb6 startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.


### Server configuration

[Server]

# Title         If set, a server title is displayed by the ps
#               program in the eloqdb6 command line instead of the
#               default "eloqdb6: active" If you are using multiple
#              eloqdb6 server processes on a single system this can
#               be used to distinguish between different server
#               instances. The default value is empty.

#Title =

# Service       The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for requests. The default value is eloqdb.
#
# ServiceHttp   The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#

#Service = eloqdb
#ServiceHttp =
```

```
# panic         This option defines what should happen if a fatal
#               error is encountered.
#
#               The following options are valid:
#               restart Restart the server process (default)
#               exit    Terminate the process.
#              dump    Terminate the process and create a core dump.
#
#             panic = dump is a problem tracking option. Unless you
#             know what you need the coredump for you probably want
#               to stay with panic = exit or panic = restart

#panic = restart

# UID           The name (or numeric id) of the system account to
#               run client processes as when started as root.
# GID           The name (or numeric id) of the system group to run
#               client processes as when started as root.
#
#               Please note, that the server will refuse to start
#               as root unless UID and GID are valid.

UID = eloqdb
GID = eloqdb

# EnableIPC     If set, shared memory can be used to transmit data
#               between the database server and a client running on
#               the same system. This provides better performance
#               than using sockets because data are not passed
#               through the kernel.
#               The default value is 0 (disabled).
#
#               The following options are supported:
#
#               EnableIPC = 0 (default)
#                 Disables use of shared memory communication.
#
#               EnableIPC = 1
#                 Enables use of shared memory communication.
#                This mode uses a separate memory segment for each
#                 connection.
#
#               EnableIPC = 2 (recommended)
#                 Enables use of shared memory communication.
#                 This mode uses a common memory segment for all
#                 connections.
#
#               When setting EnableIPC configuration of kernel
#               parameters may be required.
```

```
#EnableIPC = 0

# SyncMode      If set, this causes the eloqdb6 server to operate in
#                sync write mode. The sync write mode is more
#                resistent against operating system and hardware
#                failures. When sync mode is disabled (set to 0) the
#                eloqdb6 uses the faster async  write strategy which
#                performs fewer disk writes but could lead to a
#                damaged database environment in case of a system
#                failure.
#                The default value is 1 (sync write mode enabled).

#SyncMode = 1

# LogFile       This defines where log messages are written to.
#               This configuration value either specifies a path/
#               file or one of the keywords below:
#
#               console  - log messages are written to the console
#               syslog   - log messages will be sent to the
#                          syslog daemon
#
#               The default value is "syslog".

#LogFile = syslog

# SysIdent      When logging to the syslog daemon, you can define
#               a syslog identifier. The default is eloqdb.
#               See syslogd(1M) for more information
#
# SysFacility   When logging to the syslog daemon, you can define
#               a syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
#               The default setting is "USER".
#               See syslogd(1M) for more information

#SysIdent = eloqdb
#SysFacility = USER

# LogFlags      Each log message has an associated origin and
#               severity.
#              The log flags define, which messages will be logged.
#             The "*" origin matches all message origins, so it can
#               be used to setup a default which can be overriden
#               for a specific message origin (eg. "*1N0"):
#               Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  A = Configuration subsystem
#                  X = Network transport
```

```
#                  P = Protocol handling
#                  T = Thread kernel
#                  I = IMAGE subsystem
#                  B = BTREE subsystem
#                  F = FIXREC subsystem
#                  V = Volume handling
#                  L = Transaction logging
#                  C = Page cache
#                  N = Node handling
#                  D = The server framework
#                  O = System catalog
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#               Enabling log messages with L_DEBUG or L_VDEBUG
#              severity may result in a huge number of log messages.
#              To suppress anything but fatal messages, you can set
#               LogFlags to "*0". To enable informational log
#               messages LogFlags should be set to "*1".

#LogFlags = *0

# HTTPUser      The eloqdb6 server is able to display status
#              information by supporting the HTTP protocol (you can
#              use Netscape to monitor the database server process,
#               see ServiceHttp above).
#              If set, the eloqdb6 HTTP status display will require
#               a matching user name (HTTP basic authentification)
#               before allowing access to the eloqdb6 HTTP status.
#               The default value is empty.
#
# HTTPPswd     If set, the eloqdb6 HTTP status display will require
#               a matching password (HTTP basic authentification)
#               before allowing access to the eloqdb6 HTTP status.
#               The default value is empty.

#HttpUser =
#HttpPswd =
```

```
# HTTPFrame     If set, no link information is output on the HTTP
#               status display. So the status page could be used in
#               a web frame.
#               Default value is 0.

#HttpFrame = 0


### Data base configuration

[Config]

# Threads       Number of threads in the data base server.
#               A separate thread is required for each client.
#               Default number of threads is 40.

#Threads = 40

# IOThreads     This specifies the number of I/O threads which are
#               used by eloqdb6. eloqdb6 uses either separate
#               processes or kernel threads to perform overlapped
#               I/O operations. Please note that the eloqdb6 I/O
#               threads are visible with the ps command.
#               The default value is 4. A zero value disables usage
#               of IO threads.

#IOThreads = 4

# LockConflictingItems              If set, predicate locks with
#               conflicting items are granted, however any write
#               attempt to data where another process owns a lock
#               will result in a status error -12.
#              Former Eloquence revisions rejected a predicate lock
#               with a conflicting item, because this could lead to
#               a situation where two processes own a lock on an
#               overlapping subset of data. The default value is 0.

#LockConflictingItems = 0

# AllowSecondaryBlockingLock        If set, secondary blocking
#               locks are allowed. In previous Eloquence versions,
#               secondary locks in a blocking mode (odd modes)
#               failed with database status -135 ("Second lock is
#               not allowed in modes 1,3,5,11,13 and 15.") instead
#               of blocking. Current Eloquence versions return the
#               status code -35 in case a deadlock situation caused
#              by a secondary blocking lock is detected. Therefore,
#               this setting is enabled by default. To retain the
#               behavior of previous Eloquence versions it can be
#               set to 0. The default value is 1.
```

```
#AllowSecondaryBlockingLock = 1

# BufferCache   Size of page cache in megabytes. The page cache is
#               used to reduce the number of disc accesses. Large
#               cache size will speed up random database access,
#               while a too small cache size may cause bad server
#               performance.
#               Default cache size is 5 MB.

#BufferCache = 5

# The server performs a checkpoint operation at fixed intervals.
# This flushes all modified buffers (including metadata) to the
# disk and resets the log of committed transactions. A checkpoint
# is a point where the server knows all data are in a consistent
# state. Any data modification since the last checkpoint is
# recorded in the log volume.
#
# CheckPtFreq   Checkpoint frequency in seconds.
#               Default checkpoint frequency is 60 seconds.
#
# CheckPtSize   Checkpoint frequency based on accumulated log space
#              which would be freed by a checkpoint (in megabytes).
#               A zero CheckPtSize value disables size based
#               checkpoints.
#               Default checkpoint size is 10 megabytes.
#
# The database server performs a checkpoint operation at a fixed
# interval and optionally in addition when the accumulated log
# space which could be freed by a checkpoint operation reaches a
# given threshold.
# The frequency of the checkpoint operations has a great influence
# on the size of the log volume since the log volume must hold all
# committed transactions since between checkpoints

#CheckPtFreq = 60
#CheckPtSize = 10

# The syncer thread flushes modified buffer pages to the disk when
# they are likely to become reused in the near future.
#
# SyncerFreq    Syncer thread invocation frequency (in seconds)
#               Default interval is 5 seconds.

#SyncerFreq = 5

# SyncerJournalFlushInterval        If SyncMode is enabled this
#               configuration item specifies the interval (in
#               milliseconds) at which the journal of committed
```

```
#               transactions is synchronized to disk.
#               In case of an operating system or hardware failure
#              transactions that were not synchronized to disk are
#               typically lost.
#              A smaller value reduces the amount of transactions
#               that might be lost in case of a system crash.
#               However, setting this value too low significantly
#               impacts write performance.
#               Setting this value to 0 reverts to the legacy
#               SyncMode behavior where every transaction is
#               immediately synchronized.
#               The default value is 500 milliseconds.

#SyncerJournalFlushInterval = 500


### Store/Restore Devices

[Devices]

# This section defines the "server devices" which can be used with
# dbstore and dbrestore. Each entry consists of the device name and
# an associated path.
#
# A "server device" could either be a single file, a directory or a
# device. When no server devices are configured, dbstore and
# dbrestore operation is refused by the server.
#
# The example below defines two server devices. The device "Tape"
# points to a tape device file, the device "Backup" points to a
# directory which is intended to hold the backup files.

#Tape = /dev/rmt/c1t0d0BEST
#Backup = /data/backup


### Forward log

[ForwardLog]

# FwLog         Configures the file, device or pipe to be used for
#              forward-logging. Using the %N token in the file name
#               activates automatic file management (not possible
#               for devices or pipes).
#               By default, forward-logging is inactive.
#
#               The examples below configure an automatically
#               managed file and a pipe which compresses the data
#               on-the-fly:
```

```
#FwLog = /mnt/disk2/data/db-forward-%N.log
#FwLog = |gzip -c >/mnt/disk2/data/db-forward.log.gz

# FwRecovery    Configures the file, device or pipe to be used
#              during forward recovery. If not set, the Log setting
#               is used by default.
#
#               The example below configures a pipe which
#               uncompresses the data on-the-fly:

#FwRecovery = |gzip -dc /mnt/disk2/data/db-forward.log.gz

# FwOnFailure   Configures the action to be taken in case the
#               forward-log cannot be written, e.g. due to
#               insufficient disk space.
#               Possible values are disable or panic. If set to
#               disable, forward-logging will be disabled on
#               failure. As soon as the problem is solved it can be
#               manually enabled using dbctl.
#               If set to panic, the eloqdb6 server will issue a
#               panic and abort itself.
#               The default value is disable.

#FwOnFailure = disable

# FwMaxSize     Limits the maximum size of automatically managed
#               forward-log files (in megabytes). If not set or set
#               to zero, the file size limit is 2 gigabytes. The
#               default value is 0 (not set).

#FwMaxSize = 0


### Data base environment

[Volumes]

# List of data base volumes. Initially empty.
# This is usually filled in by dbvolcreate and dbvolextend
# utilities
```

Customizing the Eloquence Configuration Files

This discussion assumes that the Eloquence software has already been installed on your system. The information in this section is directed to the system administrator for the Eloquence software.

NOTE: The configuration steps mentioned here are not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you can skip this section.

Before Eloquence can be used, its resources must be configured. Eloquence programs usually don't use system resources directly, instead they rely on a mapping of pathes, printers and device files in Eloquence configuration files.

There are three different levels of configuration:

System global — This is achived with the **`eloq.config`** configuration file which is located in the Eloquence configuration directory.

Group specific — This is achived with the **`group.<GroupName>`** configuation file which is located in the Eloquence directory.

User specific — This is achived with the **`.eloqrc`** configuration file which is located in the home directory of the user.

The Eloquence configuration files are read by the eloqcore process, when it is started. The configuration files are processed in an order such that more specific definitions override the more general ones. So a system global assignment can be overriden from a group specific configuration file, a user specific definition will override group and system global definitions.

The system global configuration file, **`eloq.config`** is usually copied during the installation process to Eloquence configuration directory and should be adapted to local requirements. Template configuration files are provided in the directory **`/opt/eloquence6/newconfig/config`**. The template configuration files provide comlete inline documentation and are included at the end of this section for your reference.

Eloquence resource configuration

Eloquence resources go back to the "dark ages" when a precedessor of Eloquence was implemented in hardware (called HP250/HP260 at that time) and the resources definition actually were real OS resources. Since programs depended on a program independent resource configuration and it a convenient mechanism

anyway, the concept was kept. Instead of real devices Eloquence resources can be mapped to whatever is appropriate. Eloquence is of course able to access native operating resources directly.

Since the following names are not commonly used, let's define them first:

VOLUME A **VOLUME** is the Eloquence concept of a directory. Instead of using the path directly, it is possible to assign an identifier for a path and refer to it in a symbolic manner.

MSI This is a short form of **MASS STORAGE IS** and species the default **VOLUME** on which pathes should be related unless an absolute path or another **VOLUME** is given.

PRINTER A **PRINTER** is the Eloquence concept of an output depvice. A PRINTER is identified by a number and could be mapped to a device file or to a sequence of commands.

The device numbers 8 to 10 have a special predefined meaning:

8: Display terminal.
9: Bit bucket (Eloquence equivalent of /dev/null)
10: Local terminal printer

PORT A PORT is the Eloquence equivalent of a (tty) device file. Eloquence provides prowefull machanisms to handle them in a efficient manner.

The eloq.config configration file

The eloq.config file provides system global definitions and is usually copied during the installation process into the Eloquence configuration directory from the template file **`d.eloq.config`**.

The group specific configuration file

To provide group specific definitions, you could install a group specific configuration file in the Eloquence configuration directory. Consider we would like to have a specific configuration for the *sales* group you would perform the following steps:

1. Change to the Eloquence confuration directory:

   ```
   cd /etc/opt/eloquence6
   ```

2. Create a group specifc configuration file from the template group configuration file. The group specific file should be named group.*sales*.

```
cp /opt/eloquence6/newconfig/config/d.group group.sales
```

3 Use a text editor, such as vi to edit the file

```
vi group.sales
```

The user specific configuration file

To provide user specific definitions, you could install a user specific configuration file in the gome directory of the user. Consider we would like to have a specific configuration for the user *mike*, you would perform the following steps:

1 Change to the home directory of the user:

```
cd ~mike
```

2 Create a user specifc configuration file from the template user configuration file. The user specific file should be named .eloqrc:

```
cp /opt/eloquence6/newconfig/config/d.eloqrc .eloqrc
```

3 Use a text editor, such as vi to edit the file

```
vi .eloqrc
```

Template eloq.config file

```
# d.eloq.config
# Eloquence configuration file
# (C) Copyright Marxmeier Software AG, 2002
# @(#)$Revision: 20.4 $
#
# This file contains global available configuration
# It must be named eloq.config and located at:
#   HP-UX 9.x    /opt/eloquence6/etc
#   HP-UX 10.x   /etc/opt/eloquence6
#   linux        /etc/opt/eloquence6
#
#  PLEASE NOTE:
#  You MUST define at least one volume (typically SYSTEM, see below),
#  or eloqcore will fail on startup.
#
# Globally defined volumes
#
# Format: VOLUME label [device] path
#
#      label   - Volume label (up to 8 characters)
#                must be unique per file
#      device  - Device specifier eg. ":F2,6,0"
#                ignored when present, no longer used
#      path    - HP-UX path to map volume on
#
# Globally defined printers
#
# Format: PRINTER no [model] type spec
#
#      no     - printer select code (-2 .. 7, 11 .. 99)
#      model  - PCL or OTHER
#                ignored when present, not used
#      type   - printer type PIPE, FILE or SYSTEM
#      spec   - path/command to process on printer selection
#
# Globally defined ports
#
# Format: PORT no spec
#
#       no    - port select code (11 .. 20)
#               may not conflict with PRINTER
#       spec  - path of tty devicefile
#
#
# Default date/time format
#
# Format: DATE spec
#         TIME spec
#
#      spec    - date/time specification. please refer to date(1) or
#                   strftime(3) for more information.
#                For backward compatibility, the former specifications
#                   "DD.MM.YY" and "MM/DD/YY" are silently converted.
#
#                   Default: DATE "%m/%d/%y"
```

```
#                           TIME "%H:%M:%S"
#
# Global MSI value
#
# Format: MSI label
#
#      label  - Volume label. Default is the first defined volume.
#

# --- sample volumes

VOLUME SYSTEM  /opt/eloquence6/share/prog
#VOLUMEEXAMPLE /opt/eloquence6/share/example

# --- sample printers

PRINTER 0  PIPE "lp -s 2>/dev/null"
#PRINTER 1  PIPE "lp -s -oc 2>/dev/null"
#PRINTER 2  FILE /dev/lp

# --- sample PORT

#PORT 11 /dev/tty0p5
```

Configuration of the GUI Dialog Server

The GUI dialog server allows Eloquence programs to apply a graphical user interface. The process about how the Eloquence program (i.e. the **`eloqcore`** process) connects to the Run and Dialog servers is configured in the **`eloq.ini`** configuration file.

Customization of the eloq.ini file

The **eloq.ini** configuration file is located in the /**etc/opt/eloquence6** directory.

The installation program does not overwrite an existing configuration file but installs the **/opt/eloquence6/newconfig/config/eloq.ini** file instead.

The ‘ini‘ File format

The **`eloq.ini`** and **`eloqcl.ini`** files contain several sections each containing a group of related configuration items. The sections and configuration items have the following format:

```
[Section]
Item=Value
```

Section is the name of a section. The enclosing brackets ([ ]) are required and they must start at the first column. *Item=Value* defines a value of a configuration item. *Item* is the name of a configuration item. It consists of any sequence of characters (case insensitive) and digits followed by an assignment operator (=). Depending on the item type, the value may either be numeric or alphanumeric (optionally enclosed in double quotes). Comment lines must start with a semicolon (;) or a hash character (#) in the first column.

The eloq.ini file

Section [runsrv]

This section specifies the defaults used with the **`runsrv`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: runsrv

Function: The default service name or port number to be used with the **`runsrv`** connection protocol. If no value is defined the port associated with the service runsrv is used.

Section [eloqdlg]

This section specifies the defaults used with the **`dlg`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: 8011

Function: The default service name or port number to be used with the **`dlg`** connection protocol. If no value is defined the port number 8011 is used.

Mapping Driver Specifications

Other sections in the **eloq.ini** file can be used to map a driver specification to a different value. When a driver is specified (e.g. in the **`DLG SET ".driver"`** statement or with the **`eloqcore -dlg`** command line option) the argument is looked-up in the **eloq.ini** configuration file after any protocol specification has been removed. If a section has been found it is used to replace the original argument.

The following configuration items are supported:

dlg

dlg= *replacement text*

Function: When contacting a DLG driver this is used as a replacement for the specified driver. The value can specify a different protocol.

host

host= *[hostname][:service]*

Function: When using the Run server to execute a remote operation or to start the Dialog server the host variable can be used to specify a different hostname (or IP address) and/or a different service name (or port number) which should be used instead.

Example

```
[runsrv]
Service = runsrv

[eloqdlg]
Service = 8011

[mike]
dlg = dlg://lxmike

[chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```

This example specifies to use a port number for the **runsrv** protocol which is associated with the service runsrv (as defined in the local SERVICES file). For the **dlg** protocol the port number 8011 should be used.

When a driver argument "mike" is specified (e.g. "**@mike**", "**dlg://mike**" or "**runsrv://mike**") the value "**dlg://lxmike**" is used instead. When a driver argument "chris" is specified it is replaced by "**runsrv://wserv:8765**". When the Run server is contacted from either the command line (using the **runclnt** utility) or the **RunSrv.DLL** and a remote name "chris" is passed, the value "**wserv:8765**" is used instead.

Running multiple runsrv instances on a single system

This is required if the Citrix Metaframe or the Windows Terminal Server product is used. Each user needs to start the RUNSRV32 with a different port number (probably during autostart). Then a driver name containing the host name and the user name is passed and translated by using the mapping defined in **eloq.ini**.

Suppose the user "chris" has a RUNSRV32 using port 8765 when working on system wserv, then the following mapping section should be added in **eloq.ini**:

```
[wserv_chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```

3

Installing Eloquence on Windows

This chapter covers the installation of the Eloquence software on Windows Server 2003, XP, 2000, NT, ME, 98 and 95.

- Software and patch installation
- Configuring the operating system
- Configuring Eloquence

Installation on Windows

This document provides installation instructions for Eloquence on Windows Server 2003, XP, 2000, NT, ME, 98 and 95.

Installation Prerequisites

The following prerequisites must be met before installing the Eloquence software on a system. Failing to do so will either cause the installation process to fail or will prohibit the successful use of Eloquence.

- Eloquence makes use of the TCP/IP protocol to interchange data between different processes even when used on the local computer only. No network card or dialup connection is required, but the TCP/IP protocol stack must be installed and configured. In case your system does not have a network card installed you can use the following configuration:

Windows NT On Windows Server 2003, XP, 2000 and NT you can install the *Microsoft Loopback Adapter*. You are required to configure an IP address for the Loopback Adapter. We recommend to use an IP address reserved for internal usage (such as 192.168.1.1 with subnet mask 255.255.255.0) as defined by RFC 1597.

Windows 9x On Windows ME, 98 and 95 you can configure the *Dial-Up Adapter* as some kind of generic device. This is sufficient to bind the TCP/IP protocol. No further configuration is required.

- As usual, all applications should be closed before installing or upgrading Eloquence. The setup program may need to update shared components on your system and may require a reboot in order to do so.
- The Eloquence server software can only be installed on the Windows Server 2003, XP, 2000 and NT operating systems. On Windows ME, 98 and 95, the installation program will disable all software packages which are not compatible. On Windows XP Home Edition the eloqsd service cannot be used because the Windows Group Policy Editor is not available on XP Home.
- When installing Eloquence on Windows Server 2003, XP, 2000 or NT, administrative capabilities are required, otherwise the installation program will refuse to run.
- We do not recommend to install the Eloquence HTML documentation (reference manuals) on a FAT filesystem. Since the documentation includes more then 1000 rather small files this may take considerable space on a FAT filesystem depending on cluster size. Instead, we recommend to install the Eloquence HTML documentation on a web server in your local network.

The Installation Program

The Eloquence installation program is based on the Microsoft Windows Installer.

Introduction

The Microsoft Windows Installer is a system component initially shipped with Windows 2000. It interoperates with so-called MSI files which are actually databases holding all information about any installed product. Therefore, the Windows Installer is always aware about all installed software on the system and the several components installed with any software. This makes it easy to keep the system in a clean state. Additionally, the Windows Installer can take the necessary steps to repair a particular installation or to uninstall a particular product.

Eloquence B.07.00 provides the **eq70.msi** file which holds the installation database. The components to be installed are divided into separate archives which are laid-out in a compressed format. The contents of the archives are listed below.

Components

The different Eloquence components can be installed independently. Each component is contained in a separate, self-contained archive file. The installation program expects to find the archive files in the same directory where the **eq70.msi** file is located. It lets you choose among the present archive files which components are to be installed.

Due to a limitation in the Microsoft Windows Installer implementation the archives have short DOS 8.3 file names.

1 **Client GUI Environment**

Archive: **eq70_01.cab**

contains the runsrv32 and dlgsrv32 graphical user interface components necessary to execute any local or remote GUI-enabled Eloquence program. Such a program can conveniently be started with either the eqstart or eqexec32 utilities which are both included as well. This is the minimum client installation.

2 **Client Runtime Environment**

Archive: **eq70_02.cab**

contains the eloqcore program which enables you to locally execute Eloquence programs. Additionally, the Eloquence configuration utility is included. If your programs all reside on remote systems it is not necessary to install this archive.

3 Client Developer Tools

Archive: **eq70_03.cab**

contains the Eloquence Integrated Development Environment (the IDE) as well as the list and store utilities. If you do not modify your programs on or from the particular system it is not necessary to install this archive.

4 Client Database Tools

Archive: **eq70_04.cab**

contains the client database tools necessary to create and maintain Eloquence databases, such as schema, dbcreate, dbimport and dbutil. If you do not maintain your databases on or from the particular system it is not necessary to install this archive.

5 Client Database Library

Archive: **eq70_05.cab**

contains the client database library which is required to access Eloquence databases from within programming languages other than Eloquence Basic. This archive also contains the Eloquence Image3K TurboIMAGE compatibility extension.

6 Server

Archive: **eq70_06.cab**

contains the eloqdb6 database server as well as the eloqsd service. If the particular system runs Windows Server 2003, XP, 2000 or NT and is intended to act as an Eloquence database and/or application server this archive must be installed. Since the Windows Service Control Manager is not available on Windows ME, 98 or 95 the server components are not installed unless your system runs Windows Server 2003, XP, 2000 or NT.

7 On-line Documentation

Archive: **eq70_07.cab**

contains the core on-line documentation and release notes and should be installed along with the client_runtime and/or client_developer archive files.

8 Reference Manuals (html)

Archive: **eq70_08.cab**

contains the entire manual set in HTML format. Since these are a lot of small files you should consider to install this archive in a central location and integrate it into your intranet web server's document hierarchy. By default, this archive is not installed. However, if you select to install this archive it will automatically integrate into the on-line documentation.

9 Reference Manuals (pdf)

Archive: **eq70_09.cab**

contains the entire manual set in PDF format to be viewed with the Adobe Acrobat Reader (TM) which is included on this media and freely available on the Internet. By default, this archive is not installed.

By default, all components except the manuals are installed. The installation program detects which archive files are present (e.g. if not all components are required and therefore only a subset has been downloaded) and will not offer the related missing components for installation.

CD-ROM Installation

To install Eloquence from the CD-ROM media, open the **\B0700\win32** directory and use the **setup.exe** program contained in this directory.

Download Installation

Eloquence can be downloaded from the Eloquence web site at:
http://www.marxmeier.com/eloquence/download

1 Download the appropriate setup executable depending on your operating system:

- **eq70_win2k.exe**

  for the Windows Server 2003, XP or 2000 platforms

- **eq70_winnt.exe**

  for the Windows NT platform

- **eq70_win9x.exe**

  for the Windows ME, 98 and 95 platforms

Unless you plan to install Eloquence B.07.00 on several machines and different platforms you need only one of these three files.

The setup executables for Windows NT, ME, 98 and 95 are about 1.4 MB bigger than the one for Windows Server 2003, XP and 2000 because they additionally include the Microsoft Windows Installer 1.2 engine which is a Windows Server 2003, XP and 2000 system component.

2 Download the archive files of the components you want to install and put them into the same directory where the setup executable is located. At least one archive file must be present, otherwise the installation program will not install anything.

The installation detects which archive files are present and offer the related components for installation.

NOTE: You can use the **eq70_all_cab.zip** file to download all archive files at once. After downloading, unpack its contents into the directory where the setup executable is located.

NOTE: A download installation requires write permissions in the directory where the setup executable is located along with the downloaded archive files. If you intend to use the downloaded files on a CD or DVD you first have to unpack the self extracting eq70_*.exe setup executable. It will extract the files setup.exe, eq70.msi and some .ini files. Afterwards, the setup.exe will be started which you can immediately cancel. Then copy the extracted files, not the eq70_*.exe setup executable, to your CD or DVD.

Installation instructions

1 On Windows Server 2003, XP, 2000 or NT please log-on as Administrator before you install Eloquence (Windows "power user" privileges are sufficient).

2 If you already have Eloquence installed:

- Please make sure that no Eloquence program is running (e.g. stop the Run server and exit from the IDE). You don't have to stop the services (eloqsd, eloqdb6), though, because the installation program will appropriately stop and uninstall them.
- The Eloquence B.07.00 installation program will automatically detect and uninstall any existing Eloquence software.

  Your existing configuration files will be preserved.

  It is not necessary and also not recommended to manually uninstall your existing Eloquence software.

  By default Eloquence B.07.00 uses an installation directory different from older Eloquence versions up to A.06.31. The installation program will automatically migrate your existing configuration files to the new installation directory.

  If you, however, have Eloquence B.06.32 or above installed the new version will be installed into the existing installation directory by default.

  Please read the section about updating your Eloquence software below.

3 Now start either the **setup.exe** installation launcher program or the self-extracting **eq70_win2k.exe**, **eq70_winnt.exe** or **eq70_win9x.exe** setup executable, depending if you do a CD-ROM or download installation.

The installation program should start and guide you through the installation process. The Custom Setup dialog lets you choose which components you want to install. By default, all components (minus those you did not download) except the manuals are selected for installation.

On Windows Server 2003, XP, 2000 or NT you can choose to install a temporary server license if Eloquence B.07.00 is installed for the first time. This option is enabled by default. You can disable this option manually if you don't want to install

the temporary server license now. Later, you can use the Software applet in the Windows Control Panel to modify your Eloquence B.07.00 installation which will allow you to install the temporary license subsequently.

4 After installation on Windows Server 2003, XP, 2000 or NT a B.07.00 license key is required in order to use the Eloquence server components. During installation a temporary license key is generated on demand which is valid until 30 days after installation.

Please visit the Eloquence Web site at **http://www.marxmeier.com/eloquence/license** for details about obtaining a permanent Eloquence license key.

Your license file is located in the **etc** subdirectory of your Eloquence installation, for example at **C:\Program Files\Eloquence\etc\license**.

Please comment out any previous license key when adding the new B.07.xx license key to your license file. Otherwise the new license may not be recognized and chklic might output the following messages:

```
B.06.32: Bad license key revision.
Duplicate sequence number: Ignoring license
```

If you're using the Eloquence Personal Edition, please use the license template file **license.sam**, it contains an updated license key for the Eloquence Personal Edition.

The **license.sam** file is located in the **etc** subdirectory of your Eloquence installation and should be copied into the **license** file located in the same directory.

Updating your existing Eloquence software

If your existing Eloquence software is version B.06.32 or above it will be uninstalled automatically. The new version will be installed into the existing installation directory. This will preserve your configuration files.

If you, however, have HP Eloquence A.06.31 or below installed please note:

- By default, Eloquence B.07.00 is installed into a different location than HP Eloquence versions up to A.06.31. The new location is **C:\Program Files\Eloquence** as opposed to the previous location which was:
  C:\Program Files\Hewlett-Packard\HP Eloquence.
- The installation program will automatically migrate your configuration and license files from a previous HP Eloquence installation. This will take the new software location into account so it is not necessary to install Eloquence B.07.00 into the same directory where your previous HP Eloquence software is installed.
- The installation program will also uninstall your previous HP Eloquence software. It is not necessary to uninstall it yourself. In fact, this would prevent the installation program from detecting your previous HP Eloquence version.
- Uninstallation takes place before the new software is copied to your disk. For HP Eloquence A.06.3x, the uninstallation process is visually indicated. Versions before

A.06.30 are however silently uninstalled. Please be patient during uninstallation.

- All standard HP Eloquence settings in the Windows registry are migrated as well. While the HP Eloquence root key is migrated by the installation program the application-specific settings are migrated by the respective programs, i.e. the Eloquence IDE, the Eqstart program and the Configuration utility.
- However, in case you manually created non-standard registry settings these must be manually migrated. The new Eloquence registry key is **HKEY_LOCAL_MACHINE\SOFTWARE\Marxmeier\Eloquence**.
- As in previous HP Eloquence versions, the corresponding **HKEY_CURRENT_USER** key is searched first to allow overriding the defaults with user-specific settings.

The Eloquence B.07.00 server components require a new license key. Previous A.06.xx and B.06.xx license keys do not work anymore. To migrate your existing license, first install the temporary license, then visit the Eloquence Web site at **http://www.marxmeier.com/eloquence/license** to obtain a new license key.

Changes to previous Eloquence installations

- **Migration and uninstallation of previous software**

  If you have a HP Eloquence version A.06.31 or below installed its configuration and license files are migrated to Eloquence B.07.00. The previous software is uninstalled afterwards.

- **Default installation location**

  As opposed to HP Eloquence versions A.06.31 and below, the default Eloquence B.07.00 installation directory has been changed to **C:\Program Files\Eloquence**.

- **Registry root key**

  As opposed to HP Eloquence versions A.06.31 and below, the Eloquence B.07.00 registry root key has been changed to **HKEY_LOCAL_MACHINE\SOFTWARE\Marxmeier\Eloquence**.

- **Default server names**

  As opposed to HP Eloquence versions A.06.31 and below, the default Eloquence server names in the Windows Service Control Manager have been changed. By default, the database server is named "eloqdb6" and the application server is named "eloqsd" (formerly "HP EloqDB6" and "HP EloqSD", respectively).

- **Component archives**

  As opposed to HP Eloquence versions A.06.31 and below, the archive files eq70_04.cab and eq70_05.cab have been added to separate the database tools and libraries from the runtime and development archives.

The eq70_05.cab archive also contains the Eloquence Image3K TurboIMAGE compatibility extension.

Due to a limitation in the Microsoft Windows Installer implementation the archives have short DOS 8.3 file names.

- **Automatic stop and uninstallation of Eloquence servers**

  Before installation or uninstallation begins, all running Eloquence servers (eloqdb6, eloqsd) are stopped and all present Eloquence servers are removed from the Windows Service Control Manager.

  After installation has finished, the default eloqdb6 and eloqsd servers are installed in the Windows Service Control Manager. However, they are not started automatically because the configuration and license probably must be adjusted manually.

- **Environment**

  The **bin** subdirectory of the Eloquence B.07.00 installation (for example: **C:\Program Files\Eloquence\bin**) is added to the PATH environment variable so that each of the Eloquence command line programs can be conveniently executed. On Windows Server 2003, XP, 2000 and NT this is applied to the system PATH environment variable.

  On uninstallation, this setting is removed.

- **Configuration files**

  The SYSTEM and EXAMPLE volume paths are configured into the **eloq.config** file located in the **etc** subdirectory. Therefore, the Eloquence B.07.00 system programs (such as QUERY, CFORM, MFORM, PFORM) and examples can be executed without further configuration.

  The system and example programs are located in the **share\prog** and **share\example** subdirectories, respectively.

  Of course, if the existing **eloq.config** file contains customized settings which are different from the default settings they are not changed to keep your **eloq.config** file functional.

- **Windows system files**

  The Eloquence TCP network services are configured into the Windows SERVICES file. No further configuration is necessary to run any Eloquence client or server programs. However, these settings are not removed on uninstallation.

- **Windows start menu**

  The Eloquence program group contains additional shortcuts for Eloquence QUERY and the on-line documentation and manuals.

- **Autostart shortcut**

  If the eq70_01.cab archive is installed an autostart shortcut is created for the Eloquence Run server (applies to "All Users" on Windows Server 2003, XP, 2000 and NT). It is removed on uninstallation.

- **License requirements**

  Eloquence B.07.00 requires a new license key. Previous A.06.xx and B.06.xx license keys do not work anymore. The installation program installs a temporary license on demand even if an A.06.xx or B.06.xx temporary license has been installed previously.

Patch installation

Eloquence patches are provided to either fix a defect or limitation that was found after release and/or to add new functionality.

For proper operation of the Eloquence software, it is important that the recent patches are installed. For this purpose, a patch bundle is provided which can be used to conveniently install all recommended patches at once.

The **\B0700\patch\bundle** directory on the CD-ROM media contains the current Eloquence patch bundle at the time of compilation.

Please note that Eloquence patch bundles do not contain special components such as the IDE. These are covered by the individual Eloquence patches contained in the **\B0700\patch** directory on the CD-ROM media.

To find out about the most recent patches, please visit the Eloquence web site at: **http://www.marxmeier.com/eloquence/support/B07/patch.html**

NOTE: Installation of the "recommended" set of Eloquence patches is strongly recommended and may be required for correct product function.

On Windows, Eloquence patches and patch bundles are self-extracting executables which install automatically when executed.

For later reference, each patch installs a README file into the **share\doc** installation subdirectory (for example **C:\Program Files\Eloquence\share\doc**). The patch installation executables allow to view the README file before the patch is installed.

Configuring Eloquence on Windows

The Windows operating system might require additional configuration after the initial installation of the Eloquence software:

- Configure local and remote host names
- Modify the standard configuration of Eloquence service names

If you installed the Eloquence server software on Windows Server 2003, XP, 2000 or NT additional configuration is required:

- Configure the eloqsd service (Eloquence application server)
- Configure the eloqdb6 service (Eloquence database server)

Configure local and remote host names

Eloquence uses either DNS or a local HOSTS file to resolve IP adresses to host names and vice-versa. Please make sure that the IP address of the local system as well as the IP addresses of all remote systems you intend to use with Eloquence are correctly resolved to their host names (can be checked with the nslookup utility which is included with recent Windows versions). Add any missing IP address along with its host name to your local HOSTS file.

The location of your HOSTS file depends on your Windows version:

Windows 2003/XP C:\Windows\System32\Drivers\etc\HOSTS

Windows 2000/NT C:\WINNT\System32\Drivers\etc\HOSTS

Windows 95 C:\Windows\HOSTS

For example, to configure a host named "server" with IP address 192.168.1.1, add a line like this to your HOSTS file:

```
192.168.1.1  server
```

Modify the standard configuration of Eloquence service names

The installation program appends the following Eloquence specific TCP service definitions to your local SERVICES file:

```
#
# Eloquence related services
#
runsrv          8010/tcp    # Eloquence RUNSRV (Windows integration)
eloqdlg          8011/tcp    # Eloquence dialog
eloqsd           8100/tcp    # Eloquence A.06.xx eloqsd server
eloqdb           8102/tcp    # Eloquence A.06.xx data base server
eloqdb5          8104/tcp    # Eloquence A.06.xx ELOQDB5 server
```

where the first column specifies the service name (eg. runsrv) and the second column the associated port number and protocol (eg. 8010/tcp). The installation program takes care that the selected port numbers are not already used otherwise.

In rare cases it might be necessary to modify this standard configuration, for example if the port numbers do not match those configured on other systems.

The location of your SERVICES file depends on your Windows version:

Windows 2003/XP C:\Windows\System32\Drivers\etc\SERVICES

Windows 2000/NT C:\WINNT\System32\Drivers\etc\SERVICES

Windows 9x/ME: C:\Windows\SERVICES

NOTE: All systems must use the same port numbers in order to communicate.

Configure the eloqsd service

If you intend to run the Eloquence application server on your system additional configuration is required. This is described in detail in the next section.

Overview:

- Grant the **Log on as a batch job** right for all users who are using the eloqsd service
- Edit the eloqsd.cfg, eloqsd.user and eloqsd.share configuration files
- Select a startup mode for the eloqsd service
- Start the eloqsd service

NOTE: If the only Eloquence component you intend to use is the database, you don't need to configure and run the eloqsd service.

NOTE: The Windows Group Policy Editor is required to grant the **Log on as a batch job** user right. Since this is not available on Windows XP Home Edition the eloqsd service cannot be used on XP Home.

Configure the eloqdb6 service

If you intend to run the Eloquence database server on your system additional configuration is required. This is described in detail in a following section.

Overview:

- Edit the eloqdb6.cfg configuration file
- Create a data base environment (not covered in this document)
- Select a startup mode for the eloqdb6 service
- Start the eloqdb6 service

Configuring the eloqsd service

The eloqsd service is an important part of Eloquence. It is responsible for the following tasks

- Eloqsd provides file sharing capabilities for the graphical Eloquence development environment (the Eloquence IDE).
- Eloqsd is used to start eloqcore processes in the background.
- Eloqsd is used to count active users and validates them against available user licenses.

NOTE: The eloqsd service is not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you don't need to configure and run the eloqsd service.

NOTE: On Windows XP Home Edition, it is not possible to use the eloqsd service because XP Home lacks the Windows Group Policy Editor.

Eloquence implements its own file sharing capabilities through the eloqsd server. This makes it independent of the availability of specific network file systems (NFS, SMB) and overcomes inappropriate limitations.

- File names are case sensitive
- No limitations on files names, except they may not include a slash (used as directory separator) and a NUL character.
- There is a consistent authorization in a heterogeneous environment
- National characters are translated according to local conventions. So a client using the ISO 8859-1 character set encoding and a local user using a different character set encoding would see a consistent file name.
- Can be used on a WAN connection.

In order to run the eloqsd service it is required to adapt your system configuration. This involves the following steps:

- Registering the eloqsd server with the Windows operating system
- Configuring your system
- Configuring the eloqsd server
- Configuring the eloqsd server startup

Registering the eloqsd server with the Windows operating system

The eloqsd is installed as a Windows service. The installation program installs the **eloqsd.exe** executable in the Windows system directory (C:\Windows\System32). In addition, it is automatically registered with the Windows Service Control Manager.

NOTE: Manual registration of the eloqsd service with the Windows Service Control Manager is usually not required since the installation program performs a default registration. Unless you are interested in the registration and unregistration details you can skip this section.

In order to manually register the eloqsd server with the Windows Service Control Manager you must execute **eloqsd.exe** with the **-install** argument:

1. Log on as Administrator.
2. Open a Command Prompt window.
3. Execute the command: **eloqsd -install**. This makes the eloqsd server available to the Windows Service Control Manager. The service name is "eloqsd".

NOTE: To manually unregister, use the command: **eloqsd -remove**. Note, however, that this is normally not required since the installation program takes care of unregistering the eloqsd service during product update or uninstallation.

Configuring your System

This involves the following steps:

- Configuring the eloqsd TCP service
- Configuring the default eloqsd account
- Grant the background login right to users
- Increase the Windows desktop heap (optional)

Configuring the eloqsd TCP service

For eloqsd, the installation program adds the following default TCP service to your local SERVICES file:

```
eloqsd          8100/tcp    # Eloquence A.06.xx eloqsd server
```

In rare cases it might be necessary to modify this standard configuration, for example if the port number does not match the one configured on other systems. Please refer to the previous section *Modify the standard configuration of Eloquence service names* for details.

Configuring the default eloqsd account

The eloqsd server requires you to specify an account name in the configuration file. This account is used to define the access rights which are used unless a specific account is defined for the user.

You may specify any existing account in the configuration file (for example the guest account) or you may create a new user account "eloqsd" which is used by the eloqsd server.

Grant the background login right to users

Each system account associated with an eloqsd user must have the right to **log on as a batch job**. This must be setup manually using the Windows Server 2003/XP/2000 security settings or the Windows NT User Manager (or User Manager for Domains).

NOTE: This requires that the Windows Group Policy Editor is available which is not the case on Windows XP Home Edition.

1. Log on as Administrator.
2. **Windows Server 2003 or XP**:
   - Open the **Windows Control Panel**
   - Select **Performance and Maintenance**
   - Select **Administrative Tools**
   - Select **Security Settings**
   - Select **Local Policies** - **User Rights Assignment**

   Windows 2000:
   - Open the **Windows Control Panel**
   - Select **Administrative Tools**
   - Select **Security Settings**
   - Select **Local Policies** - **User Rights Assignment**

   Windows NT:
   - Select **Programs** - **Administrative Tools** in the **Windows Start Menu**
   - Start the **User Manager** (or the **User Manager for Domains**)
   - Select **User Rights** from the **Policies** menu
   - Check the **Show Advanced User Rights** option
3. Select the user right named "**Log on as a batch job**".
4. On Windows Server 2003/XP/2000, open the **Settings** dialog either from the menu or using the context menu (right mouse button). This opens the list of users/groups to which this user right is granted. On Windows NT, this list is part of the User Rights dialog and therefore already open.
5. Use the **Add...** button to add the users or groups who should have access to the eloqsd server. To grant this right to all accounts you could simply select "everyone".

Increase the Windows desktop heap

This is an optional configuration step which is only required if during runtime you encounter the Windows error #1816 (ERROR_NOT_ENOUGH_QUOTA). Getting this error indicates that there are too many concurrent eloqsd sessions so that the so-called Windows desktop heap overflows.

To change the size of the Windows desktop heap (more precisely: the desktop heap for the non-interactive windowstation), a Windows system registry entry must be edited, according to the procedure described in the Microsoft knowledge-base document **http://support.microsoft.com/?id=184802** .

NOTE:

Editing the Windows registry must be done with extreme care, as doing the wrong changes to a registry value can prevent Windows from operating correctly.

1. Log-on as Administrator (Windows "power user" privileges are sufficient).
2. Locate the following value in the Windows registry:
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\SubSystems\Windows
3. This is a string value consisting of multiple sections. Locate the section starting with SharedSection=, it should have 3 parameters which are comma-separated. Example **SharedSection=1024,3072,512**
4. The 3rd parameter (512 in the example above) is the one which must be increased. If you find only 2 parameters you should add the 3rd parameter yourself.

   You can set the 3rd parameter as high as the 2nd (3072 in the example above), but this is often too much. Increase it in steps of 256 until the Windows error #1816 does not occur anymore.
5. Windows must be restarted afterwards to activate this change.

Configuring the eloqsd server

The eloqsd server is configured by editing the **eloqsd.cfg**, **eloqsd.user** and **eloqsd.share** configuration files. They are located in the **etc** subdirectory of your Eloquence installation, for example at **C:\Program Files\Eloquence\etc**. All configuration files provide complete inline documentation and are included at the end of this chapter for your reference.

Each configuration file is responsible for a specific part of the eloqsd configuration:

eloqsd.cfg Defines the general configuration of the eloqsd server.

eloqsd.user Eloqsd provides its own user configuration. This makes it possible to define eloqsd users without the need to have a system account for each individual user. Instead, eloqsd users are associated with system accounts.

eloqsd.share This configuration file is used to define resources which can be accessed through the eloqsd server’s file sharing mechanism.

Configuring the eloqsd server startup

After the eloqsd server has been registered with the Windows Service Control Manager it can be accessed using the Services applet:

Windows Server 2003 or XP:

- Open the **Windows Control Panel**
- Select **Performance and Maintenance**
- Select **Administrative Tools**
- Open the **Services** applet

Windows 2000:

- Open the **Windows Control Panel**
- Select **Administrative Tools**
- Open the **Services** applet

Windows NT:

- Open the **Windows Control Panel**
- Open the **Services** applet

Locate the **eloqsd** entry in the list and select it with the mouse. On Windows Server 2003/XP/2000, open the Properties dialog either from the menu or using the context menu (right mouse button). On Windows NT, click the **Startup...** button to open the Startup Options dialog:

- If you want the eloqsd server to be automatically started each time Windows is restarted, you should select the **Automatic** Startup Type. This is the recommended setting.
- If you select the **Manual** Startup Type, you have to manually start the eloqsd server each time Windows is restarted.
- By default, the **Log On As** option is set to "System Account". This is the recommended setting since eloqsd has its own authorization facility which switches the system account whenever a new connection is established. This is controlled by means of the eloqsd.cfg and eloqsd.user configuration files.

  Alternatively, you can specify a different account using the **This Account** option, but concerning to the eloqsd server this will not make a big difference.

After you have setup the startup options, you can define additional arguments for the eloqsd server as an option. This is usually not required since all settings are normally provided in the eloqsd.cfg configuration file. However, this can be used to track down problems (e.g. starting the eloqsd server temporarily with different log options).

Possible arguments are:

```
-c name    = configuration file
```

```
-d flags   = log flags
-l name    = log file name
-s name    = service name (tcp/ip transport)
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-s name	The service name (as defined in SERVICES) or the port number where the server should listen for requests. The default value is eloqsd.	Service

*Equivalent configuration file directive.

For example, if you wish to specify a different TCP service name (the default is "eloqsd"), you could enter the following into the Startup Parameters field:

```
-save -s 8567
```

which causes the eloqsd server to listen to the TCP port number 8567. The leading **-save** argument makes this command line persistent so that the same arguments are used each time the eloqsd server is started. If you do not specify the **-save** argument this command line will be used only once.

If you want to make sure that the command line is empty and that any persistent command line is deleted, just specify **-save** without any additional arguments.

NOTE:

After having entered a command line, use the **Start** button from within the same dialog, otherwise your command line will not come into effect. In particular, if you use the **OK** button, the dialog will close without having started the service, i.e. your command line never being applied.

At this point, everything is configured and the eloqsd server can be started:

- **Windows Server 2003/XP/2000**:

  Select the **eloqsd** entry in the list, then choose **Start** either from the menu or using the context menu (right mouse button).

- **Windows NT**:

  Select the **eloqsd** entry in the list, then click the **Start** button.

If you did not specify a log file, the eloqsd server will write log messages to the Windows Event Log. In case the eloqsd does not start the Event Log will probably contain an error message pointing out the cause of the problem. You should also periodically check the Event Log in order to get aware of possible configuration problems:

Windows Server 2003 or XP:

- Open the **Windows Control Panel**
- Select **Performance and Maintenance**
- Select **Administrative Tools**
- Open the **Event Viewer** applet
- Select the **Application Log**

Windows 2000:

- Open the **Windows Control Panel**
- Select **Administrative Tools**
- Open the **Event Viewer** applet
- Select the **Application Log**

Windows NT:

- Select **Programs** - **Administrative Tools** in the **Windows Start Menu**
- Start the **Event Viewer**
- From the **Log** menu, select **Application**

Having started the **eloqsd** service for the first time after installation, the Application Log should already contain a few entries noting that the **eloqsd** was successfully installed and started.

Controlling the eloqsd server from the command line

The **net start** and **net stop** system commands can be used to start and stop the **eloqsd** server from the command line. This is equivalent to starting and stopping the server using the Windows Services applet as explained in the previous section.

Command syntax:

```
net start eloqsd
net stop eloqsd
```

The eloqsd HTTP status display

When **ServiceHttp** is defined in the **eloqsd.cfg** configuration file, you can use a web browser such as Mozilla, Netscape or Internet Explorer to view the configuration and state of the eloqsd process in your network.

To access the eloqsd server, you provide an URL like this:

```
http://server:port/
```

where server is the host name or IP address of the system running the eloqsd server and port is the port number used for **ServiceHttp** in the **eloqsd.cfg** file.

Default eloqsd.cfg file

```
# eloqsd.cfg
#
# @(#)$Revision: 1.7 2002/12/13 00:00 $
# The purpose of this file is to define the eloqsd properties.
# The location depends on the operating system:
#
# Windows: C:/Program Files/Eloquence/etc/eloqsd.cfg
#
# This file is read once at eloqsd startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.

### Server configuration

[Config]

# Service       The service name (as defined in the services file)
#               or the port number where the server should listen
#               for requests. The default value is eloqsd.
#
# ServiceHttp   The service name (as defined in the services file)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#
# UseKeepAlive  Numeric flag if the KEEP ALIVE socket option
#               should be used. Valid values are 1/0.
#               The default value is 1.
#               If this option is active, the server will check
#               after a system defined period of inactivity, if the
#               client is still alive.

#Service = eloqsd
#ServiceHttp = eloqsdhttp
#UseKeepAlive = 1

# panic         This option defines what should happen if a fatal
#               error is encountered.
#
#               The following options are valid:
#               exit    Terminate the process. This is the default.
#              dump    Terminate the process and create a core dump.
#
#            This is a problem tracking option. Unless you know what
#           you need the coredump for you probably want to stay with
#               the default.

#panic = exit

# HttpFrame     Numeric flag if the links should be omitted in HTTP
#               status. The default value is 0.
```

```
# HttpFrame = 0

# Lang          This configuration option defines the locale, the
#               server should use. The default value is "C".
#               The only locale currently supported is "C".
#
# Messages      This configuration option defines the language
#               for server messages. This value defaults to Lang.
#               The only locale currently supported is "C".
#
# Charset       This defines the character set encoding, the server
#               should use internally.
#
#               Valid setings are:
#                  HPROMAN8 - HP Roman8
#                  ISO8859/1 - ISO 8859/1
#               The default value for HP-UX is HPROMAN8, all other
#               platforms default to ISO8859/1.
#
#               This setting is used by the server to translate
#               client strings like user or file names.

#Lang = C
#Messages = C
#Charset = ISO8859/1

# AuthPolicy    This entry specifies, how user names and passwords
#               are validated. The following entries are valid:
#
#               server  - The server will validate passwords
#                         and user names using eloqsd.user
#
#               The default value is "server".
#
# userFile      The path/name of the eloqsd.user file.
#               The default value depends on your operating system:
#            Windows NT: C:/Program Files/Eloquence/etc/eloqsd.user
#
# shareFile     The path/name of the eloqsd.share file.
#               The default value depends on your operating system:
#           Windows NT: C:/Program Files/Eloquence/etc/eloqsd.share

#AuthPolicy = server

#userFile = C:/Program Files/Eloquence/etc/eloqsd.user
#shareFile = C:/Program Files/Eloquence/etc/eloqsd.share

# DefaultUID    The default name of the system account to run client
#           processes as, unless a different setting is provided for
#            the user. If this account is located on a domain server
#             it must be prefixed with the domain name followed by a
#               backslash character (domain\user).

DefaultUID = Guest

# LogFile       This defines where log messages are written to.
#              This configuration value either specifies a path/file
#               or one of the keywords below:
#
#             syslog   - log messages will be sent to the Windows NT
```

```
#                          Event Log
#
#               The default value is "syslog".

LogFile = syslog

# LogFlags      Each log message has an associated origin and
#               severity. The log flags define, which messages will
#               be logged. The "*" origin matches all message
#               origins, so it can be used to setup a default which
#               can be overriden for a specific message origin
#               (eg. "*1N0"). Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  C = Configuration subsystem
#                  N = Network transport
#                  P = Protocol handling
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#            Enabling log messages with L_DEBUG or L_VDEBUG severity
#               may result in a huge number of log messages.
#            To enable only fatal messages, you would want to set the
#              LogFlags to "*0", to enable regular log messages you
#               would want to set the LogFlags to "*1"

LogFlags = *1
```

Default eloqsd.share file

```
# eloqsd.share
#
# @(#)$Revision: 1.7 2002/12/13 00:00 $
# The purpose of this file is to define all disk ressources which are
# known to Eloquence. The location depends on the operating system:
#
# Windows: C:/Program Files/Eloquence/etc/eloqsd.share
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# Eloquence B.06.32 provides its own file sharing capabilities.
# This will make you independend of the availability of specific
# network file systems (NFS/SMB) and overcomes possible file system
# limitations.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each share definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [share_id]
# Path      Absolute path
# Comment   Share description. This is displayed by the client.
# Options   Comma separated list of operations to allow for this
#           share (currently unused)
#            READ, WRITE, CREATE
# Users    Comma separated list of individual users or user profiles
#           (currently unused)

[example]
Path = C:/Temp
Comment = Temporary Location
```

Default eloqsd.user file

```
# eloqsd.user
#
# @(#)$Revision: 1.7 2002/12/13 00:00 $
# The purpose of this file is to define all users which are known to
# Eloquence. The location depends on the operating system:
#
# Windows: C:/Program Files/Eloquence/etc/eloqsd.user
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# This makes it possible to define Eloquence users without the need
# to have a system account for each individual user.
# On Windows NT, the password provided to connect the EloqSD service
# must match the password defined for the Windows NT account
# associated with the given user id.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each user definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [user_id]
# Name       The full user name (currently unused)
# Email      Email address of the user (currently unused)
# uid        System account to execute client processes.
#          If this account is located on a domain server, it must be
#            prefixed with the domain name followed by a backslash
#            character (domain\user).
# Allow      Comma separated list of hosts/IP addresses to allow
#            connection for this user. (currently unused)
# Deny       Comma separated list of hosts/IP addresses to deny
#            connection for this user. (currently unused)
# Options    Comma separated list of capabilities to enable
#            for this user (currently unused)
#             HTTP_Info, HTTP_Admin, FileSharing, Debug, Attach
#             RExec, EqExec, DLGSRV, ...
# Profile    Template user entry. User defaults will be taken from
#            this section.
# Home      Home path. Defaults to the home directory associated to
#            the UID by the system.
#
# There are two predefined sections.
# [public] is used, if a client does does not provide a user id. This
# can only happen if an eloqcore has been started locally and
# requests a remote operation. It is probably a good idea to define
# a very restrictive deny for this section.
# [default] is used as the default user profile.

[public]
```

```
Name = Anonymous
deny = *

[default]
Name = Default user profile
deny = *

[demo]
Name = Joe Average
UID = JoesAccount
```

Configuring the eloqdb6 service

The eloqdb6 is the Eloquence database server. Eloquence uses a client/server database approach.

In order to run the eloqdb6 service it is required to adapt your system configuration. This involves the following steps:

- Registering the eloqdb6 server with the Windows operating system
- Configuring the eloqdb6 TCP service
- Configuring the eloqdb6 server
- Creating the database environment
- Configuring the eloqdb6 server startup

Registering the eloqdb6 server with the Windows operating system

The eloqdb6 is installed as a Windows service. The installation program installs the **eloqdb6.exe** executable in the Windows system directory (C:\Windows\System32). In addition, it is automatically registered with the Windows Service Control Manager.

NOTE: Manual registration of the eloqdb6 service with the Windows Service Control Manager is usually not required since the installation program performs a default registration. Unless you are interested in the registration and unregistration details you can skip this section.

In order to manually register the eloqdb6 server with the Windows Service Control Manager you must execute **eloqdb6.exe** with the **-install** argument:

1. Log on as Administrator.
2. Open a Command Prompt window.
3. Execute the command: **eloqdb6 -install**. This makes the eloqdb6 server available to the Windows Service Control Manager. The service name is "eloqdb6".

NOTE: To manually unregister, use the command: **eloqdb6 -remove**. Note, however, that this is normally not required since the installation program takes care of unregistering the eloqdb6 service during product update or uninstallation.

Configuring the eloqdb6 TCP service

For eloqdb6, the installation program adds the following default TCP service to your local SERVICES file:

```
eloqdb          8102/tcp    # Eloquence A.06.xx data base server
```

In rare cases it might be necessary to modify this standard configuration, for example if the port number does not match the one configured on other systems. Please refer to the previous section *Modify the standard configuration of Eloquence service names* for details.

Configuring the eloqdb6 server

The eloqdb6 server is configured by editing the **eloqdb6.cfg** configuration file. It is located in the **etc** subdirectory of your Eloquence installation, for example at **C:\Program Files\Eloquence\etc**. This file provides complete inline documentation and is included at the end of this chapter for your reference.

The default configuration is not optimized for performance and does not handle a large number of concurrent connections. Therefore, you should adjust the following parameters in the **eloqdb6.cfg** configuration file to your requirements:

- Section **[Config]**, parameter **Threads**

  This parameter specifies how many connections to the database server can be established concurrently. The default is 40 which means that a maximum of 40 users can access the database at the same time. Multiple programs accessing the database count as multiple connections, while accessing multiple databases from within the same program counts as one single connection.

- Section **[Config]**, parameter **BufferCache**

  The recommended minimum value is 64 (megabytes). A higher value is recommended. The current limit is approx. 1 gigabyte. The default and minimum buffer cache size is 5 Megabytes which may lead to bad performance.

- Section **[Config]**, parameter **CheckPtSize**

  Whenever the transaction journal exceeds this size (in megabytes), eloqdb6 performs an internal checkpoint operation to recycle the journal. The default size is 10 megabytes which probably causes the checkpoint to happen too often, causing a performance impact. It is recommended to set this parameter to 20 or higher.

NOTE: If the configuration of a running eloqdb6 service is changed, the service must be restarted to activate the new configuration.

Creating the database environment

A database environment consists of at least a primary data volume and a transaction log volume. These must be created on a local disk before the eloqdb6 can be started for the first time.

Volume files should be created on NTFS file systems only. If you use a FAT file system, be aware that it limits the maximum file size to 4 gigabytes, while NTFS allows it to grow until its hard size limit of 128 gigabytes.

Additional data volumes can be created at any time to extend the available database storage. Additional transaction log volumes can be added as well, but it is unlikely that these will ever be used because the purpose of a log volume is to hold temporary data only.

The maximum number of volume files in a database environment is 255 which is equivalent to a maximum database storage size of 31.75 terabytes (254 data volumes and 1 log volume).

It is recommended to choose a dedicated directory to hold all volume files belonging to a particular database environment. The following example assumes that the initial volumes are created in the C:\data\db directory:

1. Open a Command Prompt window.
2. Change to the directory where the volumes shall be created:

   ```
   C:
   cd \data\db
   ```

3. Create the initial data volume:

   ```
   dbvolcreate data.vol
   ```

4. Create the initial transaction log volume:

   ```
   dbvolextend -t log log.vol
   ```

Both files are created with their initial minimum size which is 2.5 megabytes. They will grow on demand until they reach their maximum size. The minimum and maximum size and the amount by which the files shall grow can be configured either at creation time or afterwards with the **dbvolchange** utility.

To view the available command line options, use:

```
dbvolcreate -help
dbvolextend -help
dbvolchange -help
```

The **dbvolcreate** and **dbvolextend** utilities automatically maintain the list of volume files in the **[Volumes]** section of the **eloqdb6.cfg** configuration file.

Configuring the eloqdb6 server startup

After the eloqdb6 server has been registered with the Windows Service Control Manager it can be accessed using the Services applet.

NOTE: The following instructions refer to the default "eloqdb6" service name. If multiple instances of the eloqdb6 service are installed they are separately listed and also separately configured in the Windows Service Control Manager.

NOTE: Multiple eloqdb6 instances are covered in detail in the section *Setting up multiple eloqdb6 instances* below.

Windows Server 2003 or XP:

- Open the **Windows Control Panel**
- Select **Performance and Maintenance**
- Select **Administrative Tools**
- Open the **Services** applet

Windows 2000:

- Open the **Windows Control Panel**
- Select **Administrative Tools**
- Open the **Services** applet

Windows NT:

- Open the **Windows Control Panel**
- Open the **Services** applet

Locate the **eloqdb6** entry in the list and select it with the mouse. On Windows Server 2003/XP/2000, open the Properties dialog either from the menu or using the context menu (right mouse button). On Windows NT, click the **Startup...** button to open the Startup Options dialog:

- If you want the eloqdb6 server to be automatically started each time Windows is restarted, you should select the **Automatic** Startup Type. This is the recommended setting.
- If you select the **Manual** Startup Type, you have to manually start the eloqdb6 server each time Windows is restarted.
- By default, the **Log On As** option is set to "System Account". This is the recommended setting. Alternatively, you can specify a different account using the **This Account** option.

After you have setup the startup options, you can define additional arguments for the eloqdb6 server as an option. If a single instance of the eloqdb6 server is used this is usually not required since all settings are normally provided in the eloqdb6.cfg configuration file. However, this can be used to track down problems (e.g. starting the eloqdb6 server temporarily with different log options) and is mandantory if you have multiple eloqdb6 instances.

Possible arguments are:

```
-c name    = configuration file
-d flags   = log flags
-l name    = log file name
-s name    = service name (tcp/ip transport)
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-s name	The service name (as defined in SERVICES) or the port number where the server should listen for requests. The default value is eloqdb.	Service

*Equivalent configuration file directive.

For example, if you wish to specify a different configuration file (required for additional eloqdb6 instances), you could enter the following into the Startup Parameters field:

```
-save -c "C:\Program Files\Eloquence\etc\eloqdb6.instance.2.cfg"
```

which causes this eloqdb6 server instance to use the eloqdb6.instance.2.cfg configuration file in the C:\Program Files\Eloquence\etc directory. Note that the argument must be put into quotes because the configuration path contains a space.

The leading **`-save`** argument makes this command line persistent so that the same arguments are used each time this eloqdb6 server instance is started. If you do not specify the **`-save`** argument this command line will be used only once.

If you want to make sure that the command line is empty and that any persistent command line is deleted, just specify **`-save`** without any additional arguments.

NOTE: After having entered a command line, use the **Start** button from within the same dialog, otherwise your command line will not come into effect. In particular, if you use the **OK** button, the dialog will close without having started the service, i.e. your command line never being applied.

At this point, everything is configured and the eloqdb6 server can be started:

- **Windows Server 2003/XP/2000**:

  Select the **eloqdb6** entry in the list (or the additional instance you want to start), then choose **Start** either from the menu or using the context menu (right mouse button).

- **Windows NT**:

  Select the **eloqdb6** entry in the list (or the additional instance you want to start), then click the **Start** button.

If you did not specify a log file, the eloqdb6 server will write log messages to the Windows Event Log. In case the eloqdb6 does not start, the Event Log will probably contain an error message pointing out the cause of the problem. You should also periodically check the Event Log in order to get aware of possible configuration problems:

Windows Server 2003 or XP:

- Open the **Windows Control Panel**
- Select **Performance and Maintenance**
- Select **Administrative Tools**
- Open the **Event Viewer** applet
- Select the **Application Log**

Windows 2000:

- Open the **Windows Control Panel**
- Select **Administrative Tools**
- Open the **Event Viewer** applet
- Select the **Application Log**

Windows NT:

- Select **Programs** - **Administrative Tools** in the **Windows Start Menu**
- Start the **Event Viewer**
- From the **Log** menu, select **Application**

Having started the **eloqdb6** service for the first time after installation, the Application Log should already contain a few entries noting that the **eloqdb6** was successfully installed and started.

Controlling the eloqdb6 server from the command line

The **net start** and **net stop** system commands can be used to start and stop the **eloqdb6** server from the command line. This is equivalent to starting and stopping the server using the Windows Services applet as explained in the previous section.

Command syntax:

```
net start SERVICE
net stop SERVICE
```

where *SERVICE* is either **eloqdb6** or the name of an additional instance, for example:

```
net start eloqdb6
net stop eloqdb6
```

The eloqdb6 HTTP status display

When **ServiceHttp** is defined in the **eloqdb6.cfg** configuration file, you can use a web browser such as Mozilla, Netscape or Internet Explorer to view the configuration and state of the eloqdb6 process in your network.

To access the eloqdb6 server, you provide an URL like this:

```
http://server:port/
```

where server is the host name or IP address of the system running the eloqdb6 server and port is the port number used for **ServiceHttp** in the **eloqdb6.cfg** file.

NOTE: Each eloqdb6 instance on the same system requires an unique HTTP port number.

Setting up multiple eloqdb6 instances

Multiple instances of the eloqdb6 server can coexist on the same system. This makes sense if multiple database environments should be kept separate from each other, either to keep the databases in separate namespaces or simply to maintain discrete database storage.

The following steps are required to setup an additional eloqdb6 server instance:

1 Register an additional service name with the Windows Service Control Manager.

Each instance requires a separate service name, i.e. a separate entry in the Windows Service Control Manager. You can use any name you like, although we recommend that it starts with "eloqdb6".

The additional service name is specified after the **-install** or **-remove** arguments, respectively (see the previous section *Registering the eloqdb6 server with the Windows operating system*).

For example, to register an additional instance named "eloqdb6.instance.2", open a Command Prompt window and execute the command:

```
eloqdb6 -install eloqdb6.instance.2
```

If the service name contains spaces it must be enclosed in quotes. However, to retain compatibility with legacy Windows NT systems, any space is converted into an underscore character, so we do not recommend to use spaces in a service name.

NOTE:

During product update, the installation program does not retain additional eloqdb6 instances. After an update has been applied, only the standard "eloqdb6" service name is present. Any additional instances must be re-created manually.

2 Each eloqdb6 instance needs a separate configuration file. You can copy either the configuration of any existing instance or the **eloqdb6.cfg.sam** template file located in the **etc** subdirectory of your Eloquence installation (for example: **C:\Program Files\Eloquence\etc\eloqdb6.cfg.sam**).

In this example, we create a new configuration file named eloqdb6.instance.2.cfg in the default C:\Program Files\Eloquence\etc configuration directory. This is not mandatory, the configuration file may have any name and can be located anywhere.

```
C:
cd "\Program Files\Eloquence\etc"
copy eloqdb6.cfg.sam eloqdb6.instance.2.cfg
```

Next, the new configuration file must be edited. Besides the parameters mentioned in the previous section *Configuring the eloqdb6 server*, the following parameters must be adjusted:

- Section **[Server]**, parameter **Service**

  Each instance requires an unique TCP service name or port number. You can either configure a dedicated service name in your local SERVICES file (please refer to the previous section *Configuring the eloqdb6 TCP service* for details) or simply enter an unique port number.

- Section **[Server]**, parameter **ServiceHttp**

  If you use the eloqdb6 HTTP status display, you are required to configure an unique TCP service name or port number for each eloqdb6 instance.

- Section **[Server]**, parameter **LogFile** (optional)

  By default, all eloqdb6 log messages are written to the Windows Application Event Log. Each log message is labeled with the name of the originating eloqdb6 instance. However, it might be more convenient to configure a separate log file for each instance. This log file could be located within the dedicated, instance-specific directory (see below), for example:

  ```
  LogFile = C:\data\db\instance.2\eloqdb6.log
  ```

When started, this eloqdb6 instance will create this file and write all log messages to it. This is a plain text file which can be viewed with any text editor.

NOTE: If you copied the configuration of an existing instance, it is required that you manually delete all volume references at the end of the file below the **[Volumes]** section header.

3 Create the instance-specific database environment. You do this according to the previous section *Creating the database environment*, but you use the **`-c`** command line option to refer to the instance-specific configuration file:

- Open a Command Prompt window.
- Create the instance-specific directory. We recommend to create a dedicated directory for each instance where the instance-specific files are located:

```
C:
cd \data\db
mkdir instance.2
cd instance.2
```

- Create the initial data volume (the following command must be entered as a single line):

```
dbvolcreate
 -c "C:\Program Files\Eloquence\etc\eloqdb6.instance.2.cfg"
 data.vol
```

- Create the initial transaction log volume (the following command must be entered as a single line):

```
dbvolextend
 -c "C:\Program Files\Eloquence\etc\eloqdb6.instance.2.cfg"
 -t log log.vol
```

In this example, the argument to the **`-c`** command line option refers to the instance-specific configuration file located in the current directory. Note that it must be put into quotes because the configuration path contains a space.

4 Start the new eloqdb6 instance for the first time, setting-up a persistent command line using the **`-c`** command line option to refer to the instance-specific configuration file (for details, please refer to the previous section *Configuring the eloqdb6 server startup*):

Windows Server 2003 or XP:

- Open the **Windows Control Panel**
- Select **Performance and Maintenance**
- Select **Administrative Tools**
- Open the **Services** applet

Windows 2000:

- Open the **Windows Control Panel**
- Select **Administrative Tools**
- Open the **Services** applet

Windows NT:

- Open the **Windows Control Panel**
- Open the **Services** applet

Locate the new eloqdb6 instance in the list and select it with the mouse. On Windows Server 2003/XP/2000, open the Properties dialog either from the menu or using the context menu (right mouse button). On Windows NT, click the **Startup...** button to open the Startup Options dialog.

First, choose the Startup Type you want to configure for this instance (**Manual** or **Automatic**).

Next, in the Startup Parameter input field at the bottom of the dialog, refer to the instance-specific configuration file, using the *absolute path* (which must be put into quotes because it contains a space):

```
-save -c "C:\Program Files\Eloquence\etc\eloqdb6.instance.2.cfg"
```

Finally, click the **Start** button. Do *not* use the **OK** button to leave the dialog. The command line is only applied if you immediately start the service. If you click **OK**, the dialog will close without having started the service, i.e. your command line never being applied.

The **`-save`** option specifies that this command line is to be saved into the Windows registry. Therefore, it is not necessary to enter this command line ever again. It remains associated with this eloqdb6 instance until a different command line is registered or the instance is removed.

The new instance should now be up and running. You can start and stop it using the Windows Services applet. You can also control it from the command line using the **net start** and **net stop** system commands:

```
net start eloqdb6.instance.2
net stop eloqdb6.instance.2
```

To access a database in an additional instance, specify the instance-specific service name or port number in addition to the database name. Those utilities which do not expect a database name instead provide the **`-h`** command line option. Also, the EQ_DBSERVER environment variable can be set to specify the default instance.

The following examples assume that an additional instance was configured to use the port number 8201. They illustrate different ways to address the same instance:

```
schema -h :8201 db.schema.txt
dbcreate :8201/db
```

```
set EQ_DBSERVER=:8201
schema db.schema.txt
dbcreate db
```

Default eloqdb6.cfg file

```
# eloqdb6.cfg
#
# @(#)$Revision: B.07.00.2 $
# This file defines the eloqdb6 configuration and the database
# environment.
# The default location depends on the operating system:
#
# Windows: C:/Program Files/Eloquence/etc/eloqdb6.cfg
#
# This file is read once at eloqdb6 startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.


### Server configuration

[Server]

# Service       The service name (as defined in the services file)
#               or the port number where the server should listen
#               for requests. The default value is eloqdb.
#
# ServiceHttp   The service name (as defined in the services file)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#

#Service = eloqdb
#ServiceHttp = eloqdbhttp

# SyncMode      If set, this causes the eloqdb6 server to operate in
#             sync write mode. The sync write mode is more resistent
#              against operating system and hardware failures. When
#             sync mode is disabled (set to 0) the eloqdb6 uses the
#             faster async write strategy which performs fewer disk
#            writes but could lead to a damaged database environment
#               in case of a system failure.
#               The default value is 1 (sync write mode enabled).

#SyncMode = 1

# LogFile       This defines where log messages are written to.
#              This configuration value either specifies a path/file
#               or one of the keywords below:
#
#               console  - log messages are written to the console
#              syslog   - log messages will be sent to the Windows NT
#                                 Event Log
#
#               The default value is "syslog".
```

```
#LogFile = syslog

# LogFlags      Each log message has an associated origin and
#               severity. The log flags define, which messages will
#               be logged. The "*" origin matches all message
#               origins, so it can be used to setup a default which
#               can be overriden for a specific message origin
#               (eg. "*1N0"). Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  A = Configuration subsystem
#                  X = Network transport
#                  P = Protocol handling
#                  T = Thread kernel
#                  I = IMAGE subsystem
#                  B = BTREE subsystem
#                  F = FIXREC subsystem
#                  V = Volume handling
#                  L = Transaction logging
#                  C = Page cache
#                  N = Node handling
#                  D = The server framework
#                  O = System catalog
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#            Enabling log messages with L_DEBUG or L_VDEBUG severity
#               may result in a huge number of log messages.
#              To suppress anything but fatal messages, you can set
#             LogFlags to "*0". To enable informational log messages
#               you can set the LogFlags to "*1".

#LogFlags = *0

# HTTPUser      The eloqdb6 server is able to display status
#            information by supporting the HTTP protocol (you can use
#               Mozilla or Internet Explorer to monitor the database
#               server process, see ServiceHttp above).
#             If set, the eloqdb6 HTTP status display will require a
#            matching user name (HTTP basic authentification) before
#               allowing access to the eloqdb6 HTTP status.
#               The default value is empty.
#
# HTTPPswd     If set, the eloqdb6 HTTP status display will require a
#             matching password (HTTP basic authentification) before
#               allowing access to the eloqdb6 HTTP status.
#               The default value is empty.
```

```
#HttpUser =
#HttpPswd =

# HTTPFrame     If set, no link information is output on the HTTP
#              status display. So the status page could be used in a
#               web frame. Default value is 0.

#HttpFrame = 0


### Data base configuration

[Config]

# Threads      Number of threads in the data base server. A separate
#               thread is required for each client.
#               Default number of threads is 40.

#Threads = 40

# LockConflictingItems              If set, predicate locks with
#            conflicting items are granted, however any write attempt
#            to data where another process owns a lock will result in
#            a status error -12. Former Eloquence revisions rejected
#             a predicate lock with a conflicting item, because this
#            could lead to a situation where two processes own a lock
#               on an overlapping subset of data.
#               The default value is 0.

#LockConflictingItems = 0

# AllowSecondaryBlockingLock       If set, secondary blocking locks
#             are allowed. In previous Eloquence versions, secondary
#                locks in a blocking mode (odd modes) failed with
#               database status -135 ("Second lock is not allowed in
#               modes 1,3,5,11,13 and 15.") instead of blocking.
#              Current Eloquence versions return the status code -35
#               in case a deadlock situation caused by a secondary
#              blocking lock is detected. Therefore, this setting is
#             enabled by default. To retain the behavior of previous
#               Eloquence versions it can be set to 0.
#               The default value is 1.

#AllowSecondaryBlockingLock = 1

# BufferCache   Size of page cache in megabytes. The page cache is
#             used to reduce the number of disc accesses. Large cache
#              size will speed up random database access, while a too
#                small cache size may cause bad server performance.
#                Default cache size is 5 MB.

#BufferCache = 5

# The server performs a checkpoint operation at fixed intervals. This
# flushes all modified buffers (including metadata) to the disk and
# resets log of committed transactions. A checkpoint is a point where
# the server knows all data are in a consistent state. Any data
# modification since the last checkpoint is recorded in the log
# volume.
#
# CheckPtFreq   Checkpoint frequency in seconds.
```

```
#               Default checkpoint frequency is 60 seconds.
#
# CheckPtSize   Checkpoint frequency based on accumulated log space
#              which would be freed by a checkpoint (in megabytes).
#               A zero CheckPtSize value disables size based
#             checkpoints. Default checkpoint size is 10 megabytes.
#
# The database server performs a checkpoint operation at a fixed
# interval and optionally in addition when the accumulated log space
# which could be freed by a checkpoint operation reaches a given
# threshold. The frequency of the checkpoint operations has a great
# influence on the size of the log volume since the log volume must
# hold all committed transactions since between checkpoints.

#CheckPtFreq = 60
#CheckPtSize = 10

# The syncer thread flushes modified buffer pages to the disk when
# they are likely to become reused in the near future.
#
# SyncerFreq    Syncer thread invocation frequency (in seconds)
#               Default interval is 5 seconds.

#SyncerFreq = 5

# SyncerJournalFlushInterval        If SyncMode is enabled this
#               configuration item specifies the interval (in
#               milliseconds) at which the journal of committed
#               transactions is synchronized to disk.
#               In case of an operating system or hardware failure
#              transactions that were not synchronized to disk are
#               typically lost.
#              A smaller value reduces the amount of transactions
#             that might be lost in case of a system crash. However,
#            setting this value too low significantly impacts write
#               performance.
#             Setting this value to 0 reverts to the legacy SyncMode
#               behavior where every transaction is immediately
#              synchronized. The default value is 500 milliseconds.

#SyncerJournalFlushInterval = 500


### Store/Restore Devices

[Devices]

# This section defines the "server devices" which can be used with
# dbstore and dbrestore. Each entry consists of the device name and
# an associated path.
#
# A "server device" could either be a single file or a directory.
# When no server devices are configured, dbstore and dbrestore
# operation is refused by the server.
#
# The example below defines two server devices. The device "file"
# points to a single file, the device "backup" points to a directory
# which is intended to hold the backup files.

#File = C:/Temp/Backup.dat
#Backup = C:/Backup
```

```
### Forward log

[ForwardLog]

# FwLog         Configures the file, device or pipe to be used for
#              forward-logging. Using the %N token in the file name
#             activates automatic file management (not possible for
#               devices or pipes).
#               By default, forward-logging is inactive.
#
#             The examples below configure an automatically managed
#             file and a pipe which compresses the data on-the-fly:

#FwLog = /mnt/disk2/data/db-forward-%N.log
#FwLog = |gzip -c >/mnt/disk2/data/db-forward.log.gz

# FwRecovery   Configures the file, device or pipe to be used during
#              forward-recovery. If not set, the Log setting is used
#               by default.
#
#             The example below configures a pipe which uncompresses
#               the data on-the-fly:

#FwRecovery = |gzip -dc /mnt/disk2/data/db-forward.log.gz

# FwOnFailure   Configures the action to be taken in case the
#            forward-log cannot be written, e.g. due to insufficient
#               disk space. Possible values are disable or panic.
#             If set to disable, forward-logging will be disabled on
#               failure. As soon as the problem is solved it can be
#               manually enabled using dbctl. If set to panic, the
#               eloqdb6 server will issue a panic and abort itself.
#               The default value is disable.

#FwOnFailure = disable

# FwMaxSize     Limits the maximum size of automatically managed
#             forward-log files (in megabytes). If not set or set to
#               zero, the file size limit is 2 gigabytes.
#               The default value is 0 (not set).

#FwMaxSize = 0


### Data base environment

[Volumes]

# List of data base volumes. Initially empty.
# This is usually filled in by dbvolcreate and dbvolextend utilities
```

Customizing the Eloquence Configuration Files

This discussion assumes that the Eloquence software has already been installed on your system. The information in this section is directed to the system administrator for the Eloquence software.

NOTE: The configuration steps mentioned here are not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you can skip this section.

Before Eloquence can be used, its resources must be configured. Eloquence programs usually don't use system resources directly, instead they rely on a mapping of paths, printers and device files in Eloquence configuration files.

There are two different levels of configuration:

System global — This is achived with the **eloq.config** configuration file which is located in the Eloquence configuration directory.

User specific — This is achived with the **eloq.rc** configuration file which is located in the home directory of the user.

The Eloquence configuration files are read by the eloqcore process when it is started. The configuration files are processed in an order such that more specific definitions override the more general ones. So a system global assignment can be overridden from a group specific configuration file, a user specific definition will override group and system global definitions.

The system global configuration file **eloq.config** is located in the **etc** subdirectory of your Eloquence installation (for example **C:\Program Files\Eloquence\etc\eloq.config**). In addition, the **etc** subdirectory contains the **eloq.config.sam** template file which is updated during installation. The template configuration file provides complete inline documentation and is included at the end of this chapter for your reference.

Eloquence resource configuration

Eloquence resources go back to the "dark ages" when a precedessor of Eloquence was implemented in hardware (called HP250/HP260 at that time) and the resources definition actually were real OS resources. Since programs depended on a program independent resource configuration and it is a convenient mechanism anyway, the concept was kept. Instead of real devices Eloquence resources can be mapped to whatever is appropriate. Eloquence is of course able to access native operating system resources directly.

Since the following names are not commonly used, let's define them first:

VOLUME A **VOLUME** is the Eloquence concept of a directory. Instead of using the path directly, it is possible to assign an identifier for a path and refer to it in a symbolic manner.

MSI This is a short form of **MASS STORAGE IS** and species the default **VOLUME** on which paths should be related unless an absolute path or another **VOLUME** is given.

PRINTER A **PRINTER** is the Eloquence concept of an output device. A PRINTER is identified by a number and could be mapped to a device file or to a sequence of commands.

The device numbers 8 to 10 have special predefined meanings:

8: Display terminal.
9: Bit bucket (Eloquence equivalent of /dev/null)
10: Local terminal printer

The eloq.config configration file

The **eloq.config** file provides system global definitions and is usually copied during installation from the **eloq.config.sam** template file in case it is not already present.

The user specific configuration file

To provide user specific definitions, you could install a user specific configuration file in the home directory of the user. Consider we would like to have a specific configuration for the user *mike*, you would perform the following steps:

1. Change to the home directory of the user (the directory where the HOME environment variable refers to):

   ```
   C:
   cd \home\mike
   ```

2. Copy the default **eloq.config.sam** configuration file to the home directory of the user and rename it to **eloq.rc**.

3. Use a text editor, such as notepad to edit the file

   ```
   notepad eloq.rc
   ```

Template eloq.config file

```
# Sample global Eloquence configuration file
#
# (C) Copyright Marxmeier Software AG, 2002
# @(#)$Revision: 22.2 $
#
# This file defines the global Eloquence configuration
# It must be named eloq.config and located in the Eloquence
# configuration directory. The location of this directory
# depends on your operating system and the Eloquence
# installation base directory.
#
# On the Windows platform, it is located by default in the
# C:/Program Files/Eloquence/etc directory.
#
# PLEASE NOTE:
# You MUST define at least one volume (typically SYSTEM, see below),
# or eloqcore will fail on startup.
#
# Globally defined volumes
#
#       Format: VOLUME label [device] path
#
#       label   - Volume label (up to 8 characters)
#                 must be unique
#       device  - HP260 device specifier eg. :F2,6,0
#                 ** optional, ignored when present
#       path    - path to map volume
#
# Globally defined printers
#
#       Format: PRINTER no [model] type spec
#
#       no      - printer select code (-2 .. 7, 11 .. 99)
#       model   - PCL or OTHER
#                 ** optional, unused
#       type    - printer type PIPE, FILE or SYSTEM
#       spec    - path/command to process on printer selection
#
# Globally defined ports (not available on Windows)
#
#       Format: PORT no spec
#
#       no      - port select code (11 .. 20)
#                 may not conflict with PRINTER
#       spec    - path of tty devicefile
#
#
# Default date/time format
#
#       Format: DATE spec
#               TIME spec
#
#      spec    - date/time specification. please refer to date(1) or
#                strftime(3) for more information.
#              For backward compatibility, the former specifications
```

```
#                 "DD.MM.YY" and "MM/DD/YY" are silently converted.
#
#                 Default: DATE "%m/%d/%y"
#                          TIME "%H:%M:%S"
#
# Global MSI value
#
#       Format: MSI label
#
#      label   - Volume label. Default is the first defined volume.
#
#*****************************************************************

# sample date format

# DATE "%m/%d/%y"
# TIME "%H:%M:%S"

# sample volume configuration

VOLUME LOCAL    .
VOLUME SYSTEM   "C:/Program Files/Eloquence/share/prog"
VOLUME EXAMPLE  "C:/Program Files/Eloquence/share/example"

# sample global printers

# This is an example how to access a native Windows printer
# The printer name must be given exactly as defined in Windows.
PRINTER 0 SYSTEM "HP LaserJet 4/4M Plus PS 600"
```

Configuration of the GUI Dialog Server

The GUI dialog server allows Eloquence programs to apply a user interface with native Windows look-and-feel. It consists of two components:

- the Run server (RUNSRV32) which permanently runs in the background on the user's PC, waiting for incoming TCP network requests from a (remote) Eloquence program
- the Dialog server (DLGSRV32) which is started by the Run server to connect with the Eloquence program and handle TCP network dialog requests

The process about how the Eloquence program (i.e. the **`eloqcore`** process) connects to the Run and Dialog servers is configured in the **`eloq.ini`** configuration file. This file is located on the system where the program is run, for example on a remote system.

On the user's PC, the **`eloqcl.ini`** configuration file specifies the configuration of the RUNSRV32 and DLGSRV32 processes, e.g. which TCP port the Run server listens to for incoming connections and the command line the Run server uses to start the Dialog server.

Customization of the eloq.ini and eloqcl.ini files

Eloquence configuration files are located in the **etc** subdirectory of your Eloquence installation directory (for example **C:\Program Files\Eloquence\etc**).

The installation program does not overwrite existing configuration files but installs the **eloq.ini.sam** and **eloqcl.ini.sam** files instead.

The 'ini' File format

The **`eloq.ini`** and **`eloqcl.ini`** files contain several sections each containing a group of related configuration items. The sections and configuration items have the following format:

```
[Section]
Item=Value
```

Section is the name of a section. The enclosing brackets ([ ]) are required and they must start at the first column. *Item=Value* defines a value of a configuration item. *Item* is the name of a configuration item. It consists of any sequence of characters (case insensitive) and digits followed by an assignment operator (=). Depending on the item type, the value may either be numeric or alphanumeric (optionally enclosed in double quotes). Comment lines must start with a semicolon (;) or a hash character (#) in the first column.

The eloq.ini file

Section [runsrv]

This section specifies the defaults used with the **`runsrv`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: runsrv

Function: The default service name or port number to be used with the **`runsrv`** connection protocol. If no value is defined the port associated with the service runsrv is used.

Section [eloqdlg]

This section specifies the defaults used with the **`dlg`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: 8011

Function: The default service name or port number to be used with the **`dlg`** connection protocol. If no value is defined the port number 8011 is used.

Mapping Driver Specifications

Other sections in the **eloq.ini** file can be used to map a driver specification to a different value. When a driver is specified (e.g. in the **`DLG SET ".driver"`** statement or with the **`eloqcore -dlg`** command line option) the argument is looked-up in the **eloq.ini** configuration file after any protocol specification has been removed. If a section has been found it is used to replace the original argument.

The following configuration items are supported:

dlg

dlg= *replacement text*

Function: When contacting a DLG driver this is used as a replacement for the specified driver. The value can specify a different protocol.

host

host= *[hostname][:service]*

Function: When using the Run server to execute a remote operation or to start the Dialog server the host variable can be used to specify a different hostname (or IP address) and/or a different service name (or port number) which should be used instead.

Example

```
[runsrv]
Service = runsrv

[eloqdlg]
Service = 8011

[mike]
dlg = dlg://lxmike

[chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```

This example specifies to use a port number for the **runsrv** protocol which is associated with the service runsrv (as defined in the local SERVICES file). For the **dlg** protocol the port number 8011 should be used.

When a driver argument "mike" is specified (e.g. "**@mike**", "**dlg://mike**" or "**runsrv://mike**") the value "**dlg://lxmike**" is used instead. When a driver argument "chris" is specified it is replaced by "**runsrv://wserv:8765**". When the Run server is contacted from either the command line (using the **runclnt** utility) or the **RunSrv.DLL** and a remote name "chris" is passed, the value "**wserv:8765**" is used instead.

Running multiple runsrv instances on a single system

This is required if the Citrix Metaframe or the Windows Terminal Server product is used. Each user needs to start the RUNSRV32 with a different port number (probably during autostart). Then a driver name containing the host name and the user name is passed and translated by using the mapping defined in **eloq.ini**.

Suppose the user "chris" has a RUNSRV32 using port 8765 when working on system wserv, then the following mapping section should be added in **eloq.ini**:

```
[wserv_chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```

The eloqcl.ini file

Section [runsrv]

This section configures the RUNSRV32. The following configuration items are supported:

DlgSrv

DlgSrv= *command to start the DLGSRV32*

Default: `DLGSRV32 -connect %s -IDMfont 1 -IDMcolor 1`

Function: The command line used to start the DLGSRV32. This should contain exactly one occurrence of the `%s` token which will be replaced with the connection argument at runtime.

Service

Service= *service name or port number*

Default: runsrv

Function: RUNSRV32 listens for incoming TCP network requests on this service/port.

Debug

Debug= *debug level*

Default: 0

Function: This is used for internal diagnostics only and should be set to 0.

DDEWait

DDEWait= *milliseconds*

Default: 2500

Function: Number of milliseconds to wait after a DDE server has been started and before the DDE transaction is initiated.

If you encounter the status message "Module ... is not active and I don't know how to start it" although the module is correctly defined (see *Section [modules]* below) you should increase this value.

TrayDoubleClick

TrayDoubleClick= *0 or 1*

Default: 0

Function: Whether the RUNSRV32 tray icon should open the Run server window on single (0) or double (1) mouse click.

Section [dlgsrv]

This section configures the DLGSRV32. The following configuration items are supported in the *[dlgsrv]* section as well as in any *user-defined section*:

DefaultsFile

DefaultsFile= *absolute path to defaults file*

Default: The **`defaults.eq`** file in the **dlg** subdirectory of the Eloquence installation directory (for example: **C:\Program Files\Eloquence\dlg\defaults.eq**).

Function: Specifies the defaults file used by DLGSRV32. This file is used to provide defaults to dynamically created objects.

To use modular dialog files, change this to the **dlg\module** subdirectory (for example: **C:\Program Files\Eloquence\dlg\module\defaults.eq**).

IdmLib

IdmLib= *search path for dialog files*

Default: the value of the IDMLIB environment variable

Function: On execution of the **`DLG LOAD`** statement the specified dialog file is searched in each directory specified by the *IdmLib* item regardless of any directory specification named in the **`DLG LOAD`** statement. The directories are separated by '**`;`**'.

If *IdmLib* is specified in the *[dlgsrv]* section it overrides the value of the **`IDMLIB`** environment variable unless the token **`%IDMLIB%`** is included into the directory list.

If *IdmLib* is specified in an *user-defined section* it overrides the value of the *IdmLib* item specified in the *[dlgsrv]* section unless the token **`%IDMLIB%`** is included into the directory list.

Example:

```
IdmLib=C:\DLG\APP1;C:\DLG\APP2;%IDMLIB%
```

This causes every dialog file loaded by **`DLG LOAD`** to be searched in the directories **`C:\DLG\APP1`** and **`C:\DLG\APP2`**. If the dialog file cannot be found in these locations, searching is continued using the value of the previous *IdmLib* item, such as

the value of the IDMLIB environment variable.

DlgPath

DlgPath= *directory where dialog files are located*

Default: none

Function: If defined, DLGSRV32 will expect all .idm dialog files to be located in this directory.

NOTE: If *IdmLib* and/or the IDMLIB environment variable is present, *DlgPath* will be ignored. *DlgPath* serves for backwards compatibility and may not be supported in future releases. *IdmLib* is much more flexible and should be used instead.

Memsz

MemSz= *size of communication memory*

Default: 8192

Function: The communication memory size limits the amount of data which can be exchanged in a single network transaction. A value of 8192 is recommended.

Debug

Debug= *debug level*

Default: 0

Function: Specifies the debug level for diagnostic purposes (0, 1, 2 or 3).

LogFile

Logfile= *absolute path to debug log file*

Default: none

Function: If specified, debug log messages are written to this file. This is only effective if *Debug* is nonzero (see above).

Dlg

Dlg= *dialog file extension*

Default: dlg

Function: The file extension (case insensitive) by which Eloquence recognizes a native Eloquence dialog file (i.e. an ASCII dialog file).

Whenever Eloquence tries to load a Dlg file (e.g. **sample.dlg**) it will first try to locate and load an Idc (e.g. **sample.idc**) or Idm (e.g. **sample.idm**) file.

Idm

Idm= *dialog file extension*

Default: idm

Function: The file extension (case insensitive) by which Eloquence recognizes a Dialog Manager dialog file.

Such files are usually created by the **`cvdlg`** utility or the dialog editor of the Eloquence IDE.

Idc

Idc= *dialog file extension*

Default: idc

Function: The file extension (case insensitive) by which Eloquence recognizes a compiled Dialog Manager dialog file.

HelpDDEModule

Eloquence can use a web browser as on-line help viewer. This browser must understand the Netscape DDE commands (in particular: the WWW_OpenURL and WWW_Activate DDE topics) which is known to be the case for the Netscape and Mozilla browsers as well as for the Microsoft Internet Explorer.

HelpDDEModule= *module name of browser to be used as on-line help viever*

Default: Netscape

Function: Specifies which web browser shall be used as on-line help viewer.

Example: **`HelpDDEModule=iexplore`**

This specifies to use the Microsoft Internet Explorer as on-line help viewer.

NOTE: This functionality uses the Windows DDE protocol, so depending on the browser you want to use you should provide a corresponding entry in the section *[modules]*. If this is missing, DLGSRV32 will not be able to automatically start the on-line help viewer in case it is not already running. Please refer to the description of *Section [modules]* below and to the *Eloquence Graphical User Interface* documentation for details.

HelpDDEWait

HelpDDEWait= *milliseconds*

Default: 2500

Function: Number of milliseconds to wait after the on-line viewer has been started and before the DDE transaction is initiated.

If you encounter the status message "Module ... is not active and I don't know how to start it" although the on-line viewer is correctly defined (see *Section [modules]* below) you should increase this value.

HelpDDETimeout

HelpDDETimeout= *milliseconds*

Default: 15000

Function: Number of milliseconds to wait for a DDE transaction to finish.

If you encounter the status message "DDE server did not acknowledge" you should increase this value.

HelpBaseURL

HelpBaseURL= *URL*

Default: none

Function: Specifies the base URL of the on-line help files.

Example: `HelpBaseURL=http://www/help/`

FileBaseURL

FileBaseURL= *URL*

Default: none

Function: Specifies the base URL for the function **`EqHelpViewFile`**.

Example: `FileBaseURL=http://www/files/`

ManBaseURL

ManBaseURL= *URL*

Default: none

Function: Specifies the base URL for the function **`EqHelpManPage`**.

Example: `ManBaseURL=http://www/cgi-bin/man2html`

Section [modules]

This section is used by RUNSRV32 and DLGSRV32 in context with the Windows DDE communication protocol.

It associates Windows *module identifiers* with executable programs to enable RUNSRV32 and DLGSRV32 to start-up the required DDE server program in case it is not already running.

If you want to communicate with any Windows program using DDE, knowledge about the Windows *module identifier* and the communication topics specific to the program is required. Normally, these informations are contained in the program's (developer) documentation.

Example:

```
[modules]
WinWord=C:\Program Files\Microsoft Office\WINWORD.EXE
```

This associates the *module identifier* of *Microsoft Word* with its executable which enables RUNSRV32 to establish DDE communications with *Microsoft Word*.

DLGSRV32 has built-in support for DDE communication with a web browser to operate as on-line help viewer. To use this, you do not need any information regarding the browser's DDE implementation. All you need to do is to add an entry in the section *[modules]* so that the browser can be started in case it is not already running.

Example:

```
[modules]
Netscape=C:\Program Files\Netscape\Navigator\NETSCAPE.EXE
iexplore=C:\Program Files\Plus!\Microsoft Internet\Iexplore.exe
```

NOTE: Please refer to the *Eloquence Graphical User Interface* documentation for details about *On-line Help* and *RUNSRV DDE Communication*.

User-defined sections

User-defined sections are used by DLGSRV32. To illustrate this, let us recall the syntax of the **`DLG SET ".driver"`** statement:

`DLG SET ".driver","driver_spec [ini_section [arguments]]"`

driver_spec: *@hostname*

hostname is the name of the system where DLGSRV32 shall be executed.

ini_section: Optional name of an *user-defined section* in the **`eloqcl.ini`** file which shall override the defaults specified in the *[dlgsrv]* section.

arguments: Additional arguments can optionally be specified here and will be passed-through to the Dialog Manager.

When DLGSRV32 starts, the following tasks are performed:

1 DLGSRV32 sets up the *Dialog Manager argument list* from the **`arguments`** specified

in the **DLG SET ".driver"** statement, if any. It then reads the configuration items in the section *[dlgsrv]* (please refer to the previous description of *Section [dlgsrv]*).

2 If **ini_section** is specified in the **DLG SET ".driver"** statement, DLGSRV32 reads additional configuration items from this *user-defined section*. If these items have been previously specified in the section *[dlgsrv]*, the previous values are overridden.

Example:

```
[debug]
DefaultsFile = C:\DLG\DEBUG\DEFAULTS.EQ
```

This *user-defined section* named *[debug]* defines one item *DefaultsFile*. In order to activate this item, the name of this section must be specified in the **DLG SET ".driver"** statement, e.g.:

```
DLG SET ".driver","@my-pc debug"
```

Any previous *DefaultsFile* definition in section *[dlgsrv]* is overridden with the new value from section *[debug]*.

3 If **ini_section** is specified in the **DLG SET ".driver"** statement, DLGSRV32 searches this *user-defined section* for an item named *Arguments*. If this item exists, its value is appended to the *Dialog Manager argument list*.

Example:

```
[debug]
Arguments = -IDMtracefile C:\TMP\IDMTRACE.TXT
```

This enables an additional Dialog Manager argument which creates a trace file for debugging purposes.

4 Finally, the composed *Dialog Manager argument list* is passed to the Dialog Manager runtime system start-up function.

Any configuration item valid for section *[dlgsrv]* may also be defined in *user-defined sections* (please refer to the previous description of *Section [dlgsrv]*).

Additionally, *user-defined sections* may specify the following configuration item:

Arguments

Arguments= *additional arguments*

Default: none

Function: Additional arguments specific to the current *user-defined section* can optionally be specified here and will be passed-through to the Dialog Manager as described above.

4

Installing Eloquence on the Linux platform

This chapter covers the installation of Eloquence on the Linux platform

- Software and patch installation
- Configuring the operating system
- Configuring Eloquence

Installation Overview

This chapter describes the installation of the Eloquence product on the Linux platform and the configuration of the Eloquence software on Linux.

Please read the Eloquence release notes *before* installing or upgrating Eloquence as the release notes may include additional or more recent information.

Eloquence B.07.00 on the Linux platform requires a glibc 2.x based Linux distribution and has been tested against SuSE Linux and RedHat Linux distributions. It supports SuSE Linux versions 7.0 or newer and RedHat Linux versions 7.0 or newer. Eloquence is not distribution specific and other Linux distributions are expected to work as well but have not been tested.

Eloquence B.07.00 provides different versions for Linux based systems:

- glibc 2.2 based Linux distributions, for example SuSE Linux versions 8.0 or newer and RedHat Linux versions 7.x
- glibc 2.1 based Linux distributions, for example SuSE Linux before version 8.0 and RedHat Linux before version 7.0
- RedHat Linux versions 8.0 or newer are covered by a separate Eloquence build due to incompatibilities with the glibc version

Updating from previous Eloquence versions

If you are updating from a previous Eloquence revision, there are some special considerations which should be taken into.

- The Eloquence configuration files may have changed. You may want to consider comparing the Eloquence configuration files with the template configuration files.
- When using the Eloquence Personal Edition license you may need to copy the template license file manually.

NOTE: Any previous HP Eloquence product (B1368B) must be uninstalled before Eloquence can be installed. A rpm update is not possible. Please refer to the section *Updating from a previous HP Eloquence version* below for details.

Eloquence B.07.00 license

Eloquence B.07.00 requires a license key version B.07.00 or above. For a new Eloquence installation or when upgrading from a previous Eloquence version a new license key must be obtained to use the Eloquence server components.

If you are using the commercial Eloquence version, please request the appropriate license key *before* updating Eloquence. Currently, no temporary license key is generated on the Linux platform during installation.

The default license file includes a license key for the Eloquence Personal Edition.

The new license key can be requested by either submitting the Form enclosed with your Eloquence software delivery or on-line at the Eloquence web site: **http://www.marxmeier.com/eloquence/license/**

Please comment out any previous entries when adding the new B.07.00 license key to the license file **`/etc/opt/eloquence6/license`**. Otherwise the new license may not be recognized and **`chklic`** might output a message like below:

```
B.06.32: Bad license key revision.
Duplicate sequence number: Ignoring license
```

Installation on Linux

Eloquence B.07.00 on the Linux platform requires a glibc 2.x based Linux distribution and is installed in the **`/opt/eloquence6`** directory.

Eloquence is installed and updated using the **`rpm`** operating system utility.

The Eloquence installation archives may be obtained from a CD-ROM media or downloaded from the Internet.

glibc 2.2 based Linux distributions

It is recommended to use the Eloquence glibc2.2 build with recent Linux distributions which are based on the glibc 2.2 library (such as SuSE Linux versions 8.0 or newer or RedHat Linux versions 7.x).

Please make sure that at least the following shared library versions are installed:

glibc 2.2.5	libc.so.6 libm.so.6 ld-linux.so.2
ncurses	libncurses.so.5.2

Eloquence provides its own copy of the following libraries in the **`/opt/eloquence6/lib`** directory:

libstdc++	libstdc++-libc6.2-2.so.3

They have been included to solve compatibility problems with various Linux distributions using possibly incompatible versions of these libraries.

Eloquence B.07.00 has been compiled with gcc-2.95.3 for the glibc2.2 build.

glibc 2.1 based Linux distributions

For compatibility with previous Linux distributions (such as SuSE Linux before version 8.0 or RedHatLinux before version 7.0) a build based on the glibc 2.1 library is available.

Please make sure that at least the following shared library versions are installed:

glibc 2.1.2	libc.so.6 libm.so.6 ld-linux.so.2
ncurses	libncurses.so.4.2

libstdc++ libstdc++-libc6.1-1.so.2

Eloquence provides its own copy of the following libraries in the **`/opt/eloquence6/lib`** directory:

ncurses libncurses.so.4.2

libstdc++ libstdc++-libc6.1-1.so.2

They have been included to solve compatibility problems with various Linux distributions using possibly incompatible versions of these libraries.

Eloquence B.07.00 has been compiled with egcs-1.1.2 for the glibc2.1 build.

RedHat 8.x based Linux distributions

Due to incompatibilities with the glibc version used with RedHat versions 8.0 or newer a separate build has been created.

The following problems are currently known with Red Hat Linux 8.0 and above:

- eloqdb6 crashes with a segmentation violation (SIGSEGV) when reverse DNS lookup is enabled. Please make sure to set **NoDNS=1** in the **`eloqdb6.cfg`** configuration file of the eloqdb6 database server (this is the default as of Eloquence B.06.32).

  This problem might be related to the glibc version used in RedHat versions 8.0 and above and is not known to happen with other glibc versions.

- Eloqcore encounters unexpected SIGIO signals. Eloqcore uses the SIGIO signal to get a notification of pending keyboard (or TIO and network) events. On RedHat Linux versions 8.0 and above a SIGIO is also encountered when writing to stdout. This causes a performance degradation and may result in subsequent failures due to corner cases.

  This might indicate a problem in the Linux kernel as used in RedHat versions 8.0 and above and is not known to happen with other Linux distributions.

This Eloquence B.07.00 build has been compiled with gcc-3.2 and depends on the glibc-2.2.93 library version.

Linux kernel version

Eloquence should work with any Linux kernel version 2.2 or newer. You are encouraged to use the latest stable kernel version which fits your installed system. New kernel versions include bug fixes, updated drivers and security fixes. Your distribution should provide an updated kernel package which fits your base system.

We do not recommend to use early Linux 2.4 kernels (before version 2.4.6) in production environments.

For new installations, a recent glibc 2.2 based distribution (such as SuSE Linux versions 7.0 or newer or RedHat Linux versions 7.0 or newer) is recommended.

Updating from a previous HP Eloquence version

When updating from a previous HP Eloquence release (B1368B product) the following actions should be performed in additon to updating the software.

Eloquence cannot update a previous HP Eloquence installation. Eloquence uses a different package name than previous HP Eloquence releases and cannot be installed without uninstalling any previous HP Eloquence release.

To uninstall a previous HP Eloquence release, execute the command below:

```
rpm -e B1368B
```

This removes the HP Eloquence product but leaves your configuration files in place.

Shutting down Eloquence before installation

To update an existing Eloquence installation please shut down any running Eloquence servers before installing the new software.

On SuSE Linux versions 7.1 and newer, use the command below:

```
/etc/init.d/eloq6 stop
```

On SuSE Linux versions before 7.1 use the command below:

```
/sbin/init.d/eloq6 stop
```

On RedHat Linux, use the command below:

```
/etc/rc.d/init.d/eloq6 stop
```

After having installed Eloquence B.07.00, use the analogous **`eloq6 start`** command to activate the Eloquence servers.

Installing from CD-ROM media

As “superuser” follow the steps below to install the Eloquence software.

- Use a command line like below to mount the CD-ROM media.

  ```
  mount -t iso9660 /dev/cdrom /cdrom
  ```

  where */dev/cdrom* is the device file associated with the CD-ROM drive and */cdrom* is the directory where the CD-ROM should be mounted.

The Eloquence software for the Linux platform is in the **`B0700/linux`** subdirectory.

- Choose the appropriate installation package for your Linux system:
  - **Eloquence-B0700-glibc2.2-*.i386.rpm** - glibc 2.2 based Linux distribution
  - **Eloquence-B0700-glibc2.1-*.i386.rpm** - glibc 2.1 based Linux distribution
  - **Eloquence-B0700-rh8-*.i386.rpm** - RedHat Linux versions 8.0 or newer
- Install Eloquence B.07.00 with **`rpm`**. The following command is used both for a new installation and a product update:

  ```
  rpm -U /cdrom/B0700/linux/Eloquence-B0700-<arch>-*.i386.rpm
  ```

  where */cdrom* is the directory where the cdrom is mounted and <*arch*> is the architecture (such as glibc2.2).

Download Installation

Eloquence can be downloaded from the Eloquence web site at:
http://www.marxmeier.com/eloquence/download

Download the appropriate version of Eloquence for your Linux system

- **Eloquence-B0700-glibc2.2-*.i386.rpm** - glibc 2.2 based Linux distribution
- **Eloquence-B0700-glibc2.1-*.i386.rpm** - glibc 2.1 based Linux distribution
- **Eloquence-B0700-rh8-*.i386.rpm** - RedHat Linux versions 8.0 or newer

As “superuser” install Eloquence B.07.00 with **`rpm`**. The following command is used both for a new installation and a product update:

```
rpm -U /tmp/Eloquence-B0700-<arch>-*.i386.rpm
```

where */tmp* is the directory where the Eloquence installation package is located and <*arch*> is the architecture (such as glibc2.2).

NOTE:

Installation of automatic startup of Eloquence servers during boot is only performed for SuSE Linux and RedHat Linux distributions. If you are using a Linux distribution different from SuSE or RedHat you need to add the start and shutdown scripts manually. Template files are provided in the directory **`/opt/eloquence6/newconfig/startup`**.

Installing Eloquence Patches

Eloquence patches are provided to either fix a defect or limitation that was found after release and/or to add new functionality.

For proper operation of the Eloquence software, it is important that the recent patches are installed. For this purpose, a patch bundle is provided which can be used to conveniently install all recommended patches at once.

The **/B0700/patch/bundle** directory on the CD-ROM media contains the current Eloquence patch bundle at the time of compilation. The individual Eloquence patch files are contained in the **/B0700/patch** directory.

To find out about the most recent patches, please visit the Eloquence web site at: **http://www.marxmeier.com/eloquence/support/B07/patch.html**

NOTE: Installation of the "recommended" set of Eloquence patches is strongly recommended and may be required for correct product function.

Installation instructions

To install a patch or a patch bundle you need to uncompress and unpack it with tar. Installation requires root privileges.

```
cd /opt/eloquence6
tar xzf /path/to/PE70-<id>-linux-<arch>.tar.gz
```

where <id> is the patch id (such as 0304041) and <arch> is the patch architecture (such as glibc2.2). The architecture should be equivalent to the Eloquence distribution that was installed initially.

The following architectures are supported with Eloquence B.07.00:

- **glibc2.2** - glibc 2.2 based Linux distribution
- **glibc2.1** - glibc 2.1 based Linux distribution
- **rh8** - RedHat Linux versions 8.0 or newer

Linux patches include a README file that is placed in the **share/doc** subdirectory for later reference.

Configuring Eloquence on Linux

Your system might require additional configuration after the initial installation of the Eloquence software:

- Configure or verify service names and port numbers
- Install the license key
- Configure automatic startup/shutdown
- Create new users and groups for use with Eloquence

Configure or verify service names and port numbers

You may want to verify the service names and port numbers added to your **`/etc/services`** file during installation are appropriate.

In your **`/etc/services`** file should be entries like below. Please make sure the specified port numbers are available on your system:

```
eloqsd    8100/tcp    # Eloquence A.06.00 eloqsd server
eloqdb    8102/tcp    # Eloquence A.06.00 eloqdb6 server
eloqdb5   8104/tcp    # Eloquence A.06.00 eloqdb5 server
runsrv    8010/tcp    # Eloquence RunSRV (Windows)
```

where the first column specifies the service name (eg. eloqsd) and the second column the associated port number and protocol (eg. 8100/tcp). The selected port numbers may not already be in use by another programs.

You may want to define the Eloquence specific service names in your **`/etc/services`** file (if you are using NIS it is probably required to do this on the NIS master server). This is optional, as you can specify the port number directly instead of a service name.

Install the license key

Add the supplied license keys to the **`/etc/opt/eloquence6/license`** file. We recommend that you comment out any existing license keys.

If a temporary license was created during installation this step may be skipped and the permanent license added later.

If the Personal Edition license should be used this step is not necessary as the PE license key is included in the default license template file.

The **`/opt/eloquence6/etc/chklic`** utility may be used to verify the license file.

Configure automatic startup/shutdown

The Eloquence server processes may be configured to start automatically by default if runlevel 2 is entered (after a reboot or single user mode) and shut down automatically on reboot.

The automatic server startup is configured in the Eloquence startup configuration file **`/etc/rc.config.d/eloquence6`**. Please refer to the next section for details

Create new users and groups for use with Eloquence

It is strongly recommended to run the eloqdb6 server and the eloqsd server with a dedicated user and group.

- **eloqsd** - a dedicated user and group should be created (default eloqsd/eloqsd) and configured in the eloqsd.cfg file. Please refer to the ssection on configuring the eloqsd process for details.
- **eloqdb6** - a dedicated user and group should be created (default eloqdb/eloqdb) and configured in the eloqdb6 configuration file. Please refer to the section on configuring the eloqdb6 process for details.

Eloquence startup/shutdown script

Manually starting and stopping the server processes

The Eloquence startup script may be used to manually start/stop/restart server processes or verify a server process is active.

Please note that using the startup/shutdown script to control server processes need root capabilities. Status inquiries do not require root capabilities.

The Eloquence startup/shutdown script supports the following options:

start [arg list ...]
Start the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or database server instances) are started. If the argument list is present, only the specified services (or instances) are started.

Please note: A database server instance is not be started automatically if the ELOQDB6_START[0] configuration option is set to 0.

stop [arg list ...]
Stop the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are stopped. If the argument list is present, only the specified services (or instances) are stopped.

Please note that the Eloquence start/stop script only manages configured services. Previous versions of the shutdown script stopped unconfigured Eloquence services as well.

restart [arg list ...]
Restart the configured Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are restarted.

status [arg list ...]
Display the status of Eloquence services. If the optional argument list is not provided, all configured Eloquence services (or instances) are returned.

info [arg list ...]
Display the startup configuration of Eloquence services. If the optional argument list is not provided, the configuration of all Eloquence services (or instances) is returned.

An optional argument list may be used to specify which Eloquence service(s) should be managed. An argument specifies either an Eloquence service or a database instance which is configued in the startup configuration file.

eloqsd	Specfies the Eloquence eloqsd service
eloqdb6	Specfies all configured Eloquence eloqdb6 instances

For example:

```
Start all configured services
# /etc/init.d/eloq6 start

Restart eloqsd service
# /etc/init.d/eloq6 restart eloqsd

Stop all configured eloqdb6 instances
# /etc/init.d/eloq6 stop eloqdb6

Start eloqdb6 instances prod and test
# /etc/init.d/eloq6 start prod test

# /etc/init.d/eloq6 start
Starting eloqsd daemon                              [OK]
Starting eloqdb6[eloqdb] daemon                     [OK]

# /etc/init.d/eloq6 status
eloqsd process is active (pid 16645)                [running]
eloqdb6[eloqdb] process is active (pid 16652)       [running]

# /etc/init.d/eloq6 info eloqdb
 eloqdb6 instance id = eloqdb
  configuration file = /etc/opt/eloquence6/eloqdb6.cfg
  service = eloqdb
  args =
  run prefix =
  automatic start = 1
```

Location of the Eloquence startup/shutdown script

The location of the Eloquence startup/shutdown script depends on the operating system:

HP-UX	/sbin/init.d/eloq6
SuSE Linux	/etc/init.d/eloq6 (SuSE Linux 7.1 and above) /sbin/init.d/eloq6 (SuSE Linux before 7.1)
Red Hat Linux	/etc/init.d/eloq6 or /etc/rc.d/init.d/eloq6
LSB compliant Linux	/etc/init.d/eloq6

Location of the Eloquence startup/shutdown configuration file

The Eloquence startup/shutdown script uses a new configuration template file. It is backwards compatible to previous Eloquence versions. The location of the startup configuration file depends on the operating system:

HP-UX	/etc/rc.config.d/eloquence6
SuSE Linux	/etc/sysconfig/eloquence6 (SuSE 8.0 Linux and above) /etc/rc.config.d/eloquence6 (SuSE Linux 7.3 and before)
Red Hat Linux	/etc/sysconfig/eloquence6
LSB compliant Linux	/etc/sysconfig/eloquence6

During the update from a previous Eloquence installation a previous configuration file is moved to the new location. However, we recommend to start with the configuration template file if updating from a previous HP Eloquence version.

The example command shown below is specific to a LSB compliant Linux distribution, please change the target location depending on the operating system.

```
cp /opt/eloquence6/newconfig/startup/eloquence.rc \
   /etc/sysconfig.d/eloquence6
```

Eloquence startup/shutdown configuration options

The Eloquence startup/shutdown configuration file specifies the operation of the startup/shutdown script. The following global configuration options are available:

START_ELOQ	If START_ELOQ is set to anything besides 1 it disables the automatic Eloquence startup entirely. To retain compatibility to previous Eloquence releases on Linux "yes" is accepted as well. The default is START_ELOQ=1
START_ELOQSD	If START_ELOQSD is set to 1 the eloqsd daemon is started automatically. The default is START_ELOQSD=1
ELOQSD_ARGS	The ELOQSD_ARGS allows specifying eloqsd command line arguments. The default is ELOQSD_ARGS=""
ELOQSD_RUNPFX	The ELOQSD_RUNPFX variable allows specifying a command which is then expected to start eloqsd. The default is ELOQSD_RUNPFX="" For example: ELOQSD_RUNPFX="/usr/bin/nice -n10" starts eloqsd with a nice value of 10.
START_ELOQDB6	If START_ELOQDB6 is set to 1 then the eloqdb6 daemon is started automatically. The default is

START_ELOQDB6=1

ELOQDB6_DEFAULT_ARGS The ELOQDB6_DEFAULT_ARGS specifies the default command line arguments which are used with all eloqdb6 instances unless defined specifically. The default is ELOQDB6_DEFAULT_ARGS=""

The Eloquence startup/shutdown configuration file may be used to configure multiple eloqdb6 instances.

Unless specified here, the Eloquence startup/shutdown script only supports the default database server instance. This section allows specifying database server instances which are to be maintained by the eloquence startup script. Each database server instance must be specified with a distinct index, starting with the index 0.

The following options may be specified for each eloqdb6 instance:

ELOQDB6_CFG[0] This option specifies the configuratin file which is used with this database server instance (required). This file can be specified with an absolute path or relative to the directory /etc/opt/eloquence6
The default is ELOQDB6_CFG[0]=eloqdb6.cfg

ELOQDB6_ID[0] This configuration option may be used to specify a symbolic name to the database server instance (optional). If not specified, this defaults to the service name or port number, specified in the config file or "eloqdb". The instance id may be used as an optional argument with the Eloquence startup/shutdown script to specify a database server instance.

ELOQDB6_START[0] This configuration option may be used to specify if the Eloquence startup script should start this service automatically. If set to 1, the database server instance is started automatically. If set to 0, the startup script will ignore this entry for the start and restart option and the database server instance must be managed manually by providing the instance id name on the script command line.
The default is ELOQDB6_START[0]=1

ELOQDB6_ARGS[0] The ELOQDB6_ARGS[0] option may be used to specify eloqdb6 command line arguments for this database instance. If not specified this defaults to ELOQDB6_DEFAULT_ARGS.

ELOQDB6_RUNPFX[0] This configuration option may be used to specify a command which is then expected to start eloqdb6. The default is ELOQDB6_RUNPFX[0]=""

For example:

```
START_ELOQSD=0
ELOQDB6_CFG[0]=eloqdb6_prod.cfg
ELOQDB6_START[0]=1
ELOQDB6_ID[0]=prod
ELOQDB6_CFG[1]=eloqdb6_test.cfg
ELOQDB6_START[0]=1
ELOQDB6_ID[0]=test
```

This example configuration file specifies to not start the eloqsd service and defines two eloqdb6 instances, prod and test.

Eloquence startup/shutdown configuration file template

```
# @(#) eloquence.rc - B.07.00
# Eloquence automatic startup configuration
#
# Eloquence startup confguration file.
# This file is sourced by the startup/shutdown script.
#
# The location of this file depends on the operating system:
# HP-UX: /etc/rc.config.d/eloquence
# RedHat Linux, SuSE Linux 8.0+: /etc/sysconfig/eloquence
# SuSE Linux 7.0 to 7.3 /etc/rc.config.d/eloquence

### global settings

# If START_ELOQ is set to anything besides 1 it disables the
# automatic Eloquence startup entirely. To retain compatibility
# to previous Eloquence releases on Linux "yes" is accepted
# as well.
#START_ELOQ=1

# If START_ELOQSD is set to 1 the eloqsd daemon is started
# automatically. The ELOQSD_ARGS allows specifying eloqsd command
# line arguments. The ELOQSD_RUNPFX variable allows specifying
# a command which is then expected to start eloqsd (e.g. nice).
#START_ELOQSD=1
#ELOQSD_ARGS=""
#ELOQSD_RUNPFX=""

# If START_ELOQDB6 is set to 1 then the eloqdb6 daemon is started
# automatically. The ELOQDB6_DEFAULT_ARGS specifies the default
# command line arguments, which are used with all eloqdb6
# instances unless defined specifically.
#START_ELOQDB6=1
#ELOQDB6_DEFAULT_ARGS=""

### eloqdb6 instances ###

# Unless specified here, the Eloquence startup/shutdown script
# only supports the default database server instance. This
# section allows to specify database server instances which are
# to be maintained by the eloquence startup script.
# Each database server instance must be specified with a distinct
# index, starting with the index 0.
#
# ELOQDB6_CFG[0] =
```

```
#  Configuration file used with this database instance (required).
#  This can be an absolute path or relative to /etc/opt/eloquence6
#
# ELOQDB6_ID[0] =
#  The instance id is used to specify an alias to a eloqdb6 instance
#  (optional). If not specified, this defaults to the service name
#  or port number, specified in the config file (defaults to eloqdb).
#  The instance id may be used as a startup script argument to
#  specify a database server instance.
#
# ELOQDB6_ARGS[0]=""
#  Command line arguments for this instance (optional). If not
#  specified (eg. commented out), the ELOQDB6_DEFAULT_ARGS is used.
#
# ELOQDB6_START[0]={0|1}
#  Specifies if the Eloquence startup/shutdown script should make
#  use of this entry (optional). This setting defaults to 1 which
#  specifies a database instance is used with the startup script.
#  If set to 0, the startup script will ignore this entry for the
#  start and restart option.
#
# ELOQDB6_RUNPFX[0]=""
#  Allows to define a command which is then expected to start
#  the eloqdb6 instance (e.g. nice). On HP-UX this may be used
#  to define a memory window used by the database server instance.
#
#ELOQDB6_CFG[0]=eloqdb6.cfg
#ELOQDB6_START[0]=1
#ELOQDB6_ARGS[0]=""
#ELOQDB6_ID[0]=""
#ELOQDB6_RUNPFX[0]=""
```

Configuring the User Environment

There are two files where the user environment can be configured:

- The environment defined in **`/etc/profile`** affects all users.
- The file **`.profile`** located in a user's home directory configures the user-specific environment.

Perform the following steps:

- In order to make Eloquence executables accessible to all users, please edit the file **`/etc/profile`**. Search for a line containing the directive **`PATH=`** and append **`:/opt/eloquence6/bin`**.
- If you want to make Eloquence executables accessible only to specific users, edit the **`.profile`** files in the appropriate home directories and append **`:/opt/eloquence6/tools`** to the **`PATH=`** directive.
- Make sure that the terminal type is configured properly for all users.
- Make your configuration changes active. This is done by logging off and back on again.

Configuring the eloqsd server

The eloqsd server is an important part of Eloquence. It is responsible for the following tasks:

- Eloqsd coordinates the TASKID values.
- Eloqsd provides file sharing capabilities for the new graphical Eloquence development environment.
- Eloqsd is used to start eloqcore processes in the background.
- Eloqsd is used to count active users and does validate it against available user licenses.
- Eloqsd optionally provides a HTTP interface so server status information can be queried with a web browser.

NOTE:

The eloqsd server is not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you don't need to configure and run the eloqsd server.

Eloquence implements some limited file sharing capabilities for the new graphical development environment through the eloqsd server. This makes it independent of the availability of specific network file systems (NFS/ SMB) and overcomes inappropriate limitations.

Running an eloqsd server on your system is not mandantory unless you are using eloq (providing virtual terminal capabilities). However, when no eloqsd process is active, TASKID values are no longer unique and are set to 1 by default.

In order to run the eloqsd server it is required to adapt your system configuration. This involves the following steps:

- Configuring the eloqsd TCP service
- Configuring the default eloqsd account and group
- Configuring the eloqsd server startup
- Configuring the eloqsd server

Configuring the eloqsd TCP service

You may want to define the eloqsd service names in your /etc/services file. This is optional, as you can specify the port number directly in the eloqsd.cfg configuration file.

Please add lines like below to your **`/etc/services`** file:

```
eloqsd     8100/tcp    # Eloquence A.06.00 eloqsd server
```

the first column specifies the service name (eg. eloqsd) and the second column the associated port number and protocol (eg. 8100/tcp). The selected port numbers may not already be in use by another programs.

NOTE: All systems must use the same port numbers in order to communicate.

Configuring the default eloqsd account and group

The **`eloqsd`** server requires you to specify an account and group name in the configuration file. Whenever **`eloqsd`** is started with root capabilities it will switch to the specified account/group instead. This is required, because for one it is generally not a good idea to run programs with root capabilities unless necessary, on the other hand this is used as the default account and group for users accessing files through the eloqsd or starting a background process.

While you can specify any user or group account in the configuration file, we recommend to create a specific user account and group for Eloquence which is used by the eloqsd server.

We recommend to create the user account *eloqsd* and the group *eloqsd* which should have the account *eloqsd* as a member. The account should be marked "disabled" (by putting an asterisk into the password field of the **`/etc/passwd`** file) to prevent logins using the eloqsd account.

Configuring the eloqsd server startup

To specify thet the eloqsd server is started at system boot time, set the **`START_ELOQSD`** variable to **`1`** in the Eloquence startup/shutdown configuration file **`/etc/sysconfig/eloquence6`**. Change the line defining the **`START_ELOQSD`** variable like this:

```
START_ELOQSD=1
```

NOTE: The **/etc/sysconfig/eloquence6** location of the startup/shutdown configuration file applies to recent LSB compliant Linux distributions. Older distributions such as SuSE Linux 7.3 and before use the **/etc/rc.config.d/eloquence6** location instead.

Configuring the eloqsd server

The eloqsd server is configured by editing the **`eloqsd.cfg`**, **`eloqsd.user`** and **`eloqsd.share`** configuration files. All configuration files provide complete inline documentation and are included at the end of this section for your reference.

The configuration files are located in the **`/opt/eloquence6/etc`** directory. Each configuration file is responsible for a specific part of the eloqsd configuration:

eloqsd.cfg This is used for the general configuration of the server.

eloqsd.user Eloqsd provides its own user configuration. This makes it possible to define eloqsd users without the need to have a system account for each individual user. Instead eloqsd users are associated with system accounts and groups.

As passwords are defined in this file we consider it good practice to make this file unreadable for regular users. You should chown it to root and chmod id to 400.

eloqsd.share This configuration file is used to define resources which can be accessed through the eloqsd server.

The eloqsd command line options

The eloqsd server supports the following command line options which can be used to temporarily override configured settings in the **`eloqsd.cfg`** configuration file.

```
usage: eloqsd [options]
options:
 -help       = show usage (this list)
 -c name     = configuration file
 -d flags    = debug mode
 -l name     = log file name (or console/syslog/default)
 -f          = run in foreground
 -s name     = service name (tcp/ip transport)
 -F facility = syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
 -I ident    = syslog identifier
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-f	Run in foreground. This is used for debugging the eloqsd server.	

Option	Description	Equiv.*
-s name	The service name (as defined in /etc/services) or the port number where the server should listen for requests. The default value is eloqdb6.	Service
-F facility	When logging to the syslog daemon, you can define a syslog facility (USER/DAEMON LOCAL0..LOCAL7)	SysFacility
-I ident	When logging to the syslog daemon, you can define a syslog identifier. The default is eloqsd	SysIdent

*Equivalent configuration file directive.

The eloqsd HTTP status display

When the **ServiceHttp** is defined in the **`eloqsd.cfg`** configuration file, you can use a web browser such as Netscape or Mozilla to view the configuration and state of the eloqsd process in your network.

To access the eloqsd server, you need to provide an URL like below:

```
http://server:port/
```

where server is the host name or IP number of the system running the eloqsd server and port is the port number used for **ServiceHttp** in the **`eloqsd.cfg`** file.

Default eloqsd.cfg file

```
# eloqsd.cfg
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define the eloqsd properties.
# The location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.cfg
#   Linux: /etc/opt/eloquence6/eloqsd.cfg
#
# This file is read once at eloqsd startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
# Default values are provided commented out.

### Server configuration

[Config]

# Service       The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for requests. The default value is eloqsd.
#
# ServiceHttp   The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#
# UseKeepAlive  Numeric flag if the KEEP ALIVE socket option
#               should be used. Valid values are 1/0.
#               The default value is 1.
#               If this option is active, the server will check
#               after a system defined period of inactivity, if the
#               client is still alive.
#

#Service = eloqsd
#ServiceHttp =
#UseKeepAlive = 1

# panic         This option defines what should happen if a fatal
#               error is encountered.
#
#               The following options are valid:
#               exit    Terminate the process. This is the default.
#              dump    Terminate the process and create a core dump.
#
#            This is a problem tracking option. Unless you know what
#            you need the coredump for you probably want to stay with
#               the default

#panic = exit
```

```
# HttpFrame     Numeric flag if the links should be omitted in
#               HTTP status. The default value is 0.

# HttpFrame = 0

# Lang          This configuration option defines the locale, the
#               server should use. The default value is "C".
#               The only locale currently supported is "C".
#
# Messages      This configuration option defines the language
#               for server messages. This value defaults to Lang.
#               The only locale currently supported is "C".
#
# Charset       This defines the character set encoding, the server
#               should use internally.
#
#               Valid setings are:
#                  HPROMAN8 - HP Roman8
#                  ISO8859/1 - ISO 8859/1
#               The default value for HP-UX is HPROMAN8, all other
#               platforms default to ISO8859/1.
#
#               This setting is used by the server to translate
#               client strings like user or file names.

#Lang = C
#Messages = C
#Charset = HPROMAN8

# AuthPolicy    This entry specifies, how user names and passwords
#               are validated. The following entries are valid:
#
#               server  - The server will validate passwords
#                         and user names using eloqsd.user
#
#               The default value is "server".
#
# userFile      The path/name of the eloqsd.user file.
#               The default value depends on your operating system:
#               HP-UX: /etc/opt/eloquence6/eloqsd.user
#               Linux: /etc/opt/eloquence6/eloqsd.user
#
# shareFile     The path/name of the eloqsd.share file.
#               The default value depends on your operating system:
#               HP-UX: /etc/opt/eloquence6/eloqsd.share
#               Linux: /etc/opt/eloquence6/eloqsd.share

#AuthPolicy = server
#userFile = /etc/opt/eloquence6/eloqsd.user
#shareFile = /etc/opt/eloquence6/eloqsd.share

# DefaultUID   The default name (or numeric id) of the system account
#             to run client processes as, unless a different setting
#               is provided for the user.
#
# DefaultGID   The default name (or numeric id) of the system group
#             to run client processes as, unless a different setting
#               is provided for the user.

DefaultUID = eloqsd
```

```
DefaultGID = eloqsd

# LogFile       This defines where log messages are written to.
#              This configuration value either specifies a path/file
#               or one of the keywords below:
#
#               console  - log messages are written to the console
#               syslog   - log messages will be sent to the
#                          syslog daemon
#
#               The default value is "syslog".

LogFile = syslog

# SysIdent      When logging to the syslog daemon, you can define
#               a syslog identifier. Default is eloqsd.
#               See syslogd(1M) for more information
#
# SysFacility   When logging to the syslog daemon, you can define
#               a syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
#               The default setting is "USER".
#               See syslogd(1M) for more information

#SysIdent = eloqsd
#SysFacility = USER

# LogFlags      Each log message has an associated origin and
#               severity. The log flags define, which messages will
#             be logged. The "*" origin matches all message origins,
#               so it can be used to setup a default which can be
#              overriden for a specific message origin (eg. "*1N0"):
#               Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  C = Configuration subsystem
#                  N = Network transport
#                  P = Protocol handling
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#            Enabling log messages with L_DEBUG or L_VDEBUG severity
#               may result in a huge number of log messages.
#            To enable only fatal messages, you would want to set the
#              LogFlags to "*0", to enable regular log messages you
#               would want to set the LogFlags to "*1"

LogFlags = *0

# Configuration items below are the more traditional eloqsd
```

```
# settings.
#
# MaxUsers      Maximum number of eloqcore processes on the local
#               system. The default value is 40.
#
# MaxTasks     Maximum number of TASKIDs to reserve for "secondary"
#             eloqcore processes. If you don't know what this is good
#               for, you probably don't need it :-)
#               The default value is 20
#

MaxUsers = 40
MaxTasks = 20
```

Default eloqsd.user file

```
# eloqsd.user
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define all users which are known to
# Eloquence. The location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.user
#   Linux: /etc/opt/eloquence6/eloqsd.user
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# This makes it possible to define Eloquence users without the
# need to have a system account for each individual user.
# As passwords are defined in this file we consider it good practice
# to make this file unreadable for regular users. You should chown
# it to the administrator (probably root) and chmod id to 400.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each user definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [user_id]
# Name       The full user name (currently unused)
# Email      Email address of the user (currently unused)
# Password   The user password. This is currently clear text.
# uid        System account to execute client processes
# gid        System group to execute client processes
# Profile    Template user entry. User defaults will be taken from
#            this section.
# Home      Home path. Defaults to the home directory associated to
#            the UID by the system.
#
# There are two predefined sections:
#
# [public] is used, if a client does does not provide a user id. This
# can only happen, if an eloqcore has been started locally and
# requests a remote operation. (currently unused)
#
# [default] is used as the default user profile.

[public]
Name = Anonymous

[default]
Name = Default user profile

[demo]
Name = Jon Doe
```

```
Password = secret
```

Default eloqsd.share file

```
# eloqsd.share
#
# @(#) $Revision: B.07.00.2 $
# The purpose of this file is to define all disk resources which are
# known to Eloquence. The location depends on the operating
# system:
#
#   HP-UX: /etc/opt/eloquence6/eloqsd.share
#   Linux: /etc/opt/eloquence6/eloqsd.share
#
# This file is read at the startup time of the eloqsd process.
# Changes are automatically detected and honored.
#
# Eloquence A.06.00 provides its own file sharing capabilities.
# This will make you independent of the availability of specific
# network file systems (NFS/SMB) and overcomes possible file system
# limitations.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.
#
# Each share definition is a different section.
#
# The following configuration items are recognized for each section:
#
# [share_id]
# Path      Absolute path
# Comment   Share description. This is displayed by the client.
# Users    Comma separated list of individual users or user profiles
#           (currently unused)

[example]
Path = /opt/eloquence6/share
Comment = Eloquence shared files
```

Configuring the eloqdb6 server

The eloqdb6 is the Eloquence database server. Eloquence uses a client/server database approach.

In order to run the eloqdb6 server it is required to adapt your system configuration. This involves the following steps:

- Configuring the eloqdb TCP service
- Configuring the default eloq account and group
- Configuring the eloqdb6 server startup
- Configuring the eloqdb6 server
- Creating the database environment

Configuring the eloqdb TCP service

It is recommended, that you define the eloqdb service names in your /etc/services file. This is optional, as you can specify the port number directly in the eloqsd.cfg configuration file.

Please add lines like below to your **`/etc/services`** file:

```
eloqdb      8102/tcp    # Eloquence A.06.00 eloqdb6 server
```

the first column specifies the service name (eg. eloqdb6) and the second column the associated port number and protocol (eg. 8102/tcp). The selected port numbers may not already be in use by another programs.

NOTE: All systems must use the same port numbers for the same service in order to communicate.

You can have more than one instance of the eloqdb6 server running on a single system, however they must use different services/port numbers.

NOTE: Multiple eloqdb6 instances are covered in detail in the section *Setting up multiple eloqdb6 instances* below.

Configuring the default eloqdb account and group

The eloqdb6 server requires you to specify an account and group name in the configuration file. Whenever it is started with root capabilities it will switch to the specified account/group instead. This is required, because for one it is generally not a good idea to run programs with root capabilities unless necessary. In addition, all data base volumes are owned by this user and are thus protected from illegal access from other users.

While you can specify any user or group account in the configuration file, we recommend to create a specific user account and group for Eloquence which is used by the eloqdb6 server.

We recommend to create the user account *eloqdb* and the group *eloqdb* which should have the account *eloqdb* as a member. The account should be marked "disabled" (by putting an asterisk into the password field of the **`/etc/passwd`** file) to prevent logins using the eloqdb account.

Configuring the eloqdb6 server startup

If you want to specify that the eloqdb6 server is started at system boot time, set the **`START_ELOQDB6`** variable to **`1`** in the Eloquence startup/shutdown configuration file **`/etc/rc.config.d/eloquence6`**. Change the line defining the **`START_ELOQDB6`** variable like this:

```
START_ELOQDB6=1
```

NOTE:

The **/etc/sysconfig/eloquence6** location of the startup/shutdown configuration file applies to recent LSB compliant Linux distributions. Older distributions such as SuSE Linux 7.3 and before use the **/etc/rc.config.d/eloquence6** location instead.

Configuring the eloqdb6 server

The eloqsd server is configured by editing the **`eloqdb6.cfg`** configuration file. It is located in the **`/opt/eloquence6/etc`** directory. The eloqdb6.cfg configuration file provides complete inline documentation. The default configuration file is included at the end of this section for your reference.

The default configuration is not optimized for performance and does not handle a large number of concurrent connections. Therefore, you should adjust the following parameters in the eloqdb6.cfg configuration file to your requirements:

- Section **[Server]**, parameters **UID** and **GID**

  UID and GID must be set to the default eloqdb account as explained in the previous section *Configuring the default eloqdb account and group*.

- Section **[Server]**, parameter **EnableIPC**

  If the EnableIPC configuration item is set, eloqdb6 is enabled to make use of shared memory for communication between client and server. This results in a 25% performance increase.

  The recommended setting is **EnableIPC = 2**.

- Section **[Config]**, parameter **Threads**

This parameter specifies how many connections to the database server can be established concurrently. The default is 40 which means that a maximum of 40 users can access the database at the same time. Multiple programs accessing the database count as multiple connections, while accessing multiple databases from within the same program counts as one single connection.

- Section **[Config]**, parameter **BufferCache**

  The recommended minimum value is 64 (megabytes). A higher value is recommended. The current limit is approx. 1 gigabyte. The default and minimum buffer cache size is 5 Megabytes which may lead to bad performance.

- Section **[Config]**, parameter **CheckPtSize**

  Whenever the transaction journal exceeds this size (in megabytes), eloqdb6 performs an internal checkpoint operation to recycle the journal. The default size is 10 megabytes which probably causes the checkpoint to happen too often, causing a performance impact. It is recommended to set this parameter to 20 or higher.

NOTE: If the configuration of a running eloqdb6 server is changed, it must be restarted to activate the new configuration.

Creating the database environment

A database environment consists of at least a primary data volume and a transaction log volume. These must be created on a local disk before the eloqdb6 can be started for the first time.

On Linux, the size of a single volume file is currently limited to 2 gigabytes.

Additional data volumes can be created at any time to extend the available database storage. Additional transaction log volumes can be added as well, but it is unlikely that these will ever be used because the purpose of a log volume is to hold temporary data only.

The maximum number of volume files in a database environment is 255 which is equivalent to a maximum database storage size of 510 gigabytes (254 data volumes and 1 log volume).

It is recommended to choose a dedicated directory to hold all volume files belonging to a particular database environment. The following example assumes that the initial volumes are created in the /data/db directory:

1 Change to the directory where the volumes shall be created:

```
cd /data/db
```

2 Create the initial data volume:

```
dbvolcreate data.vol
```

3 Create the initial transaction log volume:

```
dbvolextend -t log log.vol
```

Both files are created with their initial minimum size which is 2.5 megabytes. They will grow on demand until they reach their maximum size. The minimum and maximum size and the amount by which the files shall grow can be configured either at creation time or afterwards with the **dbvolchange** utility.

To view the available command line options, use:

```
dbvolcreate -help
dbvolextend -help
dbvolchange -help
```

The **dbvolcreate** and **dbvolextend** utilities automatically maintain the list of volume files in the **[Volumes]** section of the eloqdb6.cfg configuration file.

The eloqdb6 command line options

The eloqdb6 server supports the following command line options which can be used to temporarily override configured settings in the configuration file.

```
usage: eloqdb6 [options]
options:
 -help       = show usage (this list)
 -c name     = configuration file
 -d flags    = debug mode
 -l name     = log file name (or console/syslog/default)
 -f          = run in foreground
 -s name     = service name (tcp/ip transport)
 -F facility = syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
 -I ident    = syslog identifier
```

Option	Description	Equiv.*
-c name	Specifies the configuration file name	
-d flags	Specifies the server log flags.	LogFlags
-l name	Specifies the server log file.	LogFile
-f	Run in foreground. This is used for debugging the eloqdb6 server.	
-s name	The service name (as defined in /etc/services) or the port number where the server should listen for requests. The default value is eloqdb6.	Service

Option	Description	Equiv.*
-F facility	When logging to the syslog daemon, you can define a syslog facility (USER/DAEMON LOCAL0..LOCAL7)	SysFacility
-I ident	When logging to the syslog daemon, you can define a syslog syslog identifier. The default is eloqdb6	SysIdent

*Equivalent configuration file directive.

The eloqdb6 HTTP status display

When the **ServiceHttp** is defined in the **`eloqdb6.cfg`** configuration file, you can use a web browser such as Netscape or Mozilla to view the configuration and state of the eloqdb6 server in your network.

To access the eloqdb6 server, you need to provide a URL like below:

```
http://server:port/
```

where server is the host name or IP number of the system running the eloqdb6 server and port is the port number used for **ServiceHttp** in the **`eloqdb6.cfg`** file.

Setting up multiple eloqdb6 instances

Multiple instances of the eloqdb6 server can coexist on the same system. This makes sense if multiple database environments should be kept separate from each other, either to keep the databases in separate namespaces or simply to maintain discrete database storage.

The following steps are required to setup an additional eloqdb6 server instance:

1 Each eloqdb6 instance needs a separate configuration file. You can copy either the configuration of any existing instance or the **eloqdb6.cfg** template file located in the **newconfig/config** subdirectory of your Eloquence installation.

In this example, we create a new configuration file named eloqdb6.instance.2.cfg in the default /etc/opt/eloquence6 configuration directory. This is not mandatory, the configuration file may have any name and can be located anywhere.

Please note that the **`cp`** command below must be entered as a single line:

```
cd /etc/opt/eloquence6
cp /opt/eloquence6/newconfig/config/eloqdb6.cfg
 eloqdb6.instance.2.cfg
```

Next, the new configuration file must be edited. Besides the parameters mentioned in the previous section *Configuring the eloqdb6 server*, the following parameters must be adjusted:

- Section **[Server]**, parameter **Title**

  The server title is displayed when you list the eloqdb6 processes with **`ps`**. You should specify an unique title for each eloqdb6 instance to be able to distinguish the different instances in the process listing.

- Section **[Server]**, parameter **Service**

  Each instance requires an unique TCP service name or port number. You can either configure a dedicated service name in your local SERVICES file (please refer to the previous section *Configuring the eloqdb6 TCP service* for details) or simply enter an unique port number.

- Section **[Server]**, parameter **ServiceHttp**

  If you use the eloqdb6 HTTP status display, you are required to configure an unique TCP service name or port number for each eloqdb6 instance.

- Section **[Server]**, parameter **LogFile** (optional)

  By default, all eloqdb6 log messages are written to the syslog. Each log message is labeled with the process id of the originating eloqdb6 instance. However, it might be more convenient to configure a separate log file for each instance. This log file could be located within the dedicated, instance-specific directory (see below), for example:

  ```
  LogFile = /data/db/instance.2/eloqdb6.log
  ```

NOTE: If you copied the configuration of an existing instance, it is required that you manually delete all volume references at the end of the file below the **[Volumes]** section header.

2 Create the instance-specific database environment. You do this according to the previous section *Creating the database environment*, but you use the **`-c`** command line option to refer to the instance-specific configuration file:

- Create the instance-specific directory. We recommend to create a dedicated directory for each instance where the instance-specific files are located:

  ```
  cd /data/db
  mkdir instance.2
  cd instance.2
  ```

- Create the initial data volume (the following command must be entered as a single line):

  ```
  dbvolcreate
   -c /etc/opt/eloquence6/eloqdb6.instance.2.cfg data.vol
  ```

- Create the initial transaction log volume (the following command must be entered as a single line):

```
dbvolextend
 -c /etc/opt/eloquence6/eloqdb6.instance.2.cfg -t log log.vol
```

In this example, the argument to the **-c** command line option refers to the instance-specific configuration file located in the current directory.

3 Create an entry for the new eloqdb6 instance at the end of the **/etc/sysconfig/eloquence6** startup/shutdown configuration file:

```
ELOQDB6_CFG[1]=eloqdb6.instance.2.cfg
ELOQDB6_START[1]=1
ELOQDB6_ID[1]=instance.2
```

Use the next available index if there is already an instance defined which uses the index **[1]**. ELOQDB6_CFG refers to the instance-specific configuration file. If this is located outside the default /etc/opt/eloquence6 directory, the absolute path must be provided. If you don't want this instance to be started at system boot time, set the ELOQDB6_START variable to 0.

NOTE: The **/etc/sysconfig/eloquence6** location of the startup/shutdown configuration file applies to recent LSB compliant Linux distributions. Older distributions such as SuSE Linux 7.3 and before use the **/etc/rc.config.d/eloquence6** location instead.

The new eloqdb6 instance is now configured. You can control it using the **/etc/init.d/eloq6** startup/shutdown script as follows:

```
/etc/init.d/eloq6 start instance.2
/etc/init.d/eloq6 status instance.2
/etc/init.d/eloq6 restart instance.2
/etc/init.d/eloq6 stop instance.2
```

NOTE: The **/etc/init.d/eloq6** location applies to recent LSB compliant Linux distributions. Older distributions may use the **/sbin/init.d/eloq6** or **/etc/rc.d/init.d/eloq6** locations instead.

To access a database in an additional instance, specify the instance-specific service name or port number in addition to the database name. Those utilities which do not expect a database name instead provide the **-h** command line option. Also, the EQ_DBSERVER environment variable can be set to specify the default instance.

The following examples assume that an additional instance was configured to use the port number 8201. They illustrate different ways to address the same instance:

```
schema -h :8201 db.schema.txt
dbcreate :8201/db
```

```
export EQ_DBSERVER=:8201
schema db.schema.txt
dbcreate db
```

Default eloqdb6.cfg file

```
# eloqdb6.cfg
# @(#) $Revision: B.07.00.2 $
#
# This file defines the eloqdb6 configuration and the database
# environment.
# The default location depends on the operating system:
#
#   HP-UX: /etc/opt/eloquence6/eloqdb6.cfg
#   Linux: /etc/opt/eloquence6/eloqdb6.cfg
#
# This file is read once at eloqdb6 startup.
#
# Format:
#
# The section names are not case sensitive. String values can be
# enclosed in double quotes to protect leading or trailing spaces.
# Everything after a hash (#) character is considered a comment.


### Server configuration

[Server]

# Title         If set, a server title is displayed by the ps
#               program in the eloqdb6 command line instead of the
#               default "eloqdb6: active" If you are using multiple
#              eloqdb6 server processes on a single system this can
#               be used to distinguish between different server
#               instances. The default value is empty.

#Title =

# Service       The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for requests. The default value is eloqdb.
#
# ServiceHttp   The service name (as defined in /etc/services)
#               or the port number where the server should listen
#               for HTTP requests. If this is not specified, the
#               HTTP status is disabled.
#

#Service = eloqdb
#ServiceHttp =
```

```
# panic         This option defines what should happen if a fatal
#               error is encountered.
#
#               The following options are valid:
#               restart Restart the server process (default)
#               exit    Terminate the process.
#              dump    Terminate the process and create a core dump.
#
#             panic = dump is a problem tracking option. Unless you
#             know what you need the coredump for you probably want
#               to stay with panic = exit or panic = restart

#panic = restart

# UID           The name (or numeric id) of the system account to
#               run client processes as when started as root.
# GID           The name (or numeric id) of the system group to run
#               client processes as when started as root.
#
#               Please note, that the server will refuse to start
#               as root unless UID and GID are valid.

UID = eloqdb
GID = eloqdb

# EnableIPC     If set, shared memory can be used to transmit data
#              between the database server and a client running on
#               the same system. This provides better performance
#               than using sockets because data are not passed
#               through the kernel.
#               The default value is 0 (disabled).
#
#               The following options are supported:
#
#               EnableIPC = 0 (default)
#                 Disables use of shared memory communication.
#
#               EnableIPC = 1
#                 Enables use of shared memory communication.
#                This mode uses a separate memory segment for each
#                 connection.
#
#               EnableIPC = 2 (recommended)
#                 Enables use of shared memory communication.
#                 This mode uses a common memory segment for all
#                 connections.
#
#               When setting EnableIPC configuration of kernel
#               parameters may be required.
```

```
#EnableIPC = 0

# SyncMode      If set, this causes the eloqdb6 server to operate in
#                sync write mode. The sync write mode is more
#                resistent against operating system and hardware
#                failures. When sync mode is disabled (set to 0) the
#                eloqdb6 uses the faster async  write strategy which
#                performs fewer disk writes but could lead to a
#                damaged database environment in case of a system
#                failure.
#                The default value is 1 (sync write mode enabled).

#SyncMode = 1

# LogFile       This defines where log messages are written to.
#               This configuration value either specifies a path/
#               file or one of the keywords below:
#
#               console  - log messages are written to the console
#               syslog   - log messages will be sent to the
#                          syslog daemon
#
#               The default value is "syslog".

#LogFile = syslog

# SysIdent      When logging to the syslog daemon, you can define
#               a syslog identifier. The default is eloqdb.
#               See syslogd(1M) for more information
#
# SysFacility   When logging to the syslog daemon, you can define
#               a syslog facility (USER/DAEMON/LOCAL0..LOCAL7)
#               The default setting is "USER".
#               See syslogd(1M) for more information

#SysIdent = eloqdb
#SysFacility = USER

# LogFlags      Each log message has an associated origin and
#               severity.
#              The log flags define, which messages will be logged.
#             The "*" origin matches all message origins, so it can
#               be used to setup a default which can be overriden
#               for a specific message origin (eg. "*1N0"):
#               Default LogFlags are "*0"
#
#               The following origin are in use:
#                  * = All origins
#                  A = Configuration subsystem
#                  X = Network transport
```

```
#                  P = Protocol handling
#                  T = Thread kernel
#                  I = IMAGE subsystem
#                  B = BTREE subsystem
#                  F = FIXREC subsystem
#                  V = Volume handling
#                  L = Transaction logging
#                  C = Page cache
#                  N = Node handling
#                  D = The server framework
#                  O = System catalog
#
#               The following severities are in use:
#                  L_ERROR  = 0   - error messages
#                  L_INFO   = 1   - information
#                  L_DEBUG  = 2   - debug
#                  L_VDEBUG = 3   - verbose debug
#
#               When using syslog, the following priorities
#               are mapped:
#                  L_ERROR  = LOG_ERR
#                  L_INFO   = LOG_NOTICE
#                  L_DEBUG  = LOG_DEBUG
#                  L_VDEBUG = LOG_DEBUG
#
#               Enabling log messages with L_DEBUG or L_VDEBUG
#              severity may result in a huge number of log messages.
#              To suppress anything but fatal messages, you can set
#               LogFlags to "*0". To enable informational log
#               messages LogFlags should be set to "*1".

#LogFlags = *0

# HTTPUser      The eloqdb6 server is able to display status
#              information by supporting the HTTP protocol (you can
#              use Netscape to monitor the database server process,
#               see ServiceHttp above).
#              If set, the eloqdb6 HTTP status display will require
#               a matching user name (HTTP basic authentification)
#               before allowing access to the eloqdb6 HTTP status.
#               The default value is empty.
#
# HTTPPswd     If set, the eloqdb6 HTTP status display will require
#               a matching password (HTTP basic authentification)
#               before allowing access to the eloqdb6 HTTP status.
#               The default value is empty.

#HttpUser =
#HttpPswd =
```

```
# HTTPFrame     If set, no link information is output on the HTTP
#               status display. So the status page could be used in
#               a web frame.
#               Default value is 0.

#HttpFrame = 0


### Data base configuration

[Config]

# Threads       Number of threads in the data base server.
#               A separate thread is required for each client.
#               Default number of threads is 40.

#Threads = 40

# IOThreads     This specifies the number of I/O threads which are
#               used by eloqdb6. eloqdb6 uses either separate
#               processes or kernel threads to perform overlapped
#               I/O operations. Please note that the eloqdb6 I/O
#               threads are visible with the ps command.
#               The default value is 4. A zero value disables usage
#               of IO threads.

#IOThreads = 4

# LockConflictingItems              If set, predicate locks with
#               conflicting items are granted, however any write
#               attempt to data where another process owns a lock
#               will result in a status error -12.
#              Former Eloquence revisions rejected a predicate lock
#               with a conflicting item, because this could lead to
#               a situation where two processes own a lock on an
#               overlapping subset of data. The default value is 0.

#LockConflictingItems = 0

# AllowSecondaryBlockingLock        If set, secondary blocking
#               locks are allowed. In previous Eloquence versions,
#               secondary locks in a blocking mode (odd modes)
#               failed with database status -135 ("Second lock is
#               not allowed in modes 1,3,5,11,13 and 15.") instead
#               of blocking. Current Eloquence versions return the
#              status code -35 in case a deadlock situation caused
#             by a secondary blocking lock is detected. Therefore,
#               this setting is enabled by default. To retain the
#               behavior of previous Eloquence versions it can be
#               set to 0. The default value is 1.
```

```
#AllowSecondaryBlockingLock = 1

# BufferCache   Size of page cache in megabytes. The page cache is
#               used to reduce the number of disc accesses. Large
#               cache size will speed up random database access,
#               while a too small cache size may cause bad server
#               performance.
#               Default cache size is 5 MB.

#BufferCache = 5

# The server performs a checkpoint operation at fixed intervals.
# This flushes all modified buffers (including metadata) to the
# disk and resets the log of committed transactions. A checkpoint
# is a point where the server knows all data are in a consistent
# state. Any data modification since the last checkpoint is
# recorded in the log volume.
#
# CheckPtFreq   Checkpoint frequency in seconds.
#               Default checkpoint frequency is 60 seconds.
#
# CheckPtSize   Checkpoint frequency based on accumulated log space
#              which would be freed by a checkpoint (in megabytes).
#               A zero CheckPtSize value disables size based
#               checkpoints.
#               Default checkpoint size is 10 megabytes.
#
# The database server performs a checkpoint operation at a fixed
# interval and optionally in addition when the accumulated log
# space which could be freed by a checkpoint operation reaches a
# given threshold.
# The frequency of the checkpoint operations has a great influence
# on the size of the log volume since the log volume must hold all
# committed transactions since between checkpoints

#CheckPtFreq = 60
#CheckPtSize = 10

# The syncer thread flushes modified buffer pages to the disk when
# they are likely to become reused in the near future.
#
# SyncerFreq    Syncer thread invocation frequency (in seconds)
#               Default interval is 5 seconds.

#SyncerFreq = 5

# SyncerJournalFlushInterval        If SyncMode is enabled this
#               configuration item specifies the interval (in
#               milliseconds) at which the journal of committed
```

```
#               transactions is synchronized to disk.
#               In case of an operating system or hardware failure
#              transactions that were not synchronized to disk are
#               typically lost.
#              A smaller value reduces the amount of transactions
#               that might be lost in case of a system crash.
#               However, setting this value too low significantly
#               impacts write performance.
#               Setting this value to 0 reverts to the legacy
#               SyncMode behavior where every transaction is
#               immediately synchronized.
#               The default value is 500 milliseconds.

#SyncerJournalFlushInterval = 500


### Store/Restore Devices

[Devices]

# This section defines the "server devices" which can be used with
# dbstore and dbrestore. Each entry consists of the device name and
# an associated path.
#
# A "server device" could either be a single file, a directory or a
# device. When no server devices are configured, dbstore and
# dbrestore operation is refused by the server.
#
# The example below defines two server devices. The device "Tape"
# points to a tape device file, the device "Backup" points to a
# directory which is intended to hold the backup files.

#Tape = /dev/rmt/c1t0d0BEST
#Backup = /data/backup


### Forward log

[ForwardLog]

# FwLog         Configures the file, device or pipe to be used for
#              forward-logging. Using the %N token in the file name
#               activates automatic file management (not possible
#               for devices or pipes).
#               By default, forward-logging is inactive.
#
#               The examples below configure an automatically
#               managed file and a pipe which compresses the data
#               on-the-fly:
```

```
#FwLog = /mnt/disk2/data/db-forward-%N.log
#FwLog = |gzip -c >/mnt/disk2/data/db-forward.log.gz

# FwRecovery    Configures the file, device or pipe to be used
#              during forward recovery. If not set, the Log setting
#               is used by default.
#
#               The example below configures a pipe which
#               uncompresses the data on-the-fly:

#FwRecovery = |gzip -dc /mnt/disk2/data/db-forward.log.gz

# FwOnFailure   Configures the action to be taken in case the
#               forward-log cannot be written, e.g. due to
#               insufficient disk space.
#               Possible values are disable or panic. If set to
#               disable, forward-logging will be disabled on
#               failure. As soon as the problem is solved it can be
#               manually enabled using dbctl.
#               If set to panic, the eloqdb6 server will issue a
#               panic and abort itself.
#               The default value is disable.

#FwOnFailure = disable

# FwMaxSize     Limits the maximum size of automatically managed
#               forward-log files (in megabytes). If not set or set
#               to zero, the file size limit is 2 gigabytes. The
#               default value is 0 (not set).

#FwMaxSize = 0


### Data base environment

[Volumes]

# List of data base volumes. Initially empty.
# This is usually filled in by dbvolcreate and dbvolextend
# utilities
```

Customizing the Eloquence Configuration Files

This discussion assumes that the Eloquence software has already been installed on your system. The information in this section is directed to the system administrator for the Eloquence software.

NOTE: The configuration steps mentioned here are not related to Eloquence database operation. If the only Eloquence component you intend to use is the database, you can skip this section.

Before Eloquence can be used, its resources must be configured. Eloquence programs usually don't use system resources directly, instead they rely on a mapping of pathes, printers and device files in Eloquence configuration files.

There are three different levels of configuration:

System global — This is achived with the **`eloq.config`** configuration file which is located in the Eloquence configuration directory.

Group specific — This is achived with the **`group.<GroupName>`** configuation file which is located in the Eloquence directory.

User specific — This is achived with the **`.eloqrc`** configuration file which is located in the home directory of the user.

The Eloquence configuration files are read by the eloqcore process, when it is started. The configuration files are processed in an order such that more specific definitions override the more general ones. So a system global assignment can be overriden from a group specific configuration file, a user specific definition will override group and system global definitions.

The system global configuration file, **`eloq.config`** is usually copied during the installation process to Eloquence configuration directory and should be adapted to local requirements. Template configuration files are provided in the directory **`/opt/eloquence6/newconfig/config`**. The template configuration files provide comlete inline documentation and are included at the end of this section for your reference.

Eloquence resource configuration

Eloquence resources go back to the "dark ages" when a precedessor of Eloquence was implemented in hardware (called HP250/HP260 at that time) and the resources definition actually were real OS resources. Since programs depended on a program independent resource configuration and it a convenient mechanism

anyway, the concept was kept. Instead of real devices Eloquence resources can be mapped to whatever is appropriate. Eloquence is of course able to access native operating resources directly.

Since the following names are not commonly used, let's define them first:

VOLUME A **VOLUME** is the Eloquence concept of a directory. Instead of using the path directly, it is possible to assign an identifier for a path and refer to it in a symbolic manner.

MSI This is a short form of **MASS STORAGE IS** and species the default **VOLUME** on which pathes should be related unless an absolute path or another **VOLUME** is given.

PRINTER A **PRINTER** is the Eloquence concept of an output depvice. A PRINTER is identified by a number and could be mapped to a device file or to a sequence of commands.

The device numbers 8 to 10 have a special predefined meaning:

8: Display terminal.
9: Bit bucket (Eloquence equivalent of /dev/null)
10: Local terminal printer

PORT A PORT is the Eloquence equivalent of a (tty) device file. Eloquence provides prowefull machanisms to handle them in a efficient manner.

The eloq.config configration file

The eloq.config file provides system global definitions and is usually copied during the installation process into the Eloquence configuration directory from the template file **`d.eloq.config`**.

The group specific configuration file

To provide group specific definitions, you could install a group specific configuration file in the Eloquence configuration directory. Consider we would like to have a specific configuration for the *sales* group you would perform the following steps:

1 Change to the Eloquence confuration directory:

```
cd /etc/opt/eloquence6
```

2 Create a group specifc configuration file from the template group configuration file. The group specific file should be named group.*sales*.

```
cp /opt/eloquence6/newconfig/config/d.group group.sales
```

3 Use a text editor, such as vi to edit the file

```
vi group.sales
```

The user specific configuration file

To provide user specific definitions, you could install a user specific configuration file in the gome directory of the user. Consider we would like to have a specific configuration for the user *mike*, you would perform the following steps:

1 Change to the home directory of the user:

```
cd ~mike
```

2 Create a user specifc configuration file from the template user configuration file. The user specific file should be named .eloqrc:

```
cp /opt/eloquence6/newconfig/config/d.eloqrc .eloqrc
```

3 Use a text editor, such as vi to edit the file

```
vi .eloqrc
```

Template eloq.config file

```
# d.eloq.config
# Eloquence configuration file
# (C) Copyright Marxmeier Software AG, 2002
# @(#)$Revision: 20.4 $
#
# This file contains global available configuration
# It must be named eloq.config and located at:
#   HP-UX 9.x    /opt/eloquence6/etc
#   HP-UX 10.x   /etc/opt/eloquence6
#   linux        /etc/opt/eloquence6
#
#  PLEASE NOTE:
#  You MUST define at least one volume (typically SYSTEM, see below),
#  or eloqcore will fail on startup.
#
# Globally defined volumes
#
# Format: VOLUME label [device] path
#
#      label   - Volume label (up to 8 characters)
#                must be unique per file
#      device  - Device specifier eg. ":F2,6,0"
#                ignored when present, no longer used
#      path    - HP-UX path to map volume on
#
# Globally defined printers
#
# Format: PRINTER no [model] type spec
#
#      no     - printer select code (-2 .. 7, 11 .. 99)
#      model  - PCL or OTHER
#                ignored when present, not used
#      type   - printer type PIPE, FILE or SYSTEM
#      spec   - path/command to process on printer selection
#
# Globally defined ports
#
# Format: PORT no spec
#
#       no    - port select code (11 .. 20)
#               may not conflict with PRINTER
#       spec  - path of tty devicefile
#
#
# Default date/time format
#
# Format: DATE spec
#         TIME spec
#
#      spec    - date/time specification. please refer to date(1) or
#                   strftime(3) for more information.
#                For backward compatibility, the former specifications
#                   "DD.MM.YY" and "MM/DD/YY" are silently converted.
#
#                   Default: DATE "%m/%d/%y"
```

```
#                           TIME "%H:%M:%S"
#
# Global MSI value
#
# Format: MSI label
#
#      label  - Volume label. Default is the first defined volume.
#

# --- sample volumes

VOLUME SYSTEM  /opt/eloquence6/share/prog
#VOLUMEEXAMPLE /opt/eloquence6/share/example

# --- sample printers

PRINTER 0  PIPE "lp -s 2>/dev/null"
#PRINTER 1  PIPE "lp -s -oc 2>/dev/null"
#PRINTER 2  FILE /dev/lp

# --- sample PORT

#PORT 11 /dev/tty0p5
```

Configuration of the GUI Dialog Server

The GUI dialog server allows Eloquence programs to apply a graphical user interface. The process about how the Eloquence program (i.e. the **`eloqcore`** process) connects to the Run and Dialog servers is configured in the **`eloq.ini`** configuration file.

Customization of the eloq.ini file

The **eloq.ini** configuration file is located in the /**etc/opt/eloquence6** directory.

The installation program does not overwrite an existing configuration file but installs the **/opt/eloquence6/newconfig/config/eloq.ini** file instead.

The ‘ini‘ File format

The **`eloq.ini`** and **`eloqcl.ini`** files contain several sections each containing a group of related configuration items. The sections and configuration items have the following format:

```
[Section]
Item=Value
```

Section is the name of a section. The enclosing brackets ([ ]) are required and they must start at the first column. *Item=Value* defines a value of a configuration item. *Item* is the name of a configuration item. It consists of any sequence of characters (case insensitive) and digits followed by an assignment operator (=). Depending on the item type, the value may either be numeric or alphanumeric (optionally enclosed in double quotes). Comment lines must start with a semicolon (;) or a hash character (#) in the first column.

The eloq.ini file

Section [runsrv]

This section specifies the defaults used with the **`runsrv`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: runsrv

Function: The default service name or port number to be used with the **`runsrv`** connection protocol. If no value is defined the port associated with the service runsrv is used.

Section [eloqdlg]

This section specifies the defaults used with the **`dlg`** connection protocol. The following configuration items are supported:

Service

Service= *service name or port number*

Default: 8011

Function: The default service name or port number to be used with the **`dlg`** connection protocol. If no value is defined the port number 8011 is used.

Mapping Driver Specifications

Other sections in the **eloq.ini** file can be used to map a driver specification to a different value. When a driver is specified (e.g. in the **`DLG SET ".driver"`** statement or with the **`eloqcore -dlg`** command line option) the argument is looked-up in the **eloq.ini** configuration file after any protocol specification has been removed. If a section has been found it is used to replace the original argument.

The following configuration items are supported:

dlg

dlg= *replacement text*

Function: When contacting a DLG driver this is used as a replacement for the specified driver. The value can specify a different protocol.

host

host= *[hostname][:service]*

Function: When using the Run server to execute a remote operation or to start the Dialog server the host variable can be used to specify a different hostname (or IP address) and/or a different service name (or port number) which should be used instead.

Example

```
[runsrv]
Service = runsrv

[eloqdlg]
Service = 8011

[mike]
dlg = dlg://lxmike

[chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```

This example specifies to use a port number for the **runsrv** protocol which is associated with the service runsrv (as defined in the local SERVICES file). For the **dlg** protocol the port number 8011 should be used.

When a driver argument "mike" is specified (e.g. "**@mike**", "**dlg://mike**" or "**runsrv://mike**") the value "**dlg://lxmike**" is used instead. When a driver argument "chris" is specified it is replaced by "**runsrv://wserv:8765**". When the Run server is contacted from either the command line (using the **runclnt** utility) or the **RunSrv.DLL** and a remote name "chris" is passed, the value "**wserv:8765**" is used instead.

Running multiple runsrv instances on a single system

This is required if the Citrix Metaframe or the Windows Terminal Server product is used. Each user needs to start the RUNSRV32 with a different port number (probably during autostart). Then a driver name containing the host name and the user name is passed and translated by using the mapping defined in **eloq.ini**.

Suppose the user "chris" has a RUNSRV32 using port 8765 when working on system wserv, then the following mapping section should be added in **eloq.ini**:

```
[wserv_chris]
dlg = runsrv://wserv:8765
host = wserv:8765
```