Introduction to the eloquence database

Outline

- Product overview
- Database concepts
- TurboIMAGE compatibility
- Installation and configuration
- Database utilities

Product Overview

- Eloquence is a product of Marxmeier Software AG, Germany
- Support is available worldwide through Marxmeier Software and its partners
- Eloquence is available on the HP-UX, Linux and Windows platform

Product Overview

About 3000+ installations worldwide

Used by about 60+ VARs / ISVs worldwide

Covers a wide range of installations

- Single user Laptop to 1500+ concurrent users with 50+ GB database size
- Terminal based server solution to distributed GUI environment

Product Overview

Eloquence is typically used to implement vertical and custom specific solutions

Solutions based on Eloquence include

- ERP, Order Management
- Financial Accounting / Payroll
- Civil Services
- Bank, Financial services
- …

product history

- Eloquence was first released in 1989 as a migration solution to move HP250/HP260 applications to HP-UX
- Since then Eloquence has gone through continuous development

Product Components

Due to its history the Eloquence product consists of multiple components

- Eloquence programming language (based on HP Business Basic)
- Eloquence database (based on IMAGE)
- TurboIMAGE compatibility option
- Graphical User Interface
- Development Environment
- ODBC driver (separate product)

Product Options

- An Eloquence license is required for each server system
- The “unlimited license” option (AH0) provides an unlimited user license for a single system
- The “entry license” option (000) and additional user licenses allow competitive pricing for entry level configurations
- The corresponding TurboIMAGE compatibility option (3k) must be ordered in addition

Obtaining Eloquence

The Eloquence product is available for download from the Eloquence web site

CD-ROM media may be ordered as an alternative

Product Evaluation

The “Personal Edition” provides a free two user license with a database up to 50 MB

A temporary license provides an unlimited version of Eloquence for evaluation purposes

A temporary license key can be requested from the Eloquence web site
http://www.marxmeier.com/eloquence/license

More Information

Detailed information is available on the Eloquence web site:
http://www.marxmeier.com/eloquence

TurboIMAGE migration:
http://www.marxmeier.com/eloquence/hp3k

B.07.10 Documentation:
http://www.marxmeier.com/eloquence/support/B0710

Get in contact:
info@marxmeier.com

Database architecture

Overview

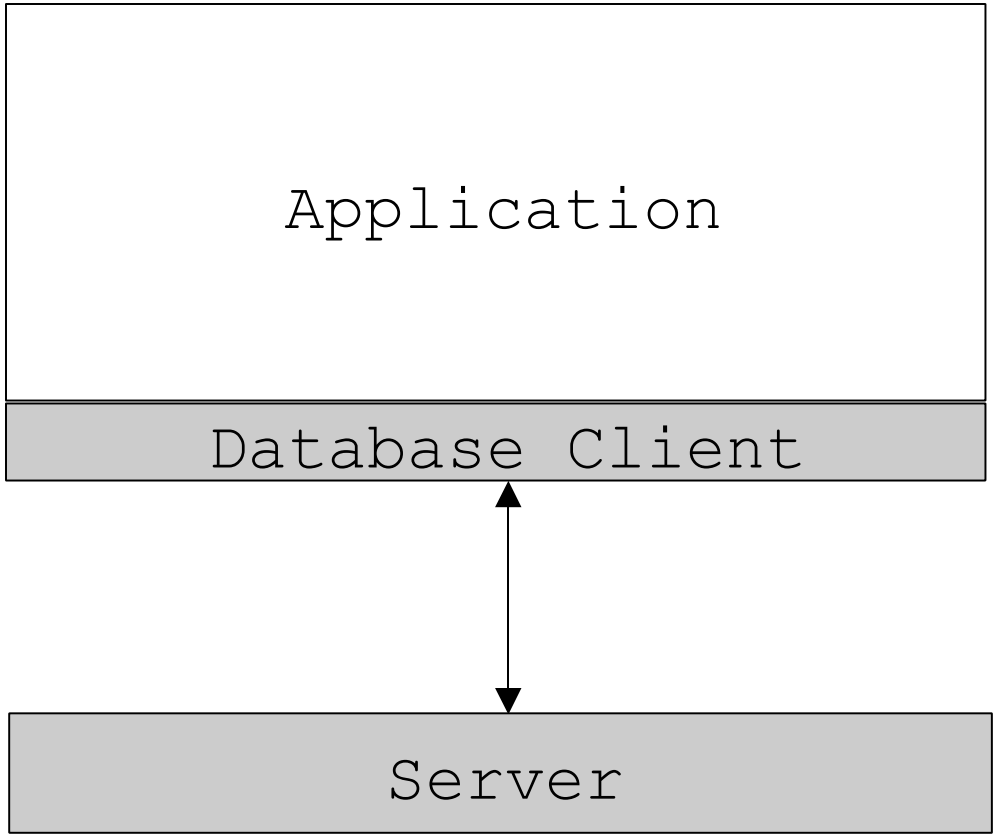
- Client/server architecture
- Indexes
- Locking
- Transactions
- Database names
- Database security
- Database environment

Client-Server Architecture

The Eloquence database uses a client/server based architecture.

- Database access and management is performed by a server process
- The application is linked with the database API
- The server is connected through the network (or shared memory)

Client-Server Architecture



Multiple Platforms

Eloquence is available for multiple operating systems and architectures

- HP-UX on PA-RISC and Itanium
- Linux on Intel IA32 (Itanium and AMD64 in beta test)
- Windows NT/2000/XP/2003 on Intel IA32

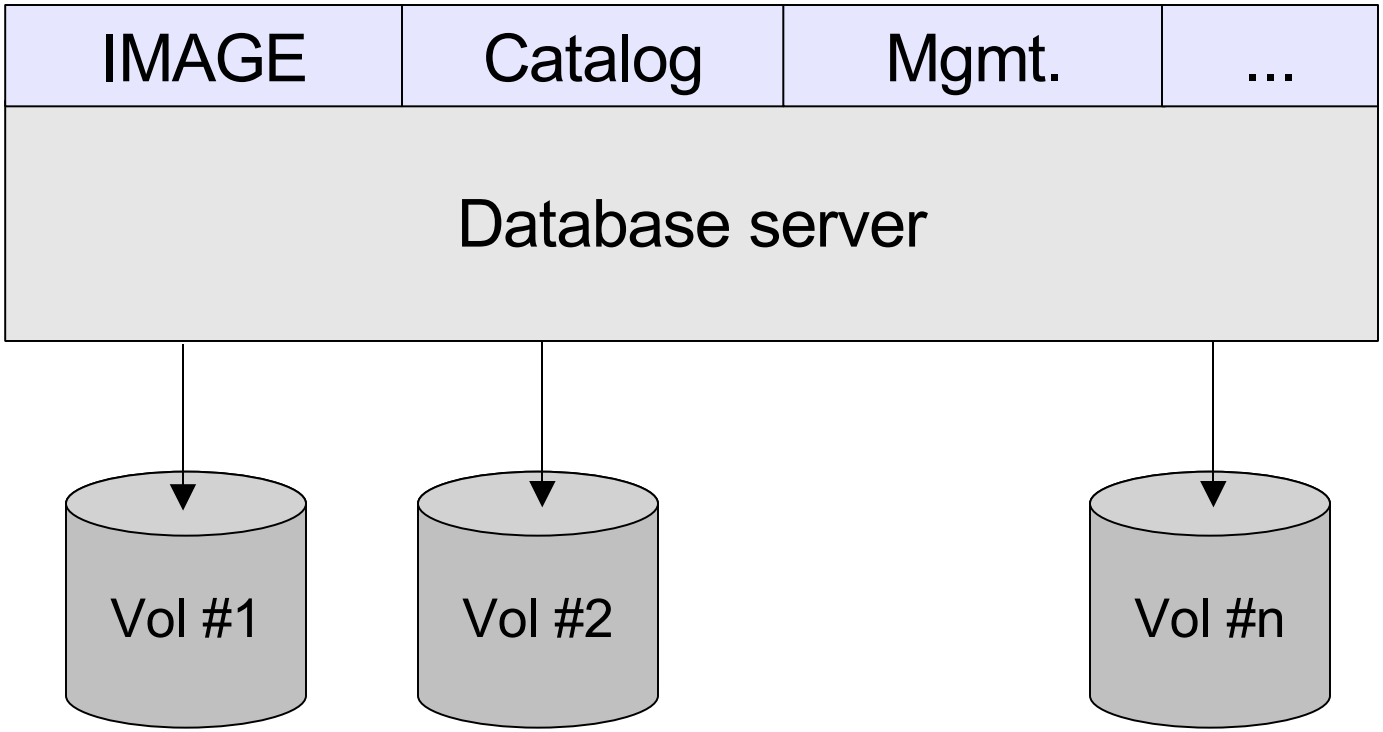
Network Transparent

Applications running on different machines and operating systems can access a database

Requests and results are translated transparently

- Character set encoding
- Byte order conversion

Database Server



TurboIMAGE compatibility

TurboIMAGE compatibility is implemented at different levels

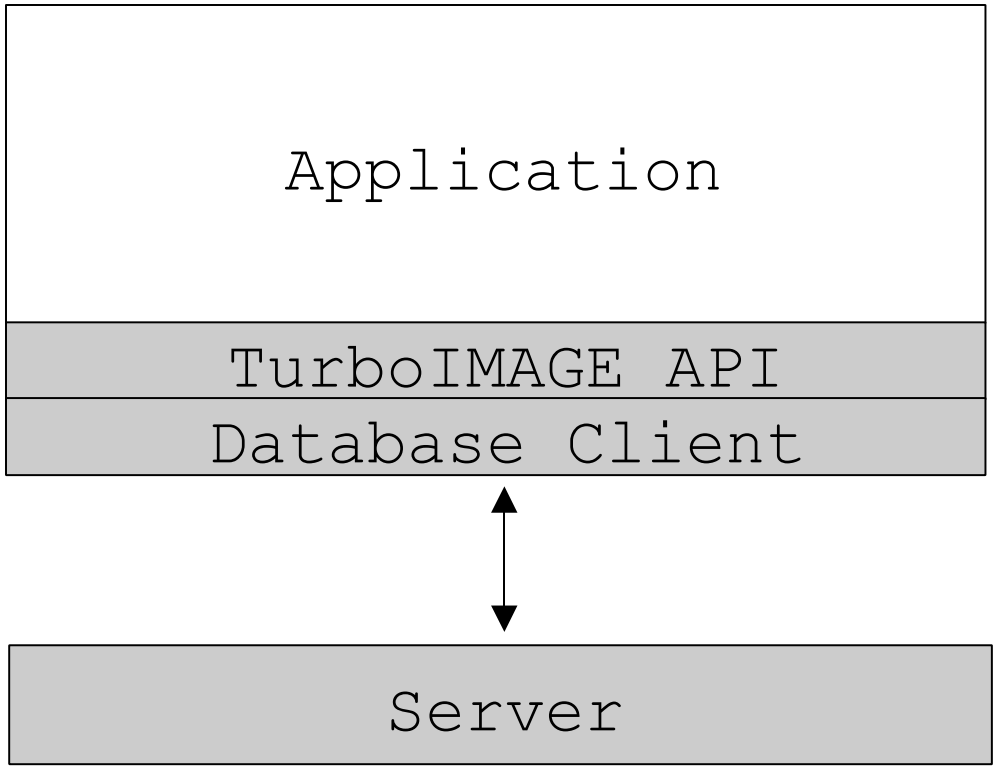
- The database server implements functionality at the backend
- The database client and utilities provide support for TurboIMAGE functionality
- The TurboIMAGE compatibility API implements source code compatibility

TurboIMAGE compatibility

Compatibility goes beyond intrinsic calls and also includes performance profile

- Applications are built on assumptions and take advantage of specific behaviour
- An IMAGE application often expects that
  - a DBFIND or DBGET execute fast
  - DBGET performance does not differ substantially between modes
- Application may depend on external utilities or third party tools

TurboIMAGE compatibility



Indexing

Eloquence comes with integrated indexing

- Uses indexes instead of hashing with master sets
- Supports both the TPI programming interface as well as IMAGE indexes (“super chains”)

Indexing / TPI

Eloquence provides a commonly used subset of the TPI functionality

Advanced indexing functions are currently not available

- keywords
- relational access
- independent indexing

Indexing vs. Hashing

Hashing is often more efficient but less flexible

- Migrating secondaries require specific locking on master sets
- Applications must be aware of migrating secondaries
- Growing a master set causes hashing to become less efficient

Indexing vs. Hashing

Caching is used to improve index performance

Additional functionality is available at no extra overhead

- partial key retrieval
- ordered retrieval
- wildcard lookup

Dynamic Dataset Expansion

- Capacity is no longer used
- Data sets grow dynamically when required
- Capacity value returned from DBINFO is highest record number ever used

Locking

- Locking is fully compatible with TurboIMAGE
- Eloquence does not impose a locking strategy
- Write operations do not require a previous lock. If a conflicting lock is granted, a status is returned

Locking

Read locking and selective unlocking are available

- DBLOCK modes 11 to 16 implement read locking at the database, data set or item level
- DBUNLOCK allows to release a specific lock

Locking

- Multiple DBLOCKs are allowed
- If configured, locks on conflicting items are accepted
- Deadlock conditions are detected and a status is returned

Transactions

- All databases are part of a transaction
- Uncommitted changes are not visible to other processes
- Transactions are not limited in size
- Nested transactions are supported
- Committed transactions are saved in a journal in the log volume
- Committed transactions are persistent, only roll forward recovery is supported

Database Names

- A database name is no longer restricted to 6 characters
- Databases do not reside in the file system but are managed through a server process
- A database name addresses a specific server instead of a file location

Database Names

[[hostname][:service]/]database

- Hostname specifies database server system
- Service specifies database server instance

For example:

```
localhost:eloqdb/SAMPLEDB
:eloqdb/SAMPLEDB
SAMPLEDB
```

All specify the same database.

Database Names

The EQ_DBSERVER environment variable can be used to specify a default server

Syntax

[hostname][:service]

- Hostname specifies database server system
- Service specifies database server instance

Example

```
EQ_DBSERVER=server:eloqdb2
```

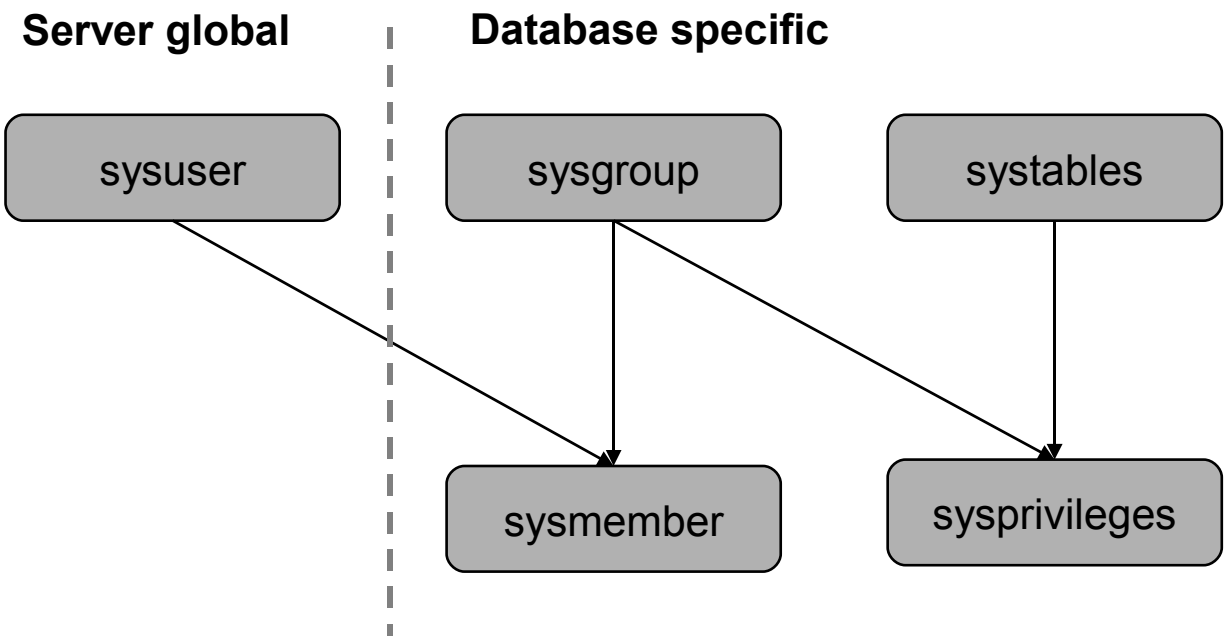
Database Security

The database server maintains a list of users

Security groups specific to a database

- Similar to TurboIMAGE user classes
- Database privileges are assigned to groups
- A user can be a member of multiple groups

Database Security



Database Security

A file can be specified which contains the user and password

The EQ_DBUSER environment variable can be used to specify a default user

The EQ_DBPASSWORD environment variable can be used to specify a default password

E*xample*

```
EQ_DBUSER=”mike”
EQ_DBPASSWORD="file:/home/mike/db/pswd”
```

Database Security

The new DBLOGON procedure can be used to specify user and password

With the TurboIMAGE compatibility API the DBOPEN password argument may be used to specify a user and password

The EQ_DBUSER and EQ_DBPASSWORD environment variables may specify a default user and password

Database Environment

- A database environment is an organizational entity
- It can manage any number of databases
- Database names must be unique
- Databases share common resources
- Multiple database environments can coexist on the same system

Database Environment

A database environment consists of

- a configuration file
- a primary data volume
- a transaction log volume

Multiple database environments can coexist on the same machine, each managed by a separate server process

Database Environment

The configuration file defines

- the server configuration
- scaling and tuning parameters
- the location of the volume files

Volumes

- Volume files are a storage container managed by the database server
- A maximum of 255 volume files are supported in a server environment
- The maximum size of a single volume file is 128 GB
- Volume files are specific to the system architecture (byte order)

Volumes

- A volume file is internally maintained in 8 KB blocks
- The first block in a volume contains additional volume information
- All volume files in an environment share the same serial number and generation
- The order in the server configuration file is not significant

Volumes

Different volume types

- Primary data volume
- Transaction log volume
- Additional data volume(s)

Volumes

The primary data volume contains the server catalog and the database contents

Additional data volumes can be added to extend the primary data volume

Volumes

A log volume contains

- a journal of recently committed transactions
- information on incomplete transactions
- temporary disk space during online backup

The transaction journal is used in case of an unexpected server abort to perform a recovery of recent transactions

The transaction log is truncated when it reaches a configured size

Server Catalog

Eloquence does not use a root file

Structural information is maintained in the database environment

The server catalog is initialized with the dbvolcreate utility and maintained with the schema and dbutil utilities

Server Catalog

The server catalog contains global and database specific information

Global catalog information

- list of users and databases

Database specific information

- schema and security information

Online Backup

In online backup mode, the data volumes are frozen

Modifications during online backup are temporarily saved into the transaction log volume

Any backup software can be used to create a consistent backup

Online Backup

The dbctl utility is used to control online backup mode

Example backup script

```
$ dbctl -u file:/root/credentials backup start
$ tar -cf /dev/rmt/0m /database
$ dbctl -u file:/root/credentials backup stop
```

Forward Logging

Forward logging is used to record all modifications since a previous backup

Forward recovery is fast and involves only minimal processing

The forward log files can be managed automatically by the server process

Database Limits

Eloquence Image limits (schema limits)

- 2048 data items
- 500 data sets
- 64 / 16 paths
- Entry length 5120 bytes

Scalability

Database / data set size is limited by the disk space allocated to the database environment

- Current limit is ~32 TB

Max. number of concurrent users per database environment is currently limited to 1000 (4000 on HP-UX)

TurboIMAGE compatibility

- The TurboIMAGE compatibility adds the TurboIMAGE intrinsics to the Eloquence database
- All TurboIMAGE intrinsics and almost all modes are supported and behave identical
- HP e3000 applications can usually be ported with only minor changes

Schema processor

The Eloquence schema processor accepts HP e3000 schema files

The -T option specifies TurboIMAGE compatibility

Item types J and R are not supported natively and are only visible from the TurboIMAGE compatibility mode

- J is mapped to I
- R is mapped to E (IEEE format)

Installation / Configuration

Obtaining the product

- Download from the Eloquence web site
  http://www.marxmeier.com/eloquence/download
- Order a CD-ROM
- Check the Eloquence web site for recent patches
  http://www.marxmeier.com/eloquence/support/

Supported HP-UX releases

Eloquence is available for the PA-RISC (HP9000) and Itanium (Integrity) architecture

- PA-RISC based systems:
  - HP-UX 11i (11.11)
  - HP-UX 11iv2 (11.23)
- Itanium based systems:
  - HP-UX 11iv2 (11.23)

Installation on HP-UX

- Eloquence is available as depot file and is installed with swinstall
- Different builds of the Eloquence product archive are available
  - **`Eloquence-B0710-*.depot`**
    PA-RISC 1.1, compatible with all PA-RISC systems
  - **`Eloquence-B0710-*-pa20.depot`**
    PA-RISC 2.0, requires a PA8xxx CPU
  - **`Eloquence-B0710-*-ia64.depot`**
    Itanium based systems

Installation on HP-UX

The download version of the product archive must be uncompressed before installation

```
# gzip –d Eloquence-B0710-*.depot.gz
```

Run “swinstall” to install the product

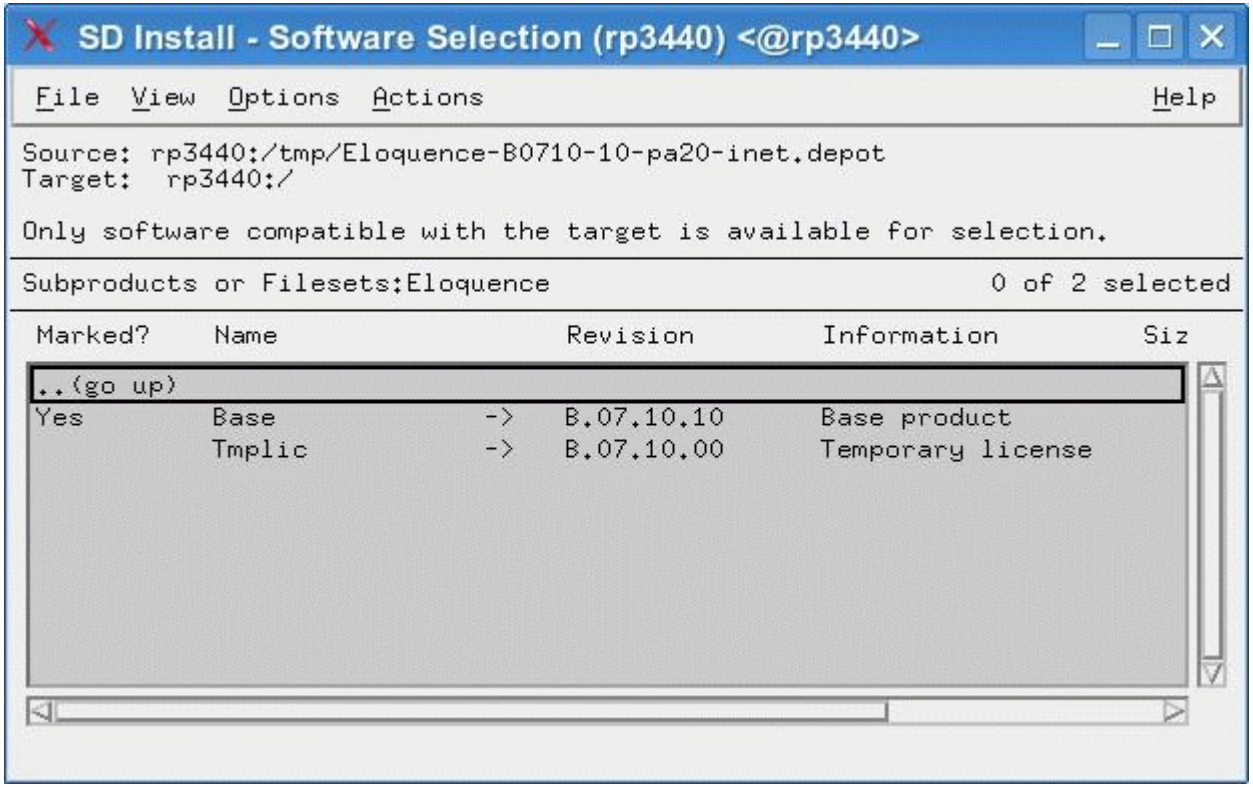
/usr/sbin/swinstall –s /path/to/Eloquence-B0710-*.depot

- the -s option specifies the absolute path of the product archive

Installation on HP-UX

- Mark the product for installation to install the default options
- Optionally select the product and select Actions -> Open Item to view the product options, then mark product options
- Select Actions -> Install to continue the installation
- By default a 30 days temporary license is installed the first time Eloquence is installed on a system

Installation on HP-UX



- swinstall with Eloquence product options

Required HP-UX patches

Required HP-UX patches are documented on the Eloquence web site.

HP-UX patches can be obtained from the hp itrc site

- http://europe.itrc.hp.com (Europe)
- http://us.itrc.hp.com (US, Asia-Pacific)

Required HP-UX 11.23 patches:

- PHKL_32542 (libsyscall-pdk.a cumulative patch)
- PHCO_31575 (libxcurses cumulative patch)

Required HP-UX 11.11 patches:

- PHCO_30778 (libxcurses cumulative patch)
- PHKL_28489 (erroneous EFAULT returned)

Install new License Key

- By default the “Personal Edition” license key is installed
- A temporary license key may be created during installation
- A temporary license key may be requested from the Eloquence web site
- Add supplied license keys to the Eloquence license file /etc/opt/eloquence6/license
- Comment out any existing license keys
- Use the chklic utility to very the license file /opt/eloquence6/etc/chklic

Configuration of TCP Services

Installation adds the default Eloquence services in the services file.

- On HP-UX and Linux the /etc/services file is used to define TCP service names
- On Windows the services file is located in the directory C:\Windows\System32\Drivers\etc

Make sure the eloqdb service name is defined in the services file

```
eloqdb   8102/tcp          # Eloquence database server
…
```

Create eloqdb user/group

- A separate user id and group are used with the database environment
  - configured in the eloqdb6.cfg
  - owner of the database volume files
  - account can be used for volume administration
- Create a user name and a group name, for example eloqdb
- On Windows the system account is used by default

HP-UX Kernel Parameters

HP-UX kernel parameters may need to be configured

- semaphores related parameters
- shared memory related parameters
- data size
- memory windows

HP-UX Kernel Parameters

Semaphores related kernel parameters

- ‘semmni’ : number of connections + 2
- ‘semmap’ : set to ‘semmni’ + 2
- ‘semmns’ : number of connections + max. number i/o threads
- ‘semmnu’ : number of servers / system
- ‘semume’ : max. number of connections

/usr/sbin/kmtune may be used to obtain current kernel configuration parameters

HP-UX Kernel Parameters

- Shared memory related parameters (only if EnableIPC configuration item is set to 1)
  - ‘shmmni’ : number of connections
  - ‘shmseg’ : number of connections
- Max. data size
  - set ‘maxdsiz’ to at least 0x08000000 (128MB)
- Enable memory windows
  - set ‘max_mem_window’ to number of servers / system + 2

Setup the Database Environment

- Server configuration file (eloqdb6.cfg)
- Create the database volume files
  - Primary data volume
  - Transaction log volume
  - Additional data volume(s)
- Start the database server

Server configuration file

This file defines server properties

- configuration
- scaling and tuning parameters
- volume files

Default server configuration file is /etc/opt/eloquence6/eloqdb6.cfg

Example Server Configuration

```
[Server]
Service = eloqdb
ServiceHTTP = 8103
UID = eloqdb
GID = eloqdb
EnableIPC = 2
SyncMode = 1
```

Example Server Configuration

```
[Config]
Threads = 100
IOThreads = 4
BufferCache = 128
CheckPtSize = 50
```

Create the Volume Files

Required

```
# dbvolcreate /var/opt/eloquence6/data01.vol
# dbvolextend –t log /var/opt/eloquence6/log.vol
```

Optional

```
# dbvolextend –t data /var/opt/eloquence6/data02.vol
# dbvolextend –t data /var/opt/eloquence6/data03.vol
…
```

HP-UX startup configuration

Modify the startup configuration file
/etc/rc.config.d/eloquence6

```
START_ELOQSD=0
START_ELOQDB6=1
```

This configures the default database environment

Start the Database Server

Start the default database server

/sbin/init.d/eloq6 start

Check if the server is active

sbin/init.d/eloq6 status

Stop the default database server

/sbin/init.d/eloq6 stop

Troubleshooting

The database server writes diagnostic messages to the syslog (default) or the configured log file

- HP-UX: /var/adm/syslog/syslog.log
- Linux: /var/log/messages
- Windows: application event log

The LogFlags configuration item defines the log level

- increase the log level to informational messages by setting LogFlags = *1 or LogFlags = *1E2 in the server configuration file

Starting, stopping as non-root

Starting or stopping the database requires root access

As of B.07.10 the database may be started and stopped without requiring root access if enabled in the startup configuration file /etc/rc.config.d/eloquence6

```
START_STOP_AS_ROOT=0
```

This enables the configured user for a database environment to start/stop this environment

Additional database environments

All database environments that are managed by the start/stop scripts needs to be configured in the startup configuration file /etc/rc.config.d/eloquence6

- Only a single database environment is configured by default
- Additional database environments are recognized by either the configured id (eg. “production”, “test”) or the service name (“eloqdb”)

Additional database environments

- ELOQDB6_CFG[0] = <config file>
  - Configuration file used with this database instance (required).
  - This can be an absolute path or relative to /etc/opt/eloquence6
- ELOQDB6_ID[0] = <instance id>
  - 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_SERVICE[0] = <service_or_port>
  - Specifies the service name or port number to use (optional). If present, overrides the configuration file setting.

Additional database environments

- ELOQDB6_ARGS[0]=""
  - Additional command line arguments for this instance (optional).
- ELOQDB6_START[0]={0|1}
  - Specifies if the Eloquence startup/shutdown script should make use of this entry (optional). Defaults to 1.
  - 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. On HP-UX this may be used with setmemwindow -n to define a memory window used by the database server instance.

Additional database environments

Example startup configuration file

```
ELOQDB6_CFG[0]=eloqdb6.cfg
#ELOQDB6_START[0]=1
#ELOQDB6_ARGS[0]=""
#ELOQDB6_SERVICE[0]=""
#ELOQDB6_ID[0]=""
ELOQDB6_RUNPFX[0]="setmemwindow -n"
ELOQDB6_CFG[1]=eloqdb6_2.cfg
ELOQDB6_ID[1]=test
ELOQDB6_RUNPFX[1]="setmemwindow -n"
ELOQDB6_CFG[2]=eloqdb6_3.cfg
ELOQDB6_ID[2]=test3
ELOQDB6_RUNPFX[2]="setmemwindow -n"
ELOQDB6_START[2]=0
```

Additional database environments

The start/stop script allows specifying a list of database environments

- If present, only the specified servers are processed
- By default all servers are processed (possibly limited by user id if not root)

```
# sbin/init.d/eloq6 status eloqdb
eloqdb6[eloqdb] process is active (pid 12155)

# sbin/init.d/eloq6 status
eloqdb6[eloqdb] process is active (pid 12155)
eloqdb6[test] process is active (pid 12183)
eloqdb6[test3] process is not active
```

Server config file

- Title = “server title”
  - If set, a server title is displayed by the ps program in the eloqdb6 command line or the HTTP status.
- Service = eloqdb
  - The service name (as defined in /etc/services) or the port number the server should listen for requests. The default value is eloqdb.
- ServiceHttp = 8103
  - The service name (as defined in /etc/services) or the port number the server should listen for HTTP requests. If this is not specified, the HTTP status is disabled.

Server config file

- panic = exit
  - This option defines what should happen if a fatal error is encountered. Options are
    - restart — Restart the server process. This is the default.
    - exit — Terminate the process.
    - dump — Terminate the process and create a core dump
- UID = eloqdb
  - The name (or numeric id) of the system account to run client processes as when started as root. Owner of the database files.
- GID = eloqdb
  - The name (or numeric id) of the system group to run client processes as when started as root. Group of the database files.

Server config file

- EnableIPC = 2
  - 0 = Disables use of shared memory communication.
  - 1 = Enables use of shared memory communication, uses separate memory segment for each connection
  - 2 = Enables use of shared memory communication, uses a common memory segment for all connections (default)
  - EnableIPC=1 on HP-UX causes a performance problem with larger environments

Server config file

- SyncMode = 1
  - 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. The default value is 1 (sync write mode enabled).
  - [Config] SyncerJournalFlushInterval parameter defines flush interval

Server config file

- LogFile = /var/opt/eloquence6/log/eloqdb6.log
  - This defines where log messages are written to. Defaults to syslog.
- LogFlags = *1E2
  - Defines server log level
    - 0 = only critical messages
    - 1 = informational
    - 2 … = debug

Server config file

- Statfile = /var/opt/eloquence6/eloqdb6.usage
  - Specifies the filename the server uses to write statistics information. The target file must be writable for the eloqdb6 process.

Server config file

- [db-access]
  - {allow|deny} = {All | ip-address [/ [netmask | addrbits]]}
    - ip-address is a IP v4 address, such as 127.0.0.1.
    - netmask is a IP v4 network mask, such as 255.255.255.0.
    - addrbits specifies the number of bits in the IP address.
    - If neither netmask nor addrbits is specified it defaults to a host address.
  - Defaults to

```
[db-access]
allow = 127.0.0.1
deny = all
```

Server config file

- Threads = 1000
  - Number of threads in the data base server. A separate thread is required for each connection.
- IOThreads = 8
  - Number of I/O threads which are used by eloqdb6 to perform I/O operations (default = 4).
  - On HP-UX the eloqdb6 I/O threads are visible with the ps command.

Server config file

- BufferCache = 256
  - Size of db specific page cache in megabytes (limit is 1 GB)
  - The server makes use of the OS cache as a secondary cache
- CheckPtSize = 50
  - Checkpoint frequency based on accumulated size of transaction journal (in MB).

Server config file

- SyncerJournalFlushInterval = 500
  - 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.
- VolumeFileSizeLimit = 0
  - Specifies the max. volume size in MB. The value cannot exceed 128 GB (131072 MB).
  - In addition the following values are supported.
    -1 = ~ 2GB (backwards compatible)
    0 = 128 GB (unlimited)

Server config file

- FwLog = /data/fwlog/fw-%N.log
  - Specifies the fw log file. The %N token in the file is replaced by volume generation/sequence
- FwMaxSize = 0
  - 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 GB.
- EnableAudit = 1
  - Enables audit information to be added to the forward log in addition to recovery data. Desabled by default.

Installation on Linux

- Installing the product
- Kernel Parameter
- Configuring automatic server startup
- Start the database server

Linux Installation

Eloquence uses the RPM package manager for installation

- RedHat RHEL3, RHEL4 and SUSE SLES8, SLES9 are fully supported with the B.07.10 release
- Other Linux distributions may be used but additional manual configuration steps (or patches) may be required

Linux Installation

For installation or updating execute the command below

$ rpm -Uhv Eloquence-B0710.glibc2.2-*.i386.rpm

- Temporary license option is not available on Linux during installation

Linux Kernel Parameter

Linux uses similar kernel parameters as HP-UX

The default Linux kernel parameters are often sufficient for small to medium installations

Use ipcs -l to list current shared memory/semaphore limits

Kernel parameters can be changed by writing the new value to appropriate file in the /proc/sys/kernel/ directory

Linux Automatic server start

Modify the startup configuration file
/etc/sysconfig/eloquence6

START_ELOQSD=0
START_ELOQDB6=1

Start the database server

Start the default database server

- /etc/init.d/eloq6 start

Check if the server is active

- /etc/init.d/eloq6 status

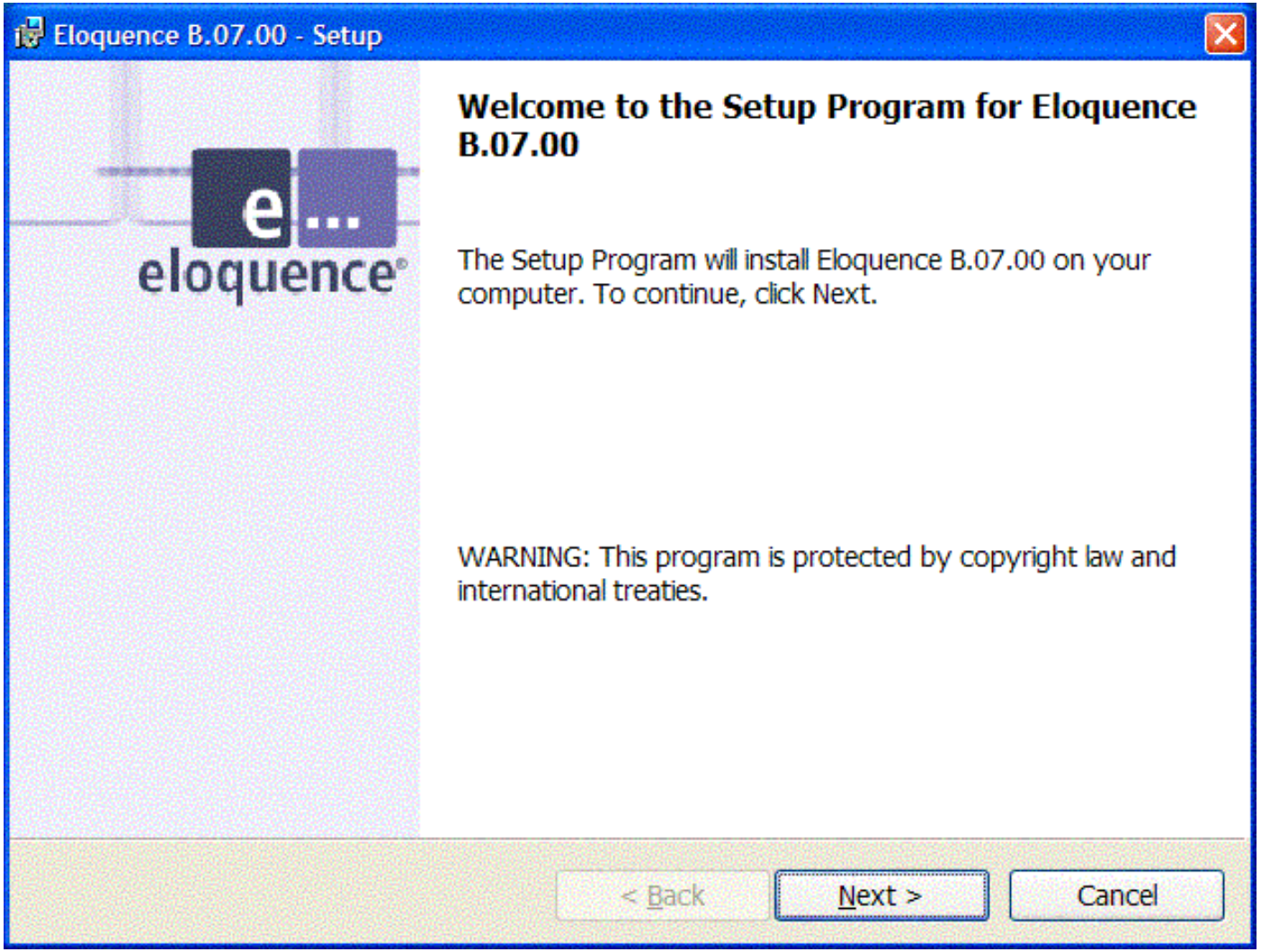
Stop the default database server

- /etc/init.d/eloq6 stop

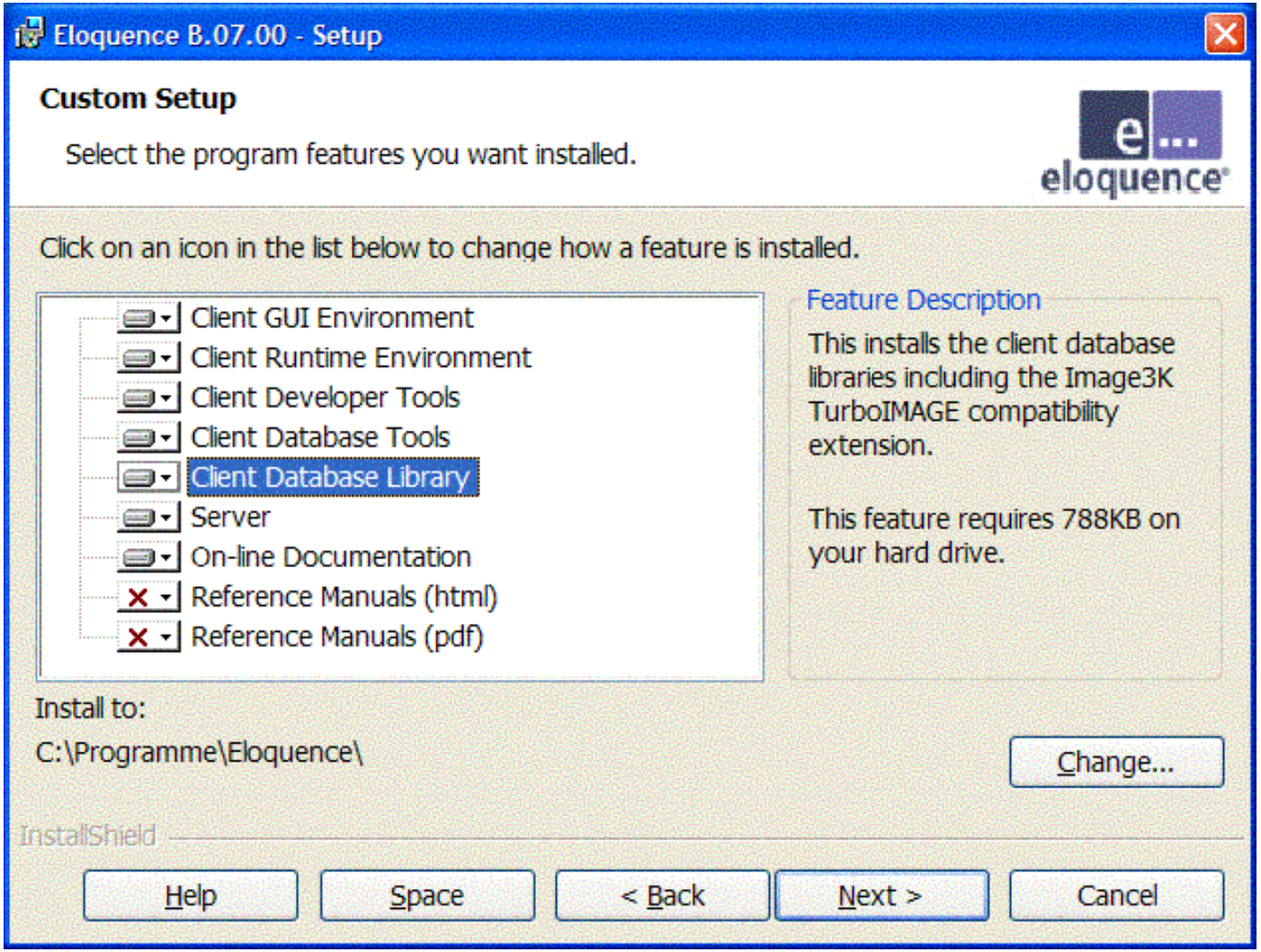
Installation on Windows

- Installing the product
- Configuring automatic startup for the database service
- Starting the database service

Windows Installation



Windows Installation



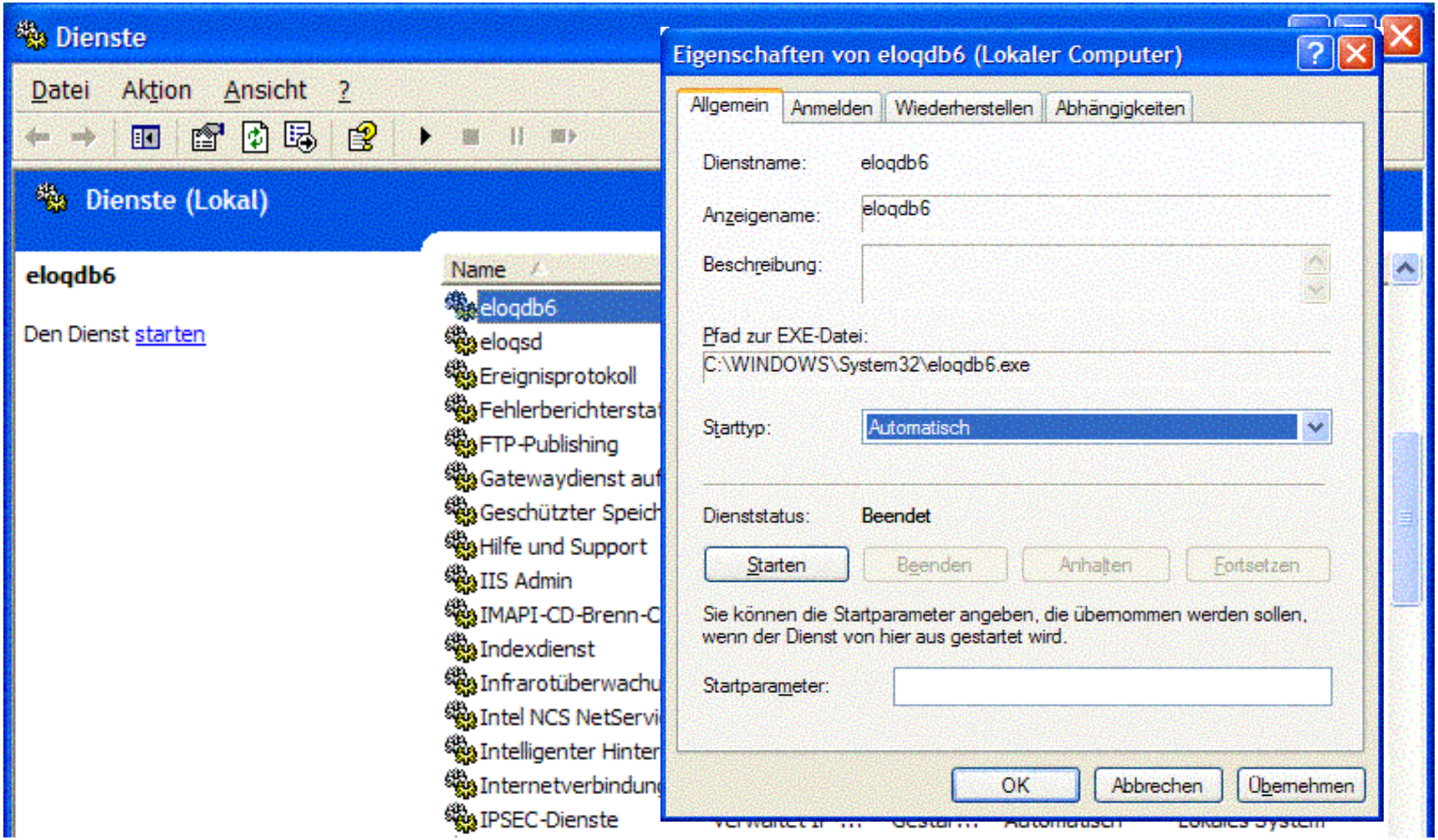
Windows Configuration

The installation program has registered the eloqdb6 service

Configure automatic start mode for the Eloquence database in the service control panel

Start the service manually for the first time

Automatic server start (Windows)



Windows Specifics

- On the Windows platform the database is implemented as a Windows service
- By default the system account is used to run the database
- On Windows, a separate service is configured for each database environment, there is no startup configuration file
- The services control manager is used to start/stop services. As an alternative, the “net start service” command may be used

Database utilities

database utilities

- Offline utilities directly access the volume files. Typically used when the database server process is not active
- Administrative utilities to control server operation or request server status information
- Database utilities to use the database server

offline utilities

- dbvolcreate, dbvolextend, dbvolchange, dblogreset
  - database volume management
- dbvoldump
  - display volume properties
- dbrecover
  - forward recovery
- dbfsck, dbcfix
  - volume consistency check and simple repair tool
- dbstore utility
  - backup single database

dbvolcreate

The dbvolcreate utility is used to create a new server instance. It creates the primary data volume and initializes the server catalog

The volume file is added to the server configuration file

dbvolcreate /db/d01.vol

Options:

-c config_file	*server config file*
-s initial_size	*initial size of the volume file*
-m max_size	*max. volume file size*
-e extension_size	*volume extension size*

dbvolextend

The dbvolextend utility extends a database environment by a volume file (data or log volume)

The volume file is added to the server configuration file

dbvolextend -t log /db/log.vol

Another use is to re-create missing log volume files

dbvolextend -R

Options:

-c config_file	*server config file*
-t volume_type	*the volume type (DATA or LOG)*
-s initial_size	*initial size of the volume file*
-m max_size	*max. volume file size*
-e extension_size	*volume extension size*

dbvolchange

The dbvolchange utility is used to change properties of a volume file

dbvolchange -m 0 /db/d01.vol

Options:

-c config_file	*server config file*
-m max_size	*max. volume file size (0 = max.)*
-e extension_size	*volume extension size (0 = fixed size)*

dblogreset

The dblogreset utility is used to reset the transaction log volume to minimal size

dblogreset processes any information in the transaction log volume before truncating the file

dblogreset

Options:

-c config_file	*server config file*

dbvoldump

The dbvoldump utility is used to display volume properties

dbvoldump

Options:

-c config_file	*server config file*

- dbvoldump *may be used when the server process is active*

dbvoldump utility

```
$ dbvoldump

ID   Type Path
1    DATA /data/db/db01.vol
2    LOG  /data/db/db02.vol
3    DATA /data/db/db03.vol
4    DATA /data/db/db04.vol
5    DATA /data/db/db05.vol
6    DATA /data/db/db06.vol

ID   Type  Cur.Sz  Ext.Sz  Max.Sz    Free    Used
1    DATA  1024.0     1.0  1024.0     0.0  1024.0
2    LOG    356.5     1.0     0.0   356.4     0.1
3    DATA  1024.0     1.0  1024.0     0.0  1024.0
4    DATA  1024.0     1.0  1024.0     0.0  1024.0
5    DATA  2000.0     1.0  2000.0     0.0  2000.0
6    DATA   642.5     1.0  1024.0     3.4   639.1
```

dbrecover

The dbrecover utility is used to perform forward recovery

dbrecover

Options:

-c config_file	*server config file*

- Requires a previous backup and archived transactions since the backup (FwLog)

dbfsck

The dbfsck utility is used to check volume consistency and perform simple repairs.

dbfsck does not verify IMAGE integrity

dbfsck

Options:

-c config_file	*server config file*
-w	*enable write mode*
-a	*automatically fix minor inconsistencies*

- dbfsck in read-only mode *may be used when the server process is in on-line backup mode*

dbcfix

The dbcfix utility can be used to verify IMAGE chain linkage consistency and perform repairs.

As dbcfix operates at the database level, a database name needs to be specified.

dbcfix –v –l log sample

Options:

-c config_file	*server config file*
-u user_id	*user name*
-p password	*password for the user*
-w	*enable write mode*
-l file	*specify log file*
-v	*show progress*

- dbcfix in read-only mode *may be used when the server process is in on-line backup mode*

administrative utilities

- dbctl
  - server management utility
- HTTP status monitor
  - access server status information with a web browser

dbctl utility

The dbctl utility is used to submit administrative requests to the database server or obtain status information

The dbctl utility provides the communication only. The functionality is implemented by the server process

```
$ dbctl help
$ dbctl -u dba shutdown
$ dbctl -u dba backup start
$ dbctl -h lxsrv:8302 list thread
```

Options:

-u user_id	*user name*
-p password	*password for the user*
-h host	*host and/or service*

dbctl commands

help [command]

- returns the list of available commands or command syntax

forwardlog {ENALE|DISABLE|RESTART|STATUS}

- shutdown the server process

backup {START|STOP}

- start or stop online backup mode

dbstore database target

- store a single database to a file, device or process

dbrestore [/info] source [dbase_name]

- restore a previously stored database

dbctl commands (cont)

list [session|db|lock|thread|dbopen]

- list active sessions, open databases, locks and internal threads

cancelthread {tid}

- unblock a session which is waiting for a lock

killthread {tid}

- terminate a session

logfile {logfile}

- change the log file of the server process

logflags {logflags}

- change the logging options of the server process

HTTP status

The HTTP status monitor is a simple web server built into the database server

A web browser can be used to check the database server status

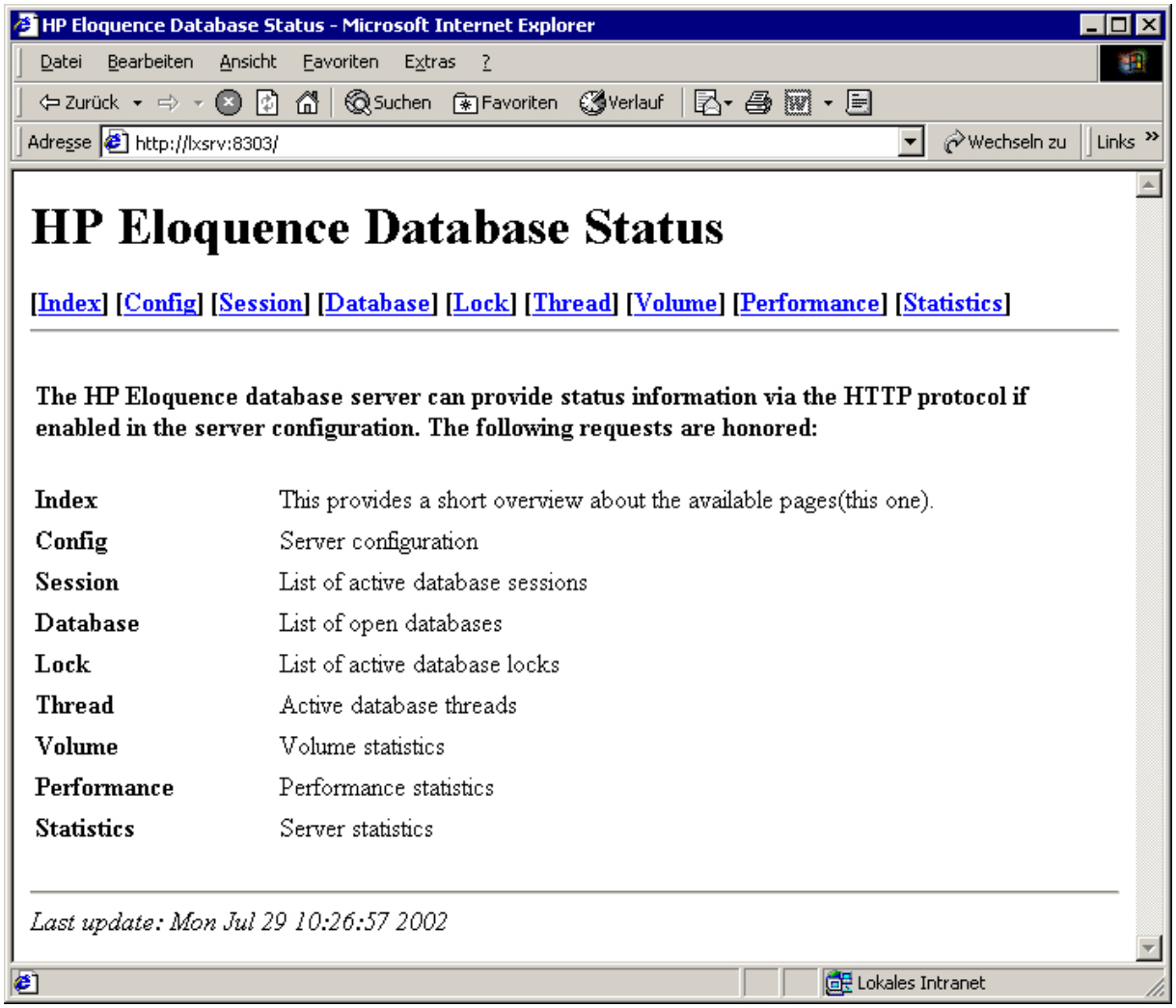
Enter the address in your web browser http://server:8103/ where 8103 is the port number configured for HttpService

The dbctl utility can be used to retrieve similar information from a script file

HTTP status

- Server configuration
- Active database sessions
- Opened databases
- Database locks
- Status of the server threads
- Volume file usage and status
- Performance information
- Internal statistics

HTTP status (index)



HTTP status (config)

eloquence

Server Status

Item	Value
Server revision	@(#)HP-B1368B A.06.31 HP ELOQUENCE / Linux (Jul 7 2001)
Server process id	32008
Semaphore id	40962
Installed licenses	Permanent, *unlimited*

Server Configuration

Configuration Item	Setting
panic	restart
service	eloqdb
serviceHTTP	8103
UseKeepAlive	Yes
NoDNS	No

HTTP status (session)

eloquence

Active Sessions

Available number of user licenses: unlimited
Number of active sessions: 3

Session	TID	Host	IP ADDR	Platform	User	Login
0x827d6d0	8	lxmike.marxmeier.com	194.64.71.106:2309	Linux	mike	public
0x82498c8	9	localhost	127.0.0.1:2822	Linux	mike	public
0x82552d0	10	localhost	127.0.0.1:2817	Linux	rolf	public

HTTP status (database)

eloquence

Open Databases

Database	OpenCount	WriteShared	WriteExclusive	ReadShared	ReadExclusive
AHDB	1	1	0	0	0
VHDB	1	1	0	0	0
BHDB	1	1	0	0	0
SAMPLE	1	1	0	0	0
PPS5C	1	1	0	0	0
KALK5	1	1	0	0	0
ZEWI5	1	1	0	0	0

HTTP status (lock)

Database Locks

Session	Database	DBID	Status	Mode	Qualifier
0x827d6d0	SAMPLE	1	GRANTED	6	expression
set=1 item=@ (effective set lock)					
0x825a440	SAMPLE	1	BLOCKED	4	set 1

HTTP status (threads)

eloquence

Active Threads

Number of active user threads: 4

ST	TID	WCHAN	Blocked	Stack	Avail	Sched	Name
R	1			120	6872	862908	idle
S	2		262552	152	16196	1	shutdown
S	3		262552	128	16220	1	ps
S	4		2	128	15120	53192	syncer
R	5			208	13536	67	http
S	6		113	128	16168	8233	tio monitor
S	7		52	140	7780	4434	checkpoint
S	8		49	140	7100	29	lxmike.marxmeier.com
S	9		197	140	7744	9	localhost
S	10		266	140	7744	12	localhost
W	11	0x827cd44	3	444	7744	5	localhost
file lockdb.c, line 320 waiting for 8							

HTTP status (volume)

eloquence

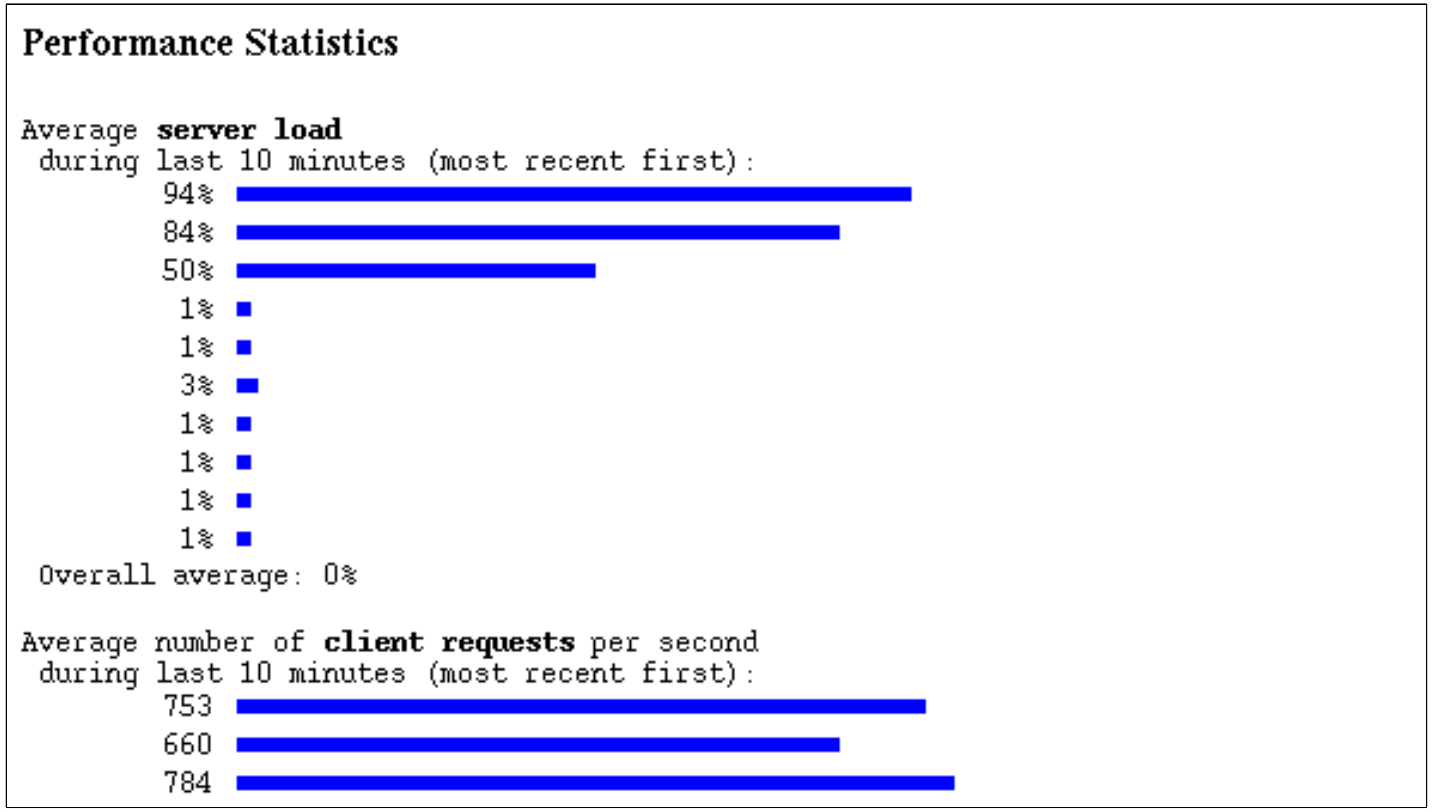
Server Volumes

ID	Type	Path
1	DATA	/data3/dalex/db/db01.vol
2	LOG	/data3/dalex/db/db02.vol
3	DATA	/data3/dalex/db/db03.vol
4	DATA	/data3/dalex/db/db04.vol
5	DATA	/data3/dalex/db/db05.vol
6	DATA	/data3/dalex/db/db06.vol
7	DATA	/data3/dalex/db/db07.vol

Volume Statistics

ID	Type	Cur.Sz	Ext.Sz	Max.Sz	Free	Used	Seek Cnt	Read Cnt	Write Cnt
1	DATA	1024.0	1.0	1024.0	0.0	1024.0	285	3251	277
2	LOG	48.5	1.0	0.0	48.4	0.1	27777	14	30279
3	DATA	1024.0	1.0	1024.0	0.0	1024.0	9	1237	6
4	DATA	1024.0	1.0	1024.0	0.0	1024.0	9	418	6
5	DATA	1024.0	1.0	1024.0	0.0	1024.0	34	1188	28
6	DATA	1024.0	1.0	1024.0	0.0	1024.0	7	1215	6
7	DATA	1024.0	1.0	0.0	931.3	92.7	2433	853	2568

HTTP status (performance)



HTTP status (statistics)

eloquence

Transaction Statistics

```
Number of top-level transactions: 123
Number of sub transactions: 1655
Number of transactions committed: 1778
Number of transactions rolled-back: 0
Number of transaction records in use: 0 (0 bytes)
 (maximum: 111 records, 4440 bytes)
```

Page Cache Statistics

```
18688130 page puts, 18683190 page gets, 15135 page new
6658 page allocs, 3242 page flushes
100% cache hit rate (18674713 hits, 8477 misses)
75% page hash efficiency (14381 primary, 4680 synonym)
8477 page reads, 30465 page writes
```

BTREE Statistics

```
0 key cache hits, 0 misses
6 page splits (4 root splits, 0 sort splits)
Key optimization reduced 0 bytes on node pages.
```

database utilities

- schema
- dbcreate, dberase, dbpurge
  - create / erase / purge a database
- dbtables
  - database structure information
- prschema
  - create schema file from database
- dbdumpcat
  - catalog information utility

database utilities (cont)

- dbexport, dbimport
  - export/import data base content to/from text file
- dbinfo
  - information on database tables
- dbutil
  - structural maintenance and database security
- QUERY utility
- fwaudit utility

schema processor

The schema processor creates the database from a schema file

$ schema -T schemafile

- The Eloquence schema processor understands both Eloquence and TurboIMAGE schema files
- Option -T selects TurboIMAGE compatibility mode
- $CONTROL options and the database name can be specified on the command line

schema processor (options)

```
Usage: schema [options] file

options:
 -help       - show usage (this list)
 -u user     - set user name
 -p pswd     - set password
 -h host     - host and service to contact
 -b name     - specify database name
 -l          - output source listing              (LIST)
 -n          - no root file, only check syntax    (NOROOT)
 -t          - output set table                   (TABLE)
 -e cnt      - Abort after cnt error messages     (ERRORS=)
 -T          - HP3000 TurboImage compatibility mode
 -W width    - Limit line length to width
 -L          - Add line number to source listing
```

dbcreate / dberase / dbpurge

The dbcreate, dberase and dbpurge utilities are used to create, erase or purge a database

When erasing a detail set (but not the associated master set) the path information in the master set is reset

```
$ dbcreate database
$ dberase database [set ...]
$ dbpurge database
```

Options:

-u user_id	*user name*
-p password	*password for the user*

prschema utility

The prschema utility is used to create a schema file from a database

If option -T is specified the schema file uses TurboIMAGE conventions where possible

$ prschema database

Options:

-u user_id	*user name*
-p password	*password for the user*
-T	*TurboIMAGE compatibility*

dbtables utility

The dbtables utility is used to output information on the database structure (items, data sets, paths, entries and indexes)

$ dbtables database

Options:

-u user_id	*user name*
-p password	*password for the user*
-l lines	*set page length*
-s	*short listing (data sets only)*
-T	*TurboIMAGE compatibility*

dbtables utility (cont)

eloquence

D A T A S E T F I E L D S

Data Set Name	Set Num	Item Name	Item Num	Item Type	Start Byte	Item Cnt	Item Len	Item Role
CUSTOMERS	1	CUSTNO	1	X6	0	1	6	Srch
		NAME1	3	X32	6	1	32	
		NAME2	4	X32	38	1	32	
		NAME3	5	X32	70	1	32	
		MATCHCODE	2	X10	102	1	10	
		STREET	6	X32	112	1	32	
		ZIPCITY	7	X32	144	1	32	
		PHONE	8	X18	176	1	18	
		TURNOVER	9	E4	194	3	24	
		SALESAREA	10	X6	218	1	6	
		IMATCHCODE	34			1	10	Indx
		MATCHCODE	2	X	0		10	Seg

dbdumpcat utility

The dbdumpcat utility returns information from the server catalog

- Unless a database name is specified, dbdumpcat returns server specific information
- If a database name is present, database specific information is returned
- The option -l provides a list of available catalog tables
- The -t <id> option allows to specify to return information from a specific table only. The id can be obtained from the output return by the -l option.
- Option –n causes simplified output (scripting)

dbdumpcat utility (cont)

```
$ dbdumpcat -l
-------------------------------------------------
| Server catalog                                |
| localhost:eloqdb                              |
|-----------------------------------------------|
| Id    | Name                          | Count |
|-----------------------------------------------|
|    20 | sysobjects                    |   869 |
|    21 | sysdevices                    |     2 |
|    22 | sysvat                        |     0 |
|    30 | sysuser                       |     2 |
|    31 | sysdb                         |     7 |
|    32 | systables                     |    10 |
|    33 | syscolumns                    |    65 |
|    34 | sysindex                      |     0 |
|    35 | sysindexseg                   |     0 |
|    36 | syscollate                    |     0 |
-------------------------------------------------
```

dbdumpcat utility (cont)

```
$ dbdumpcat -t 31
-----------------------------------------
#31 sysdb (49 entries)
-----------------------------------------
|dbid|name                  |flags   |nodeid|
-----------------------------------------
|12  |PPS5C                 |04000000|128   |
|15  |PLAN5                 |04000000|576   |
|18  |ZEWI5                 |04000000|630   |
|21  |KALK5                 |04000000|846   |
|28  |SAMPLE                |04000000|1414  |
|75  |RUECKM                |04000000|2661  |
|82  |LVS5C                 |04000000|2794  |
|86  |SHDB                  |04000000|2879  |
|87  |ATES2                 |04000000|3168  |
...
```

dbdumpcat utility (cont)

```
$ dbdumpcat -l sample
-------------------------------------------------
| Database catalog                              |
| sample                                        |
|-----------------------------------------------|
| Id    | Name                          | Count |
|-----------------------------------------------|
|     1 | systables                     |    17 |
|     2 | sysgroup                      |     3 |
|     3 | syscolumns                    |    88 |
|     4 | sysindex                      |    10 |
|     5 | sysmember                     |     2 |
|     6 | sysprivilege                  |     8 |
|     7 | sysindexseg                   |    11 |
|     8 | systablecolumn                |    40 |
|     9 | systableindex                 |     7 |
|    10 | systablepath                  |     8 |
|    11 | sysdbproperty                 |     5 |
-------------------------------------------------
```

dbexport utility

The dbexport utility is used to export the database content to text format

By default a separate file is created for each data set in the current directory named like DBNAME.###.exp

- The -o option allows to specify a different target directory
- The *-s file* option specifies a single file should be used which contains all data sets
- The *-c* option exports entries in chain order
- Any data set name(s) or number(s) specified after the database name indicates only the specified data sets should be exported

dbexport utility (options)

```
usage: dbexport [options] database [set ...]
options:
 -help   - show usage (this list)
 -u user - set user name
 -p pswd - set password
 -o path - set output directory (not single file)
 -v      - verbose output
 -c      - chained export
 -r      - create restructure information
 -s file - output into single file, '-' = stdout
 -f sep  - field separator, default is ','
 -z cset - set export code set (roman8, iso88591)
```

dbexport utility (cont)

```
$ dbexport -v TESTDB

Processing database : TESTDB
Export path         : .

DATA SET                 RECORDS   COUNT
---------------- --- - -------- --------
CUSTOMERS        001 M     1177     1177
PARTS            002 M      182      182
ID               003 A       47
ORDERS           004 D       47       47
LINEITEMS        005 D      136      136
```

dbimport utility

The dbimport utility is used to load the database from export files

By default dbimport expects the export files in the current directory with a name like DBNAME.###.exp

- The -i option allows to specify a different source directory
- The *-s file* option specifies a single file should be used which contains all data sets
- The *-v* option displays the progress
- Any data set name(s) or number(s) specified after the database name indicates only the specified data sets should be imported

dbimport utility (options)

```
usage: dbimport [options] database [set ...]

options:
 -help   - show usage (this list)
 -u user - set user name
 -p pswd - set password
 -i path - set import path
 -v      - verbose output
 -t      - trace item value assignment
 -r file - restructure database, '-' = no file
 -s file - import from single file, '-' = stdin
 -e file - log errors instead of aborting
 -f sep  - field separator, default is ','
 -z cset - set import code set (roman8, iso88591)
```

EXPORT file format

- Export files are text based
  - Each line contains a record
  - Item values are separated by comma
  - String items are enclosed in quotes

- Format documentation
  http://www.marxmeier.com/eloquence/support/misc/export.html

EXPORT file format (cont)

```
"24601",442,900126,"21089",5,"VK",1298.46
"24602",1120,880116,"25001",5,"VK",28073.01
"24603",1210,880125,"30010",5,"VK",1611.09
"24604",1258,880201,"13005",5,"VK",10508.16
"24605",1446,880227,"13007",11,"VK",0
"24606",1460,880227,"17007",7,"VK",1150.81
"24607",1462,880227,"17007",7,"VK",8300.82
"24608",2424,880704,"22002",5,"VK",3719.23
"24609",2612,880725,"22016",5,"VK",1396.02
"24610",2894,880907,"18012",5,"VK",14.56
"24611",3342,881027,"15017",5,"VK",808.33
...
```

dbinfo utility

dbinfo lists the data sets for the specified database

data set name, type, number of entries and capacity are output

$ dbinfo database [set ...]

Options:

-u user_id	*user name*
-p password	*password for the user*

dbinfo utility

```
$ dbinfo TESTDB
Processing database : TESTDB

SET NAME                  RECLEN CAPACITY ENTRIES
------------------------- ------ -------- --------
CUSTOMERS          001 M     112     1355     1177
PARTS              002 M      53      524      182
ID                 003 A       2     2259       47
ORDERS             004 D      21     1008       47
LINEITEMS          005 D      22     1008      136
```

dbutil utility

dbutil provides a central point for database administration

- security management
  - adding and deleting user accounts
  - granting access privileges
- structural maintenance
  - adding an item or an index to a database
  - changing an item type
  - adding a data set to a database

dbutil utility

usage: dbutil [options] [file]

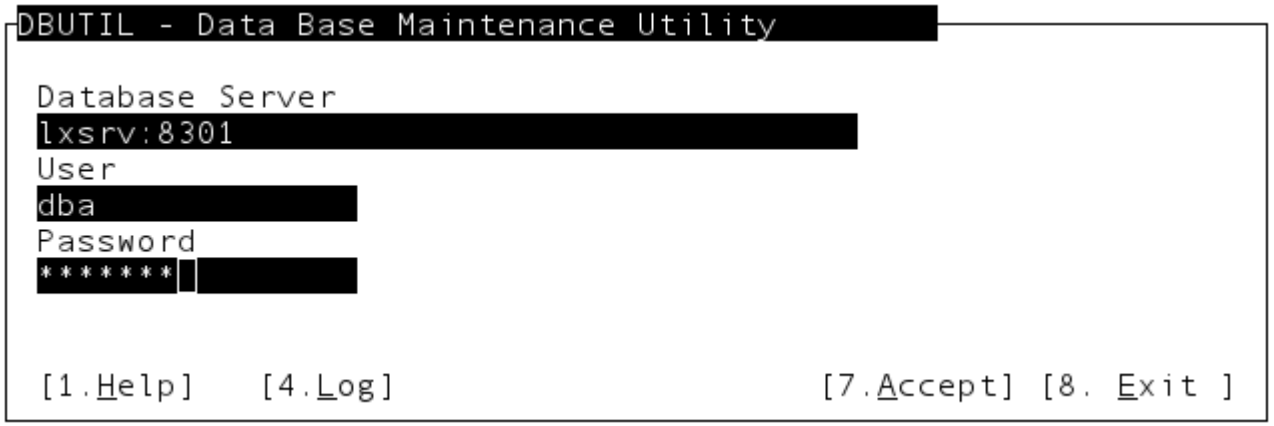
options:

-help	*= show usage*	*(this list)*
-n	*= check script*	*(batch mode only)*
-v	*= verbose*	*(batch mode only)*

If a file argument is present, dbutil will process in batch mode and execute any statements in the batch file.

If the -n option is present, no changes will be made to the database. Processing will end after checking the input file and analyzing the changes

dbutil utility




```
# connect to server
CONNECT "lxsrv:8302";
LOGON "dba" PASSWORD "file:/root/dba";

# create user mike with the password "secret"
# and allow connection to the database server
CREATE USER "mike" PASSWORD "secret";
GRANT CONNECT TO "mike";
```

QUERY utility

Eloquence comes with two different QUERY implementations:

- The HP260 based QUERY implementation
- Query3k is based original QUERY/iX ported from MPE

fwaudit utility

- The fwaudit utility is used to extract information from archived transactions or audit files
- Server must be configured to output audit information (EnableAudit=1 in server config file)
- fwaudit may create output in audit file format or text format
- Available audit information
  - OS, IP address, OS login, database login, process, additional application information
  - Database schema, old and new record image

fwaudit utility

usage: fwaudit [options] file [...]

Options:

-o filename	*name of output file (- for stdout)*
-e expr	*filter expression*
-f filename	*read filter expression from file*
-r	*print text report*
-v	*print progress info or report details*

```
$ fwaudit -o 20050705.audit /data/fwlog/*

$ fwaudit -rvv -i'albumcode,composername' \
  -e'timestamp >= 2005-07-05 14:09:06‘ /data/fwlog/*
```

fwaudit utility

```
SIGN-ON session:7
 protocol{7}os{HPUX}ip{127.0.0.1}user{mike}
 uid{102}pid{12283}pname{../putdel 5}

DBPUT MUSIC.SELECTIONS (#489) recno:148065 session:7
 timestamp: 2005-07-05 14:09:06
  ALBUMCODE              : 17358
  COMPOSERNAME           : "Ludwig Beethoven"

DBDELETE MUSIC.SELECTIONS (#489) recno:13343 session:7
 timestamp: 2005-07-05 14:09:06
  ALBUMCODE              : 17358
  COMPOSERNAME           : "Edvard Grieg"

DBUPDATE MUSIC.SELECTIONS (#489) recno:148344 session:7
 timestamp: 2005-07-05 14:09:06
  ALBUMCODE              : 27625
  COMPOSERNAME           : "Amadeus Mozart"
 -COMMENT                : "Comments"
 +COMMENT                : "Comments Updated"
```