

# Migration from TurboIMAGE to Eloquence

Tien-You Chen, Hewlett-Packard  
Michael Marxmeier, Marxmeier Software AG  
info@hp-eloquence.com  
Presentation #515

# Overview

- Eloquence product overview
- Eloquence database concepts
- TurboIMAGE compatibility
- Database utilities
- Installation and configuration
- Data migration

# Product Overview

An introduction to the  
Eloquence product

# Eloquence at a Glance

- Excellent compatibility
- Cost effective
- Supported HP product
- Supports multiple platforms
- Available now

# Eloquence at a Glance

- Eloquence is targeted at small to medium sized sites
- Eloquence provides excellent compatibility and performance for IMAGE based applications

# Eloquence at a Glance

- Eloquence saves considerable time and effort in the migration process and allows focussing on other issues
- Eloquence is easy to manage and retains existing know how
- Eloquence is priced attractively

# When not to consider Eloquence

- New developments or rewrite of applications should likely target a relational database
- Your company has a policy in place and has the staff/experience to administrate a relational database

# Product Components

- Eloquence programming language (based on HP Business Basic)
- Eloquence database (based on IMAGE)
- Graphical User Interface
- Development Environment



# Product Overview

- Eloquence is a product of Hewlett-Packard, Germany
- Support is available through the HP RC or from third party
- Eloquence is available on the HP-UX, Linux and Windows platform

# Product Overview

- About 2500+ installations worldwide
- Used by about 60+ VARs / ISVs worldwide
- Covers a wide range of installations
  - Single user Laptop to 250+ concurrent users with 10+ GB data base 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, ...

# Product Overview

- 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

# The HP 250 Branch Office Computer. You never knew power could be this simple.

---



# Product Overview

- The new Eloquence release is A.07.00
- New functionality
  - New Java / Web based user interface
  - Add object support to the programming language
  - Database enhancements, improved TurboIMAGE compatibility

# Ordering

- The Eloquence product can be ordered
  - from a Eloquence partner
  - from the Eloquence web site or Marxmeier Software AG

# Product Options

- The “unlimited license” option (AH0) provides an unlimited user license
- The “entry license” option (000) and additional user licenses allow competitive pricing for entry level configurations



# Product Options

- The corresponding TurboIMAGE compatibility option (3k) option must be ordered in addition

# Obtaining Eloquence

- The Eloquence product can be obtained by download from the Eloquence web site
- CD-ROM media can 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

# More Information

- Detailed information is available on the Eloquence web site  
**<http://www.hp-eloquence.com>**
- Get in contact:  
**[info@hp-eloquence.com](mailto:info@hp-eloquence.com)**

# Database Concepts

An introduction to the Eloquence  
database architecture

# Overview

- The Eloquence database is almost 100% compatible to TurboIMAGE at the application level
- The underlying architecture is different

# Overview

- Client/Server architecture
  - Network transparent
  - Multiple platforms supported
- Based on IMAGE
  - Excellent performance and compatibility for IMAGE based applications

# Overview

- Additional locking options available
- Dynamic, nested transactions, transaction isolation
- Integrated Indexing, hashing is not used
- Dynamic dataset expansion



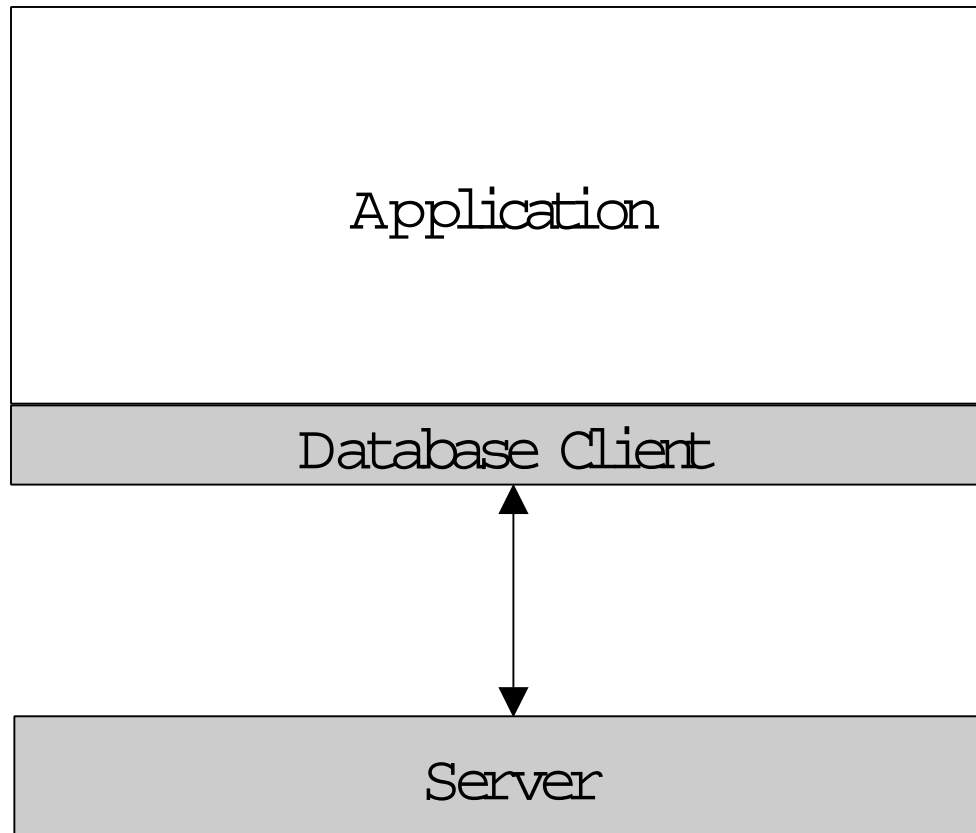
# Overview

- Deadlock detection and recovery
- New security subsystem
- Online backup and forward recovery
- Databases do not reside in the file system
- ROOT file is no longer used

# Client-Server Architecture

- Database access 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

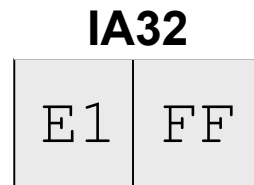
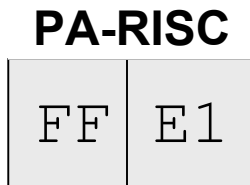
- Eloquence is available for multiple operating systems and architectures
  - HP-UX on PA-RISC
  - Linux on Intel IA32
  - Windows NT/2000/XP on Intel IA32
- IA64 architecture will be added in 2003

# 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

# Network Transparent

- PA-RISC and Intel IA32 use a different byte order
- The example below shows the different representations of the value -31



# TurboIMAGE compatibility

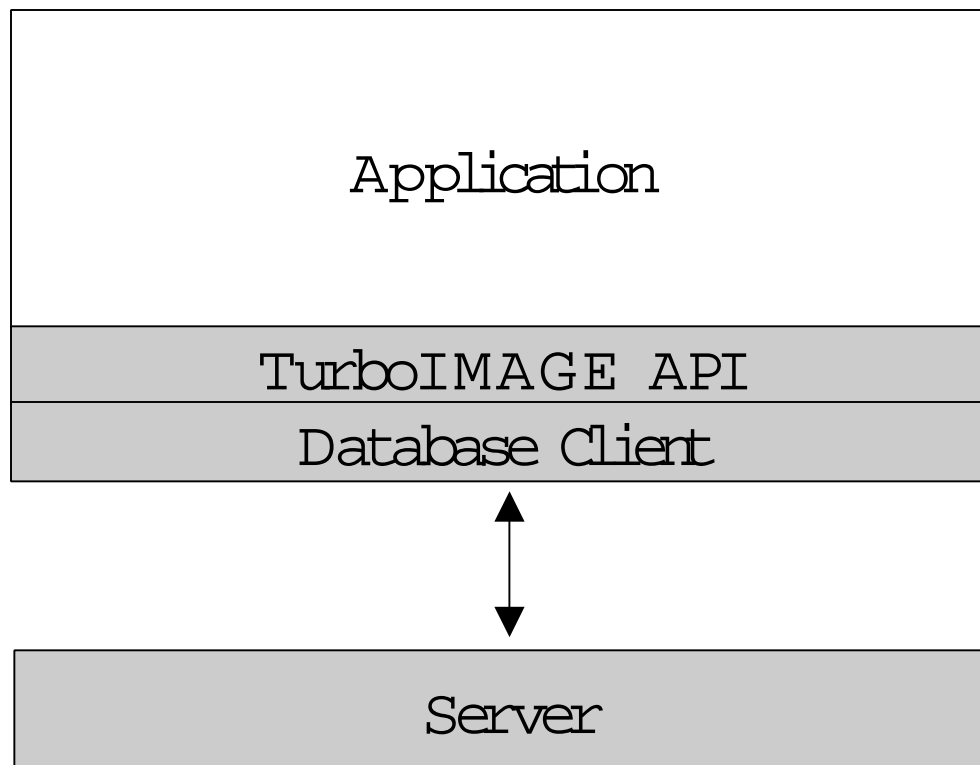
- TurboIMAGE compatibility is implemented at different levels
  - The database server implements functionality at the backend
  - The database client and utilities provide support for TurboIMAGE functionality
  - The TurboIMAGE compatibility API implements source code compatibility

# TurboIMAGE compatibility

- Compatibility goes beyond intrinsic calls and also includes performance profile
- Applications are built on assumptions and take advantage of specific behavior
- Application may depend on external utilities



# 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

- Indexes can be used by the application to implement advanced access methods
  - Partial key retrieval
  - Wildcards
  - Ordered retrieval

# 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 more efficient
- 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 provided at no overhead
  - partial key retrieval
  - ordered retrieval

# 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
- Locking is optional
- Read locking and selective unlocking are available



# Locking

- 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

- 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

# Database Names

- 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

- Syntax

`[ [hostname] [ :service] / ] database`

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

# Database Names

- The EQ\_DBSERVER environment variable can be used to specify a default server
- Syntax  
`[hostname] [: service]`
- Example  
`EQ_DBSERVER=server:eloqdb2`

# Database Security

- The database server maintains a list of users
- A database maintains security groups
  - Similar to TurboIMAGE user classes
  - Database privileges are assigned to groups
  - A user can be a member of multiple groups

# Database Security

- New DBLOGON procedure can be used to specify user and password
- The DBOPEN password argument is ignored
- A default user is used if no specific user is specified



# 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

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

# System Catalog

- Eloquence does not use a root file
- Structural information is maintained in the database environment
- The system catalog is initialized with the dbvolcreate utility and maintained with the schema and dbutil utilities

# System Catalog

- The system catalog contains global and database specific information
- Global catalog information
  - list of users and databases
- Database specific information
  - schema and security information

# 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 (currently limited to 2 GB on HP-UX and Linux)

# Volumes

- A volume file is internally maintained in 8 KB blocks
- Consecutive blocks can be used to form a cluster
- The first block in a volume contains additional volume information

# Volumes

- All volume files in an environment share the same serial number and generation
- Volume file contents are specific to the system architecture (byte order)



# Volumes

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

# Volumes

- The primary data volume contains the system 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 on-line backup

# Volumes

- The transaction journal is used in case of an unexpected server abort to perform a forward recovery of recent transactions
- The transaction log is truncated when it reaches a configured size

# On-line backup

- With on-line backup mode, the data volumes are frozen
- Modifications during on-line backup are temporarily saved into the transaction log volume
- Any backup software can be used to create a consistent backup

# On-line backup

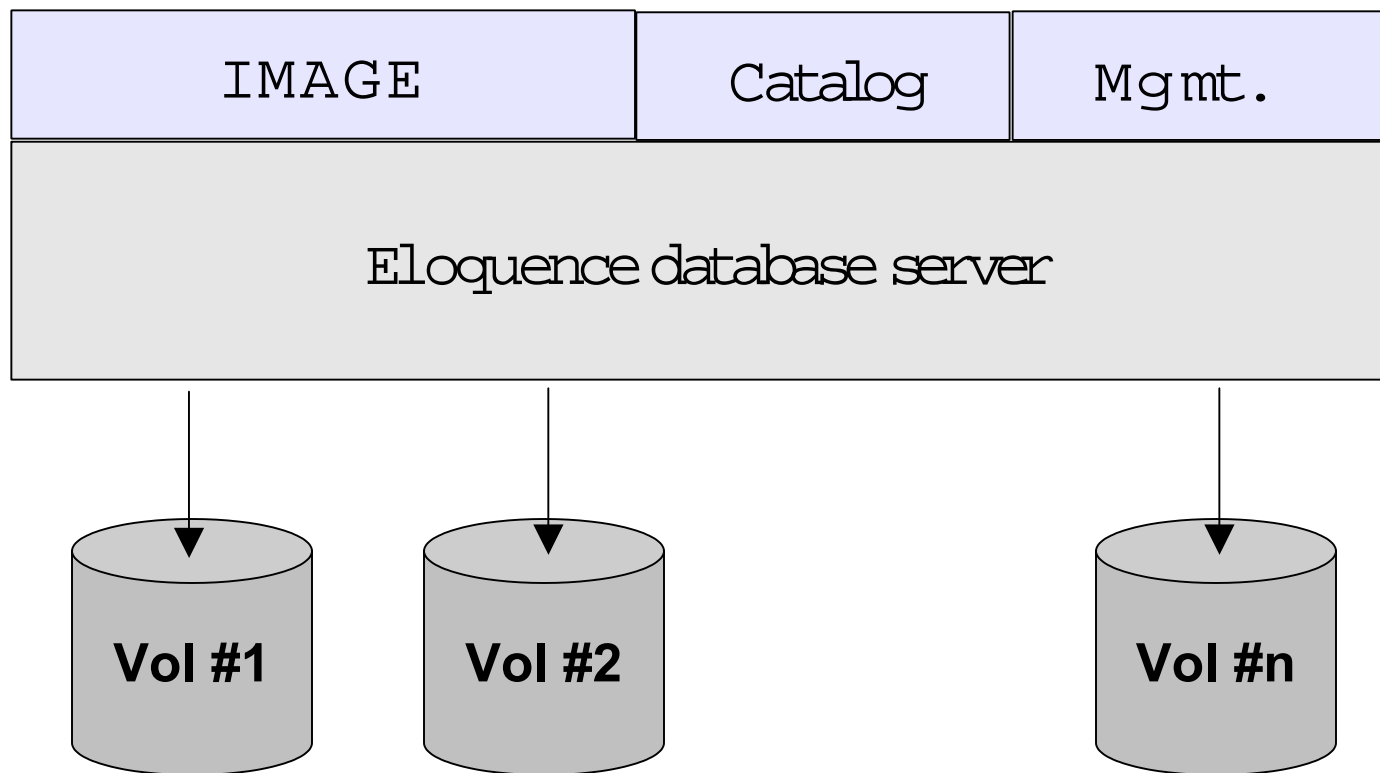
- The dbctl utility is used to control on-line backup mode
- Example backup script

```
$ 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 is fast and involves only minimal processing
- The forward log files can be managed automatically by the server process

# Server Environment





# Scalability

- A.07.00 Image limits
  - 2048 data items
  - 500 data sets
  - 64 / 16 pathes
  - Entry length 5120 bytes

# Scalability

- Database / data set size is limited by disk space allocated to a Eloquence database server.
  - Current limit is ~500 GB.
  - Hard limit is ~32 TB
- Number of concurrent users is currently limited to 1000.

# TurboIMAGE Compatibility

TurboIMAGE compatibility and  
Intrinsics comparison

# TurboIMAGE Compatibility

- All TurboIMAGE intrinsics and almost all modes are supported and behave identical
- HP e3000 applications can usually be ported with no or only minor changes

# TurboIMAGE Compatibility

- Not supported:
  - DBCONTROL modes which are specific to TurboIMAGE implementation details
  - DBCONTROL modes for btree maintenance
  - Item level security

# TurboIMAGE Compatibility

- Locking is optional
- Additional DBLOCK and DBUNLOCK modes are available
- Transaction isolation does not expose uncommitted changes to concurrent sessions
- Transactions cover all databases

# TurboIMAGE Compatibility

- Eloquence implements basic TPI functionality
- Currently not supported TPI functionality
  - Keywords
  - Relational access
  - Independent indexing

# Schema processor

- The Eloquence schema processor accepts HP e3000 schema files
- The -T option specifies TurboIMAGE compatibility
- Database name is not limited to 6 characters



# Schema processor

- 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)

# Database names

- Databases do not reside in the file system but are managed through a server process
- Syntax
  - `[[hostname] [:service] /] database`
    - Hostname specifies database server system
    - Service specifies database instance

# Database security

- The database server maintains a list of users
- A database maintains security groups
  - Similar to TurboIMAGE user classes
  - Database privileges are assigned to groups
  - A user can be a member of multiple groups

# Database security

- New DBLOGIN procedure can be used to specify user and password
- A default user is used if no specific user is specified

# Database name

## IMAGE

- Basename  
\_\_ basename[.group[.account]]
- Password  
password[/USERIDENT]

## Eloquence

- Basename  
[[server][:service]/]basename
- Password (ignored)  
password

## TurboIMAGE compatibility

- Basename  
\_\_ [[server][:service]/]basename
- Password  
password[/USERIDENT]

# DBOPEN modes

## IMAGE

- Mode 1: modify DB, allow concurrent 1,5
- Mode 2: update DB, allow concurrent 2,6
- Mode 3: modify DB, exclusively
- Mode 4: modify DB, allow concurrent 6
- Mode 5: read DB, allow concurrent 1,5
- Mode 6: read DB, allow concurrent 2/4/6/8
- Mode 7: read DB, exclusively
- Mode 8: read DB, allow concurrent 6/8

## Eloquence

- Mode 1: modify DB, allow concurrent read/modify
- Mode 3: modify DB, exclusively
- Mode 8: read DB, allow concurrent read
- Mode 9: read DB, allow concurrent read/modify

# TurboIMAGE Compatibility for DBOPEN modes

- Mode 1: modify DB, same as Eloquence mode 1
- Mode 2: modify DB, map to Eloquence mode 1
- Mode 3: modify DB, same as Eloquence mode 3
- Mode 4: modify DB, map to Eloquence mode 1
- Mode 5: read DB, map to Eloquence mode 9
- Mode 6: read DB, map to Eloquence mode 9
- Mode 7: read DB, map to Eloquence mode 8
- Mode 8: read DB, same as Eloquence mode 8

# DBCLOSE modes

## IMAGE

- Mode 1: close the database
- Mode 2: close the dataset file but still holds the lock
- Mode 3: rewind the dataset

## Eloquence

- Mode 1: close the database
- Mode 3: rewind the dataset



# TurboIMAGE Compatibility for DBCLOSE modes

- Mode 1: close the database
- Mode 2: map to mode 3 of Eloquence
- Mode 3: rewind the dataset

# DBFIND modes

## IMAGE

- Mode 1: non-B-tree find or B-tree find for X and U type
- Mode 4: B-tree find for all item types
- Mode 10: same as old DBFIND, does not use B-tree even if B-tree exist
- Mode 21: same as mode 1 except the chain information is not accurate
- Mode 24: same as mode 4 except the chain information is not accurate

## Eloquence

- Mode 1: find chain head/first match for search item or index item
- Mode 2: find first match for index item
- Mode 3: find last match for index item
- Mode 4: find first match for index item with regular expression
- Mode 5: find last match for index item with regular expression
- Mode 6: implements TPI modes, status array returns no. of qualified records
- Mode 7: implements TPI modes, status array will NOT return no. of qualified records

# TurboIMAGE Compatibility for DBFIND modes

## **IMAGE modes**

- Mode 1: non-B-tree find
- Mode 10: same as old DBFIND, does not use B-tree even if B-tree exist

## **IMAGE B-tree modes (“super-chain”)**

- Mode 1: B-tree find for X and U type
- Mode 4: B-tree find for all item types
- Mode 21: same as mode 1 except the chain information
- Mode 24: same as mode 4 except the chain information

# TurboIMAGE Compatibility for DBFIND modes (cont)

## TPI modes

- Mode 1: TPI DBFIND
- Mode 11: TPI range DBFIND for binary data items
- Mode 21: same as DBFIND mode 1 except entry count
- Mode 22: same as DBFIND mode 11 except entry count
- Modes 1xx to 5xx, TPI DBFIND for =,>,>=,<,<= relation

# DBGET modes

## IMAGE

- Mode 1: re-read, retrieves the current rec.
- Mode 2: serial read, forward
- Mode 3: serial read, backward
- Mode 4: direct read, using the record number in argument parameter
- Mode 5: chain read, forward
- Mode 6: chain read, backward
- Mode 7: calculated read, master only
- Mode 8: calculated read, primary location

## Eloquence

- Mode 1: same as IMAGE mode 1
- Mode 2: same
- Mode 3: same
- Mode 4: same
- Mode 5: same, also for index access
- Mode 6: same, also for index access
- Mode 7: same
- Mode 15: next in index order
- Mode 16: previous for index order

# TurboIMAGE Compatibility for DBGGET modes

- Mode 1: re-read, same
- Mode 2: serial read, same
- Mode 3: serial read, same
- Mode 4: direct read, same
- Mode 5: chain read, same
- Mode 6: chain read, same
- Mode 7: calculated read, same
- Mode 8: calculated read, same as mode 7

# DBPUT modes

## **IMAGE**

- Mode 1: the only mode for DBPUT

## **Eloquence**

- Mode 1: the only mode for DBPUT

# TurboIMAGE Compatibility for DBPUT modes

- Mode 1: the only mode for DBPUT



# DBDELETE modes

## **IMAGE**

- Mode 1: the only mode for DBDELETE

## **Eloquence**

- Mode 1: the only mode for DBDELETE

# TurboIMAGE Compatibility for DBDELETE modes

- Mode 1: the only mode for DBDELETE

# DBUPDATE modes

## **IMAGE**

- Mode 1: the only mode for DBUPDATE

## **Eloquence**

- Mode 1: for normal DBUPDATE
- Mode 2: for critical item update

# TurboIMAGE Compatibility for DBUPDATE modes

- Mode 1: for normal DBUPDATE
- Mode 2: DBUPDATE with CIUPDATE
  - New mode 2 is equivalent to CIUPDATE
  - If CIUPDATE mode is set (DBCCONTROL mode 5) , mode 1 is mapped to mode 2

# DBINFO modes (1xx)

## IMAGE

- Mode 101: type of access for an item
- Mode 102: describes an item
- Mode 103: items in database
- Mode 104: items in data set
- Mode 113: Btreemode1 & wildcard

## Eloquence

- Mode 101: item number for a given item
- Mode 102: describes an item
- Mode 103: items in database
- Mode 104: item in data set

# TurboIMAGE Compatibility for DBINFO modes (1xx)

- Supports mode 101 ~ 104 and 113

# DBINFO modes (2xx)

## IMAGE

- Mode 201: type of access for data set
- Mode 202: describes a data set
- Mode 203: sets in database
- Mode 204: sets with item
- Mode 205: set capacity
- Mode 206: no. of data set chunks
- Mode 207: size of each chunk for a set
- Mode 208: primary/current/max cap
- Mode 209: B-tree attachment

## Eloquence

- Mode 201: type of access for data set
- Mode 202: describes a data set
- Mode -202: describes a data set, same as 202 but processed locally
- Mode 203: sets in database
- Mode 204: sets with item

# TurboIMAGE Compatibility for DBINFO modes (2xx)

- Mode 201: same
- Mode 202: blocking factor does not apply, always 0
- Mode 203: same
- Mode 204: same
- Mode 205: few elements return static value
- Mode 206: jumbo does not apply, always 0
- Mode 207: jumbo does not apply, 0 for element 1 and 2
- Mode 208: most elements return static value
- Mode 209: Btreemode1 does not apply, 0 for element 1 and 2



# DBINFO modes (3xx)

## IMAGE

- Mode 301: path info for a data set
- Mode 302: key/search item for a data set

## Eloquence

- Mode 301: path info for a data set
- Mode 302: key/search item for a data set
- Mode 303: path info for a data set, returns search item

# TurboIMAGE Compatibility for DBINFO modes (3xx)

- Mode 301: same
- Mode 302: same

# DBINFO modes (4xx)

## IMAGE

- Mode 401: info related to user logging
- Mode 402: info related to ILR
- Mode 403: info related to dynamic txn
- Mode 404: info related to multiple database txn
- Mode 406: database informaiton

## Eloquence

- Eloquence does not have 4xx modes

# TurboIMAGE Compatibility for DBINFO modes (4xx)

- Mode 401: Eloquence does not support user logging, returns space for element 1 – 4 and 0 for the rest
- Mode 402: Eloquence does not support ILR, 0 will be returned
- Mode 403: most elements are 0, only a few elements meaningful
- Mode 404: most elements are 0, only a few elements meaningful
- Mode 406: returns up to 28 characters of database name with other info

# DBINFO modes (5xx)

## IMAGE

- Mode 501: subsystem access
- Mode 502: critical item update setting

## Eloquence

- Mode 501: identify the index item
- Mode 502: describe an index item
- Mode 503: identify all index items
- Mode 504: identify all index items in a data set
- Mode 705: identifies all data sets available which contain an index item

# TurboIMAGE Compatibility for DBINFO modes (5xx)

- Mode 501: always return 3
- Mode 502: critical item update is always “allowed”

# DBINFO modes (9xx)

## **IMAGE**

- Mode 901: language ID for the database

## **Eloquence**

- No equivalent mode for language setting

# TurboIMAGE Compatibility for DBINFO modes (9xx)

- Mode 901: always returns 0



# TurboIMAGE Compatibility for DBINFO mode (7xx)

## Eloquence specific modes

- Mode 701: returns an index item number
- Mode 702: describes a specific index item
- Mode 703: identifies index items in database
- Mode 704: identifies index items in data set
- Mode 705: identifies all data sets available which contain an index item

# TurboIMAGE Compatibility for DBINFO mode (8xx)

## TPI modes

- Mode 801: describes third-party product on the database
- Mode 802: number of external and internal index files
- Mode 803: indicates if TPI is enabled
- Mode 811: access available for a third party key
- Mode 812: describes a third party key
- Mode 813: all items (including index item) in database

# TurboIMAGE Compatibility for DBINFO mode (8xx, cont)

- Mode 814: all items (including index item) in a specific data set
- Mode 821: sets with index item
- Mode 831: identifies index items available in a specific data set
- Mode 832: identifies third party keyword search keys for a data set, always return 0
- Mode 833: describes a specific third party key
- Mode 834: describes all items grouped with the specified key, always return 0

# DBCCONTROL modes

## IMAGE

- Mode 1,2: enable/disable for autodefer
- Mode 5,6: enable/disable critical item update
- Mode 7: enable for dynamic multiple database transaction
- Mode 9,10: enable/disable for HWMPUT
- Mode 13: adding/dropping/rebuilding B-tree for PRIV program
- Mode 14: btreamode1 related setting for PRIV program
- Mode 15,16: enable/disable the btreamode1
- Mode 18,19: enable/disable to return warning for large transaction
- Mode 20: temp negate the DBUTIL “only return old IMAGE limit” setting

## Eloquence

- No equivalent intrinsic

# TurboIMAGE Compatibility for DBCCONTROL modes

- Modes 5,6: enable/disable critical item update
- Modes 15,16: enable/disable btreemode1
- Modes 1,2,7,18,19,20 are accepted but have no effect
- Modes 9,10,13,14 are not accepted

# DBERROR / DBEXPLAIN

- Equivalent functionality
- Error messages may be different

# DBLOCK modes

## IMAGE

- Mode 1: wait database lock
- Mode 2: no-wait database lock
- Mode 3: wait data set lock
- Mode 4: no-wait data set lock
- Mode 5: wait predicate lock
- Mode 6: no-wait predicate lock

## Eloquence

- Mode 1 – 6: same as IMAGE
- Mode 11 – 16: read locks for database, data set and predicate lock

# TurboIMAGE Compatibility for DBLOCK modes

- Mode 1: wait database lock
- Mode 2: no-wait database lock
- Mode 3: wait data set lock
- Mode 4: no-wait data set lock
- Mode 5: wait predicate lock
- Mode 6: no-wait predicate lock
- Mode 11: read only wait database lock
- Mode 12: read only no-wait database lock
- Mode 13: read only wait data set lock
- Mode 14: read only no-wait data set lock
- Mode 15: read only wait predicate lock
- Mode 16: read only no-wait predicate lock



# DBUNLOCK modes

## IMAGE

- Mode 1: unlock all the locks

## Eloquence

- Mode 1: unlock all locks
- Mode 3: unlock the resource which was locked by calling DBLOCK mode 3/4/13/14
- Mode 5: unlock the resource which was locked by calling DBLOCK mode 5/6/15/16

# TurboIMAGE Compatibility for DBUNLOCK modes

- Mode 1: unlock all the resources owned
- Modes 3, 5, 11, 13, 15 allow selective unlocking
- DBUNLOCK during dynamic transaction is valid but delayed until transaction completes

# DBBEGIN / DBEND modes

## IMAGE/DBBEGIN

- Mode 1: starts a transaction, a 'begin' record is written to IMAGE user log file
- Mode 3: start a transaction involves more than one database; one 'begin' record for every database is written to IMAGE log file
- Mode 4: start a transaction involves more than one databases, only one 'begin' record is written to IMAGE log file

## IMAGE/DBEND

- Mode 1: ends a transaction
- Mode 2: ends a transaction and flush the log buffer to the disk
- Mode 3: ends a multiple database transaction started by DBBEGIN mode 3, each database has one 'end' record
- Mode 4: ends a multiple database transaction started by DBBEGIN mode 4, only one log record is written to the log file

# DBMEMO modes

## IMAGE

- Mode 1: the only legal mode; writes a 'memo' record to the log file

# DBXBEGIN modes

## IMAGE

- Mode 1: starts a dynamic transaction for the current database
- Mode 3: starts a dynamic transaction for multiple database; user has to provide a database name list
- Mode 4: starts a dynamic transaction for multiple database; user can add database involved one at a time. That is only used by IMAGE/SQL

## Eloquence

- Mode 1: starts a transaction.

# TurboIMAGE Compatibility for DBXBEGIN modes

- Mode 1: starts a transaction
- Mode 3: starts a transaction
  
- Allows nested transactions
- Transactions cover all databases
- Transaction isolation does not expose partial transactions
- Transactions are not limited in size

# DBXEND modes

## IMAGE

- Mode 1: ends a dynamic transaction
- Mode 2: ends a dynamic transaction and write the contents of the XM log buffer to the disk
- Mode 3: ends a dynamic transaction with multiple databases involved. User has to provide the base name list with the global transaction ID in front
- Mode 4: ends a dynamic transaction with multiple databases involved. This is only used by IMAGE/SQL

## Eloquence

- Mode 1: commit the current sub-transaction or top transaction if there are no sub-transactions
- Mode 2: commit the top level transaction

# TurboIMAGE Compatibility for DBXEND modes

- Mode 1: ends a dynamic transaction
- Mode 2: ends a dynamic transaction
- Mode 3: ends a dynamic transaction
- Mode 5: force end of a top level transaction



# DBXUNDO modes

## IMAGE

- Mode 1: rollback the dynamic transaction
- Mode 3: rollback the dynamic transaction started by calling DBXBEGIN mode 3
- Mode 4: rollback the dynamic transaction started by calling DBXBEGIN mode 4

## Eloquence

- Mode 1: rollback the current transaction
- Mode 2: rollback a given transaction where user has to provide the transaction ID
- Mode 3: rollback to top level transaction

# TurboIMAGE Compatibility for DBXUNDO modes

- Mode 1: rollback the current transaction
- Mode 3: rollback the current transaction
- Mode 5: rollback to top level transaction

# Database Utilities

An overview on the Eloquence  
database utilities

# Offline utilities

- dbvolcreate / dbvolextend / dbvolchange / dblogreset - database volume management
- dbvoldump - display volume properties
- dbfsck - volume consistency check and simple repair tool
- dbrecover - forward recovery

# Administrative utilities

- dbctl - server management utility
- HTTP status monitor

# Database utilities

- schema - Schema processor
- dbcreate / dberase / dbpurge - create / erase / purge database
- dbtables - database cross reference
- prschema - re-create schema from database
- dbdumpcat - catalog information utility

# Database utilities

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

# dbvolcreate

- The dbvolcreate utility creates the primary database volume and initializes the system catalog
- The volume file is added to the server configuration file



# dbvolextend

- The dbvolextend utility is used to extend the database environment with additional volume files.
- The specified volume file is added to the server configuration file

# dbvolchange

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

# dblogreset

- The dblogreset utility is used to reset the transaction log volume to minimal size.
- dblogreset processes any information in the transaction journal or saved temporarily in the log volume(s)

# dbvoldump

- The dbvoldump utility is used to display volume properties

# 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

# dbfsck utility

- The dbfsck can be used to check volume consistency and perform simple repairs
- It currently does not verify IMAGE integrity

# dbrecover

- The dbrecover utility is used to perform forward recovery
- Speed depends on disk performance, only minimal processing is involved

# HTTP status

- The HTTP status monitor is a web server built into the database server.
- A web browser can be used to check the database server status.
- The dbctl command line interface 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 (config)

## 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, <i>unlimited</i>

## Server Configuration

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

# HTTP status (session)

## Active Sessions

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

Session	TID	Host	IP ADDR	Platform	User	Login
0x827d6d0	8	lxmike.marzmeier.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)

## 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
PPSSC	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)

## 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)

## Server Volumes

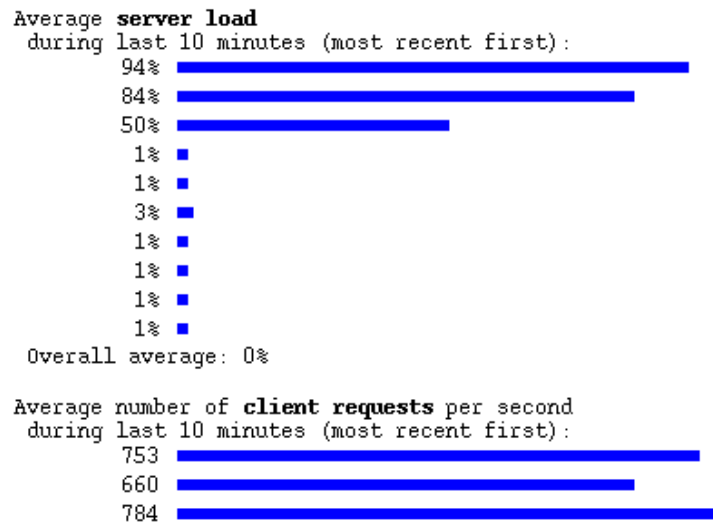
ID	Type	Path
1	DATA	/data3/dalex/db/db01.vol
2	LOG	/data3/dalex/db/db02.vol
3	DATA	/data3/dalex/db/db03.vol
4	DATA	/data3/dalex/db/db04.vol
5	DATA	/data3/dalex/db/db05.vol
6	DATA	/data3/dalex/db/db06.vol
7	DATA	/data3/dalex/db/db07.vol

## 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)

## Performance Statistics





# HTTP status (statistics)

## 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.

# dbctl utility

- The dbctl utility is used to submit administrative requests to the database server
- It is used for administrative purposes to either manage the server operation or request status information.

# dbctl commands

- help [command]
- shutdown [seconds|CANCEL]
- backup {START|STOP}
- list [session|db|lock|thread]
- dbstore database target
- dbrestore [/info] source [dbase\_name]

# dbctl commands (cont)

- cancelthread {tid}
- killthread {tid}
- logfile {logfile}
- logflags {logflags}

# schema processor

- The schema processor creates the database from a schema file
- The Eloquence schema processor understands both Eloquence and TurboIMAGE schema files
- Option -T selects TurboIMAGE compatibility mode

# schema processor

- If you omit the -T option the schema file is understood as Eloquence schema file
  - schema may succeed but the resulting database will likely be incompatible
  - Eloquence item sizes are specified in bytes while TurboIMAGE uses halfwords, bytes or nibbles, depending on item type

# 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 master set) the path information in the master set is reset

# 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



# dbtables utility

- The dbtables utility is used to output information on the database structure
  - Items
  - Data sets
  - Paths
  - Entries
  - Indexes

# dbtables utility (cont)

## DATA SET FIELDS

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 the system catalog

# dbdumpcat utility (cont)

```
$ dbdumpcat -h lxsrv:8302 -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

```
...
```

# dbexport utility

- The dbexport utility is used to export the database content to text files
- By default, a separate file is created for each data set

# 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 text files

# export file format

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

<http://www.hp-eloquence.com/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 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 central point for database administration
  - security management
  - structural maintenance
- dbutil can be used interactively (dialog based) or controlled by a script

# QUERY utility

- The Eloquence QUERY utility is different from the e3000 QUERY
  - limited capabilities
  - uses different syntax
  - supported datatypes are limited
  - a TurboIMAGE compatible QUERY version will be added to Eloquence

# Installation and Configuration

Installation and Configuration of  
the Eloquence database

# Overview

- Install the product
- Configure the operating system
- Configure automatic server startup
- Create the database environment
- Platform differences

# Obtaining the product

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



# Installation on HP-UX

- Run “swinstall” to install the product

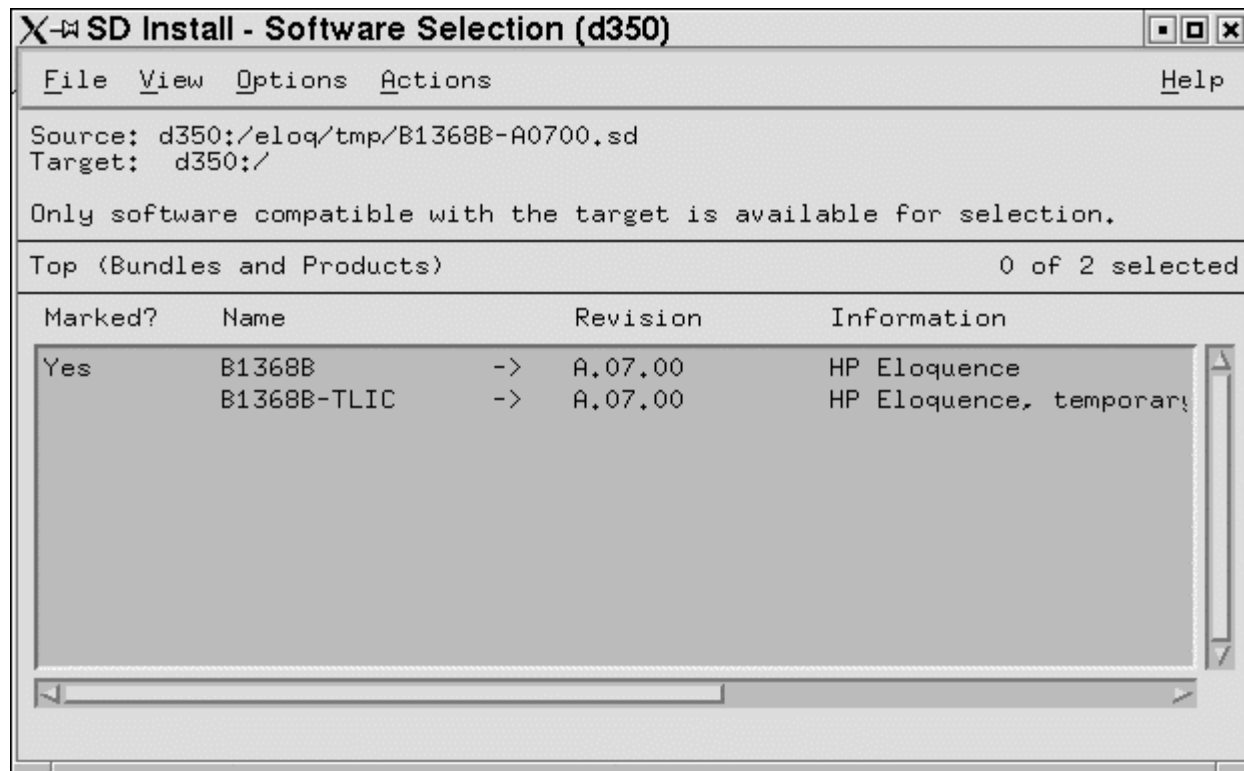
```
/usr/sbin/swinstall -s /path/to/B1368B-A0700.sd
```

- the -s argument specifies the absolute path of the product archive
- requires root privileges

# Installation on HP-UX

- Mark the product bundle for installation
  - B1368B: install the base product
  - B1368B-TLIC: install a 30 days temporary license in addition
- Select Actions -> Install to continue the installation

# Installation on HP-UX



# Install HP-UX patches

- HP-UX 11.0
  - PHKL\_24005 or successors
- HP-UX 10.20
  - PHCO\_10947 or later

# Evaluation License

- By default the “Personal Edition” license key is installed
- A temporary license key can be created during installation
- A temporary license key can be requested from the Eloquence web site

# Install new License Key

- Add supplied license keys to the `/etc/opt/eloquence6/license` file
- Comment out any existing license keys
- Use the `/opt/eloquence6/etc/chklic` utility to verify your license file

# Configuration of TCP Services

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

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

# Create eloqdb user/group

- Create a user name and a group name e.g. eloqdb to be used as the owner/group of the database files
- On Windows the system account is used by default



# HP-UX Configuration Kernel Parameters

- HP-UX kernel parameters need to be configured
  - semaphores related parameters
  - shared memory related parameters
  - data size

# HP-UX Configuration

## Automatic server start

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

```
START_ELOQSD=0  
START_ELOQDB6=1  
START_ELOQDB5=0
```

# Setup database environment

- Database environment (server instance) consists of
  - Server configuration file (eloqdb.cfg)
  - Primary data volume
  - Transaction log volume(s)
  - Additional data volume(s)

# Server configuration file

- Default server configuration file is `/etc/opt/eloquence6/eloqdb6.cfg`
- This file defines server properties
  - configuration
  - scaling and tuning parameters
  - volume files

# Example Configuration

```
[Server]
Service = eloqdb
ServiceHTTP = 8103
UID = eloqdb
GID = eloqdb
EnableIPC = 1
SyncMode = 0
```

# Example Configuration

[Config]

Threads = 100

IOThreads = 4

BufferCache = 64

CheckPtSize = 20

# Create volume files

- dbvolcreate  
/var/opt/eloquence6/data01.vol
- dbvoextend -t log  
/var/opt/eloquence6/log.vol
- dbvoextend -t data  
/var/opt/eloquence6/data02.vol

# HP-UX

## Start the database server

- Start the default database server
  - `/sbin/init.d/eloq6 start`
- Check if the server is active
  - `ps -ef | fgrep eloqdb6`
- Stop the default database server
  - `/sbin/init.d/eloq6 stop`



# Troubleshooting

- The Eloquence database writes diagnostic messages to the syslog
  - HP-UX: /var/adm/syslog/syslog.log
  - Linux: /var/log/messages
  - Windows: application event log

# Linux Installation

- Eloquence uses the RPM package manager
  - RedHat Linux 7.3 and SuSE Linux 8.0 have been certified
  - Other Linux distributions can be used but additional manual steps may be required

# Linux Installation

- For installation or update execute the command below

```
$ rpm -U B1368B-A0700.glibc2.1-1.i386.rpm
```

- Temporary license option is not available during installation

# Linux Configuration

## Automatic server start

- Modify the startup configuration file  
`/etc/sysconfig/eloq6`

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

# Linux

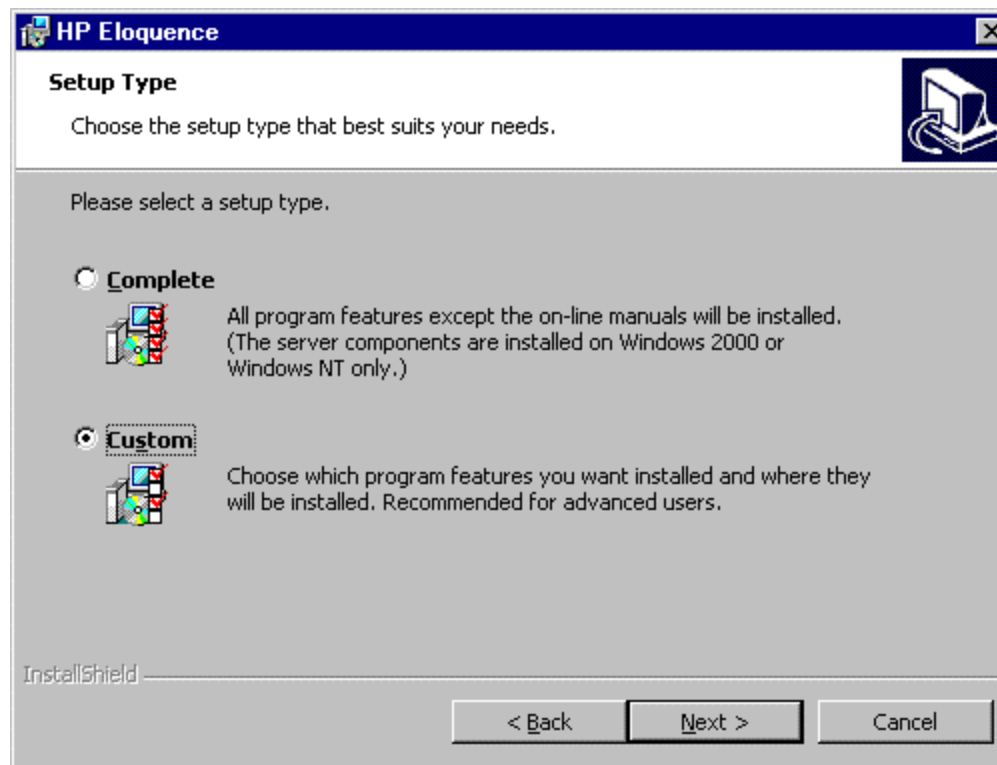
## Start the database server

- Start the default database server
  - `/etc/init.d/eloq6 start`
- Check if the server is active
  - `/etc/init.d/eloq6 status`
  - `ps ax | fgrep eloqdb6`
- Stop the default database server
  - `/etc/init.d/eloq6 stop`

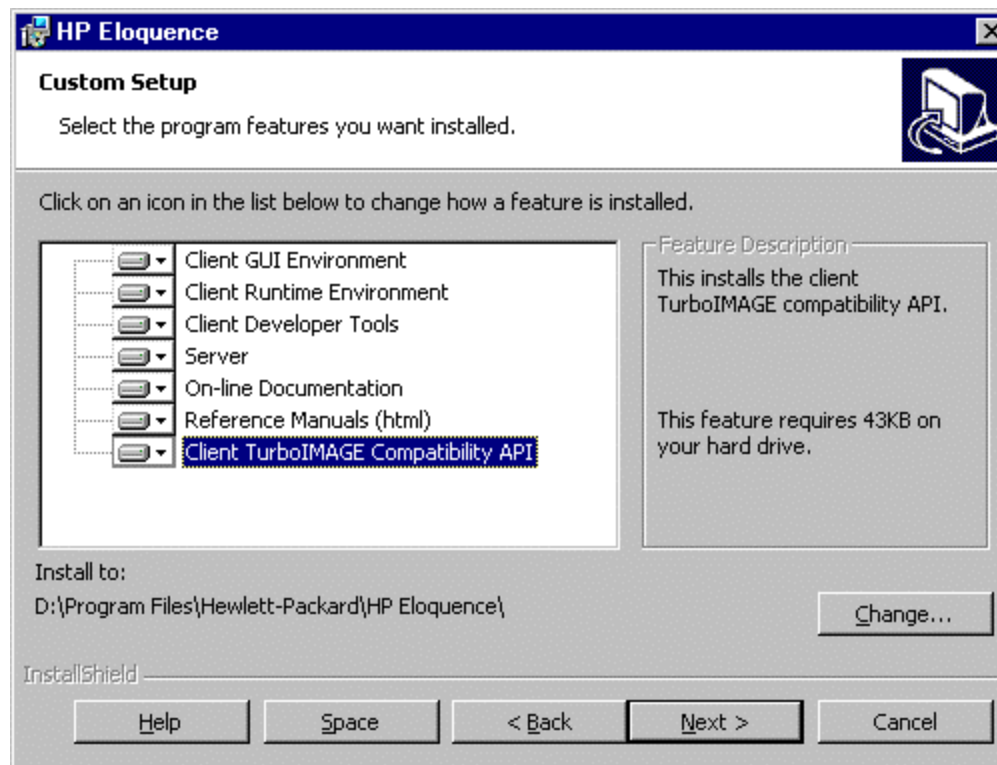
# Windows Installation

- Eloquence uses the standard Windows Installer for the product install
- Custom installation allows to select partial installation

# Windows Installation



# Windows Installation



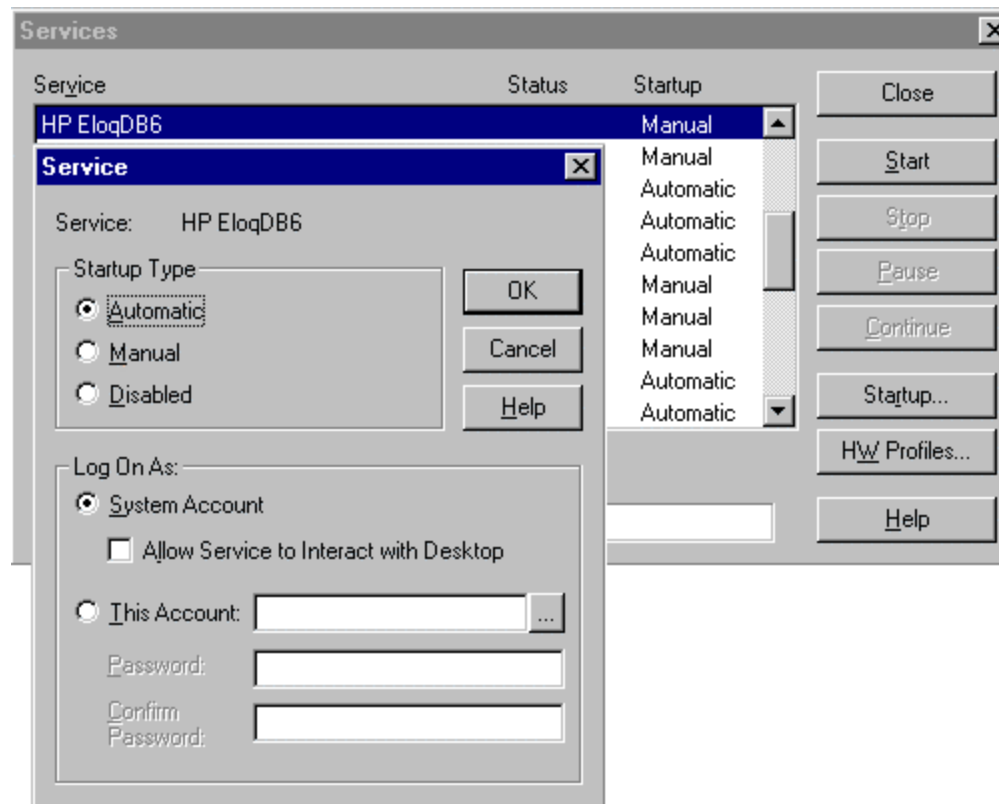


# Windows Configuration

- The installation program has registered the “HP EloqDB6” service
- Configure automatic start mode for the Eloquence database in the service control panel
- Start the service manually for the first time

# Windows Configuration

## Automatic server start



# Data Migration

Move your databases from  
TurboIMAGE to Eloquence

# Overview

- The equivalent of the Eloquence dbexport utility is available on MPE to export database content to flat files
- Transfer the schema file and the export files to the target system
- On the target system run the schema processor, the dbcreate utility and the dbimport utility

# DBEXPORT utility

- DBEXPORT is used to export the database content to one or multiple text files
- It provides an easy procedure to move your database content to Eloquence

# DBEXPORT utility

Usage: DBEXPORT [options] database [set ...]

options:

- help - show usage (this list)
- u user - set user identifier
- p pswd - set password
- o path - set output directory (not single file)
- v - verbose output
- c - chained export
- r - output restructure information
- s file - output into single file, '-' = stdout
- f sep - field separator, default is ','

# DBEXPORT utility

- Options
  - The -p passwd option specifies the database password used to open the data base
  - The -s file option specifies a single file should be used which contains all data sets
  - The -v options displays the progress
  - Any data set name or number specified after the database name indicates only the specified data sets should be exported

# DBEXPORT utility

```
: DBEXPORT "-v TESTDB"  
Processing database : TESTDB
```

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



# DBEXPORT utility

- By default the export files are named DBNAME.###.exp and are created in the current directory
- Option -o allows to specify a different target directory

# EXPORT file format

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

<http://www.hp-eloquence.com/support/misc/export.html>

# EXPORT file format

```
"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 including data set name, type, number of entries and capacity

# DBINFO utility

Usage: DBINFO [options] database [set ...]

options:

- help - show usage (this list)
- u user - set user identifier
- p pswd - set password

# DBINFO utility

- Options
  - The -p passwd option specifies the database password used to open the data base
  - Any data set name or number specified after the database name indicates only the specified data sets should be processed

# DBINFO utility

: DBINFO TESTDB

Processing database : TESTDB

SET NAME			RECLLEN	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

# MPE Migration tools

- The MPE DBINFO and DBEXPORT utilities are available as source code
- Compiled versions for MPE 6.0, 6.5 and MPE 7.0 are available in a tar or STORE archive format
- For download please refer to <http://www.hp-elocquence.com/hp3k/>



# Installation on MPE

- To install the tar archive on your system
  - Enter the POSIX shell
  - Use tar to extract the files

```
$ tar -xvf hp3kbin70.tar
```
  - tar may issue a warning about being unable to restore the user and group which can be safely ignored

# Installation on MPE

- To install the STORE archive on your system
  - : **FILE HP3KBIN= ./HP3KB70 ;DEV=DISC**
  - : **restore \*HP3KBIN ; ;LOCAL ;SHOW**
  - This extracts the files in the current directory

# Export the database

- When running from the POSIX shell the arguments are separated by a space

```
$ DBEXPORT -p SECRET -v TESTDB
```

# Export the database

- When running from the MPE shell (CI) you need to enclose the arguments in quotes

```
: DBEXPORT "-p SECRET -v TESTDB"
```

# Transfer the files

- Transfer your schema file and the export files to the Eloquence system
- When transferring by ftp
  - use text mode to transfer the schema file
  - use binary mode to transfer the export files

# Create and import the database

- On Eloquence, run the schema and dbcreate utilities to create the database
- Then use dbimport to load the database from the export files
- To use Eloquence indexes add them to the schema file or use the dbutil utility to add them subsequently

# Create the database

- Run the Eloquence schema processor

```
$ schema -T schemafile
```

- Option -T selects TurboIMAGE compatibility mode

# Create the database

- If you omit the -T option the schema file is understood as Eloquence schema file
  - schema may succeed but the resulting database will likely be incompatible
  - Eloquence item sizes are specified in bytes while TurboIMAGE uses halfwords, bytes or nibbles, depending on item type.



# Create the database

- After running schema, you need to create the database

```
$ dbcreate database
```

# Import the data

- Use dbimport to load the database

```
$ dbimport -v database
```

- The option -v displays the import progress

# Import the data

- On the Windows and Linux platform you should specify the -z roman8 option to indicate the source data uses the HP-ROMAN8 encoding
- This makes sure any national characters ("Umlaute") are converted

# More Information

- Detailed information is available on the Eloquence web site  
**<http://www.hp-eloquence.com>**
- Get in contact:  
**[info@hp-eloquence.com](mailto:info@hp-eloquence.com)**