

OPENTURBO™ for Eloquence

iMaxsoft

iMaxsoft Corporation

Version B.01.00

WE'LL ALWAYS BE HERE

*Industry leader in database migration and middleware products
Based in Cupertino, CA since 1987*

Revision History

Date	Revision	Author	Changes
6/07/2006	1.0	Oliver Wai	Version 1.0

Table of Contents

CONVENTIONS USED IN THIS MANUAL	4
OPENTURBO OVERVIEW	4
OPENTURBO LIBRARIES.....	6
<i>Eloquence Applications and/or Tools Requiring Special Linkage.....</i>	<i>6</i>
OPENTURBO CONFIG FILE.....	8
CRITICAL RULES BEFORE YOU START.....	8
CONFIG FILE COMMAND SYNTAX	9
DATABASE LEVEL CONFIGURATION	9
DATASET LEVEL CONFIGURATION	13
ENCRYPTING/DECRYPTING PASSWORDS.....	13
OPENTURBO SYNCHRONIZATION MODES	14
STARTING THE OPENTURBO LISTENER	14
<i>For Eloquence only:.....</i>	<i>15</i>
FROM HPE3000 TO HP-UX	17
FROM HP-UX TO HPE3000	17
TROUBLESHOOTING	18
OPENTURBO UTILITIES.....	20
CROSSREF FILE	20
LANUTIL	24
LISTNER	25
OTDBUTIL	28
RECOVER, READER.....	29
TIDRV D02DBG_ELOQUENCE, TIDRV D02DBG_OMNIDEX	30
TILOAD, TILOADAM.....	39
DEBUGGING CAPABILITIES.....	41
DISASTER RECOVERY: RECOVERING FROM CACHE	43
RECOVERY PROCESS.....	45
<i>HP-UX as Source Server.....</i>	<i>45</i>
<i>HPE3000 as Source Server.....</i>	<i>46</i>
<i>Recovering From Unsupported Failures</i>	<i>46</i>
APPENDIX	47
APPENDIX A: OPENTURBO ERROR MESSAGES.....	47
APPENDIX B: TURBOIMAGE DATA CONVERSION REFERENCE.....	52

Conventions Used In This Manual

- **Green Entries:** need modification when synchronizing from HP 9000 to HP e3000
- **Blue Entries:** specify the differences between [1] database to be synchronized and [2] database to be updated locally
- **Highlighted entries :** highlight special specification required
- **Red Entries:** error messages to pay attention to

OPENTURBO Overview

The OPENTURBO allows you to run your IMAGE legacy application with a Relational Database Management System (RDBMS) without any migration recoding. OPENTURBO is customizable and adaptable to a variety of infrastructure setup. With OPENTURBO it is now possible to run legacy applications on:

1. Running legacy applications on a HP 3000 server with a remote RDBMS backend system.
2. Running legacy applications on the RDBMS server (i.e. HP 9000)
3. Running legacy applications on a separate (non-HP 3000) application server with a RDBMS server.

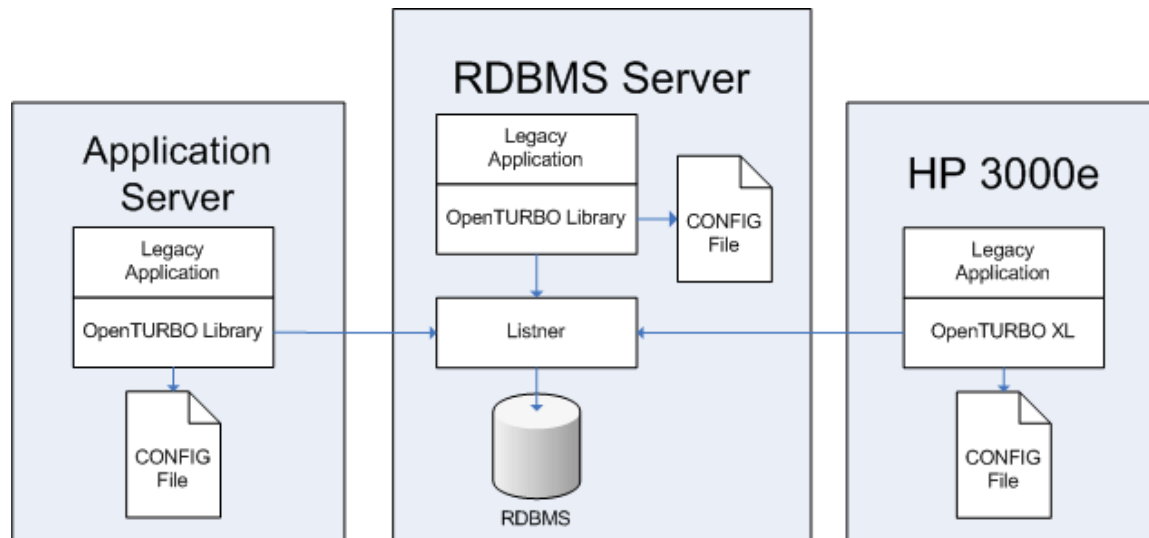


Figure 1: OPENTURBO Configurations

Alternatively, depending on the migration strategy, OPENTURBO gives the ability to maintain mirror database on a RDBMS and IMAGE DB. Known as, *DUALMODE*, OPENTURBO allows for unidirectional data replication and synchronization across an IMAGE database and any other RDBMS.

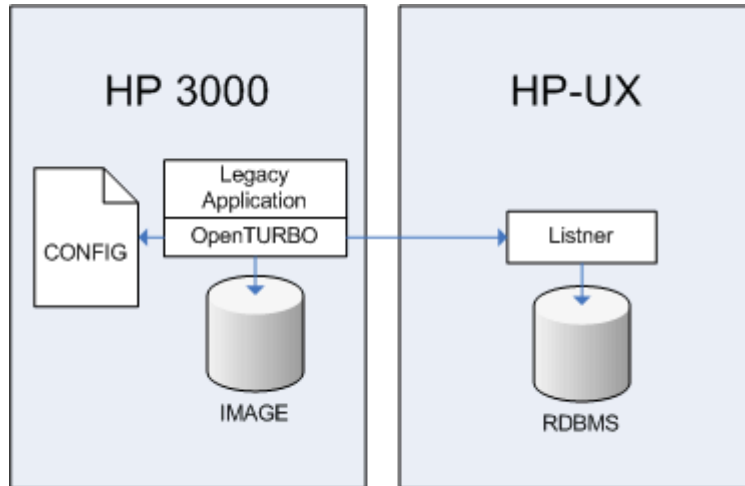


Figure 2: Mirror IMAGE & RDBMS w/legacy application on HP3000

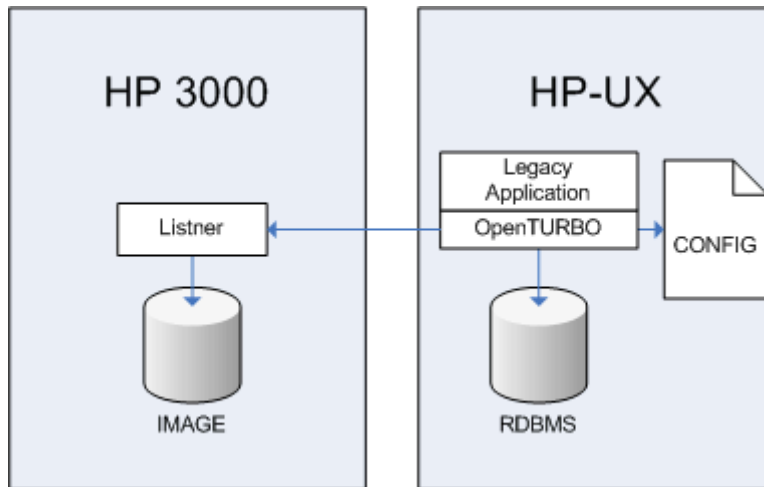


Figure 3: Mirror IMAGE & RDBMS w/ legacy application on HP-UX

The level of customization will allow enterprises flexibility in planning migration strategies, from simply migrating to a RDBMS backend to a full migration of both the business application and the database backend.

OPENTURBO Libraries

OPENTURBO for Eloquence supports 3 different HP architectures. These include:

- PA-RISC 1.1 (32bit)
- Itanium IA64 (32bit)
- Itanium IA64 (64bit)

The core OPENTURBO library package consists of the following libraries:

Library	Description
otxlelo	OPENTURBO library to be used in production applications
otxleld	Debug version of the OPENTURBO library used during development
otqelo	OPENTURBO library used with HP3000 Query (located in <code>QUERY.PUB.SYS</code> on the HP300 system). The Query program cannot link the OPENTURBO library. Consequently a special library is needed to work around this limitation.
libimaxsoft3k.sl	OPENTURBO IMAGE emulator API Library
libsdk.sl	Underlying client side database engine library

Dynamic client libraries on HPe3000 MPE/XL:

Library Name	Description
LTXL	AIM Middleware and Debugging Facility Core Library
OTXL	OPENTURBO Core Library
OTXLDBG	OPENTURBO Core Library with Debugger
OTQRY	OPENTURBO Core Library for <code>QUERY.PUB.SYS</code>
OTQRYDBG	OPENTURBO Core Library for <code>QUERY.PUB.SYS</code> with Debugger
DRIVER	Utilities Core Library
TIDRV	TurboIMAGE Test Driver Core Library

Eloquence Applications and/or Tools Requiring Special Linkage

In most cases, your applications will be able to compile and run applications with the OPENTURBO library by linking the OPENTURBO library to Eloquence. However depending on the applications and tools used in conjunction with OPENTURBO, it may not be possible to link the OPENTURBO library directly to your application.

HP3000 Query

OPENTURBO intercepts HP3000 Query program (QUERY.PUB.SYS) calls and redirects them to the remote database. OPENTURBO uses a special version of Query to perform this task. If you would normally do:

```
:run query.pub.sys
```

to access a local Image database, then the following would be used to use invoke OPENTURBO:

```
: run query.local.myacct;xl="OTQELO.pub.imaxsoft"
```

NOTE: Do not attempt to replace query.pub.sys as it is a system file. Instead, make a local copy and put it in the same account.

DISC Omnindex

HP3000 Set-Up

- XLLIST order should be: OMIDEX, OPENTURBO, and then XL.PUB.SYS. All OMNIDEX related processes are automatically handled via IMAGE TPI call-back process at the database level.

HP9000 Set-Up

- In order to use OMNIDEX, libelo3k.sl must link to the OMIDEX libodxelo_db.sl.

OPENTURBO CONFIG File

At the core of the OPENTURBO suite is the CONFIG file. The CONFIG File controls and manages the entire OPENTURBO Computing Environment. It allows you to customize replication parameters such as OS type, RDBMS type, database server, access methods, and update options. It is the most critical file and should be centrally and securely managed. Any improper changes to the CONFIG File can potentially damage the integrity of your databases and change your applications behaviors.

Critical Rules Before You Start

1. Preset OPENTURBO configuration files to the systems updating a centralized server. The OPENTURBO CONFIG file must be present on **ALL** systems using OPENTURBO. For example, if your environment uses a number of application servers that update a centralized database server, a copy of the CONFIG file must be present on each application server running OPENTURBO.

You **MUST** use environment variable `OT_CONFIG` to redirect the CONFIG file name. For example on most UNIX environments:

```
export OT_CONFIG=/pub/db/conf1
```

On HP 3000e systems, there are two functionally equivalent ways of setting the CONFIG file name:

```
# Set the OT_CONFIG variable
:setvar OT_CONFIG 'conf1'

# Or Alternatively alias 'conf1' to the 'config' file name.
: file CONFIG=conf1
```

2. All databases that the legacy application uses (including the ones will not to be synchronized) must be configured in the configuration file. These configurations include:
 - a. Details about the local [and remote] databases
 - b. If a database is to act as a mirror and must be synchronized
 - c. If an application calls an undefined database through the OPENTURBO library, an error of ****Error: "DBOPEN error -11"**** will occur
3. If you are planning to do reverse data synchronization (from the RDBMS to HPe3000 where the TurboIMAGE resides, the following variables need to be specified in the configuration file: `TI_DUALMODE_HOST` `TI_DUALMODE_SERVICE` `TI_DUALMODE_PGM`.
4. Environment variables or the passing parameters cannot be used in the configuration files.

5. The CONFIG file must be a non-numbered text file.
6. Comments must start at the beginning of a line. For example:

```
// VALID COMMENT FORMAT
OT_WRITE_CACHE = WRITE_CACHE_LOCATION // INVALID COMMENTS
```

CONFIG File Command Syntax

```
// Denotes the comment lines
{} Begin and End of a Database Definition Block
[] Begin and End of a Dataset Definition Block
```

Database Level Configuration

Field	Description
OT_TI_DBNAME	<p>Contains the fully qualified TurboIMAGE database name. OT_TI_DBNAME must match the database name stored in the source (or local) database. In addition, it must be set outside of a database definition block and set before database name alias. For example:</p> <pre>OT_TI_DBNAME = DBNAME.GROUP.ACCOUNT</pre> <p>You can have as many aliases as you want, but avoid making the same alias for different TurboIMAGE databases in the same CONFIG file. In all cases, the first found alias for a particular DB is used.</p>
OT_ROOT_FILE	<p>Points to the TIFILE root-file created by TILOAD. No environment variable is allowed in the file name. For example, if a name \$HOME/db/DBNAMEti.ti is entered an error: 'TIFile cannot be found' is returned.</p>
OT_RESERVE_WORD_FILE	<p>Points to the file that contains relational database reserve words and conversion suffix. Default for HP 3000 is RESERVE.ORACLE.IMAXSOFT. For Unix servers, the default is:</p> <pre>/opt/imaxsoft/OPENTURBOx.x/eloquence/conf/RESERVE.ORACLE</pre>
OT_ERROR_FILE	<p>Points to the file that contains the TurboIMAGE errors and messages for OPENTURBO. Default for HP3000 OTERROR.ORACLE.IMAXSOFT. For Unix servers, the default is:</p> <pre>/opt/imaxsoft/OPENTURBOx.x/eloquence/conf/OTERROR.ORACLE</pre>
OT_HOST	<p>Points to the application server or the machine where your Eloquence (or RDBMS) server is located. It is used when doing synchronization from HPe3000 to HP-UX. Field ignored when synchronization from the HP-UX to HP3000e direction.</p>

OT_SERVICE	<p>Directs OPENTURBO to the appropriate listener daemon on the OT_HOST. It is used when doing synchronization from HPe3000 to HP-UX. Field ignored when synchronization from the HP-UX to HP3000e direction.</p>
OT_DB_RDBMS	<p>Hex combination of OS and RDBMS. For example, ELOQUENCE on HP-UX takes the HEX value of 0x0200 (DB_TYPE_HPUX) and adds it to 0x0009 (DB_TYPE_ELOQUENCE) resulting in 0x0209.</p> <p><i>Note that Eloquence = ImageDB on non HP3000 platforms</i></p> <p>Operating System:</p> <pre>#define DB_TYPE_MPEXL 0x0100 #define DB_TYPE_HPUX 0x0200 /* all UNIX */ #define DB_TYPE_MSNT 0x0300 /* all INTEL */</pre> <p>Database:</p> <pre>#define DB_TYPE_ORACLE 0x0003 #define DB_TYPE_SQLSVR 0x0004 #define DB_TYPE_ELOQUENCE 0x0009 #define DB_TYPE_IMAGE 0x000a #define DB_TYPE_DB2 0x0010</pre> <p>Some common combinations:</p> <p>ORACLE on HP-UX: 0x0203 (or 515 in decimal form) ORACLE on INTEL-LINUX: 0x0303 (or 772) SQL_SERVER on Windows: 0x0304 (or 771) ELOQUENCE on HP-UX: 0x0209 (or 521) DB2 on HP-UX: 0x210 (or 528)</p>
OT_RDB_LOGON	<p>The IMAGE database has a table of 64 customizable access roles. By setting the OT_RDB_LOGON to an IMAGE role, OPENTURBO will create a single general access role based on the access control of the given role. The format of this field is:</p> <p><eloquence fully qualified name>/<encrypted password></p> <p><i>If OT_RDB_LOGON is set, then OT_RDB_OWNER must also be set.</i></p>
OT_RDB_OWNER	<p>The format of this field is:</p> <p><eloquence fully qualified name></p> <p>For Eloquence, OT_RDB_OWNER is the database name stored in the Eloquence instance. For example, if the database name is DB1.GROUP.ACCOUNT then OT_RDB_OWNER=DB1. OPENTURBO will replicate DB1.GROUP.ACCOUNT in IMAGE to DBNAME in Eloquence.</p> <p><i>If OT_RDB_LOGON is set, then OT_RDB_OWNER must also be set.</i></p>

<p>OT_SDK_SERVER_PGM</p>	<p>Contains the program name of OPENTURBO Core Server Program. By default the OPENTURBO uses:</p> <pre>/opt/imaxsoft/OPENTURBOx.x/eloquence/bin/dmdrv</pre> <p>This field only applies to OPENTURBO Client-Server mode.</p>																																
<p>OT_CIUUPDATE</p>	<p>In IMAGE, keys update (applies only to detail dataset) is not allowed unless CIUPDATE is specified. Set OT_CIUUPDATE = ON to allow critical item updates.</p> <p>IF you use CIUPDATE, you also have to set the same option on iMaxSoft's DOOR (Data Object Open Replication) product that OPENTURBO uses to replicate databases:</p> <ol style="list-style-type: none"> 1. Use TILOAD to generate map file TIFILE for main data structure. 2. Unload detail dataset and load into SQL Server with option -c (OTDRV). 3. Generate map file with option -c (DRMAPGEN). 4. Start shooter with option -c (SHOOTOT). 																																
<p>OT_IMAGEMODE</p>	<p>If OT_IMAGEMODE=ON, then only the IMAGE DB is updated. For example, if you run your applications on HP3000, then only the local IMAGE DB is updated. If are running your applications from HP9000, then it uses TI_DUALMODE_HOST, TI_DUALMODE_SERVICE and TI_DUALMODE_PRG to access IMAGE on the remote HP3000.</p>																																
<p>OT_DUALMODE</p>	<p>Different behaviors are set depending on the combination of OT_DUALMODE and OT_IMAGEMODE:</p> <table border="1" data-bbox="537 1367 1430 1766"> <thead> <tr> <th>OS</th> <th>IMAGEMODE</th> <th>DUALMODE</th> <th>Comment</th> </tr> </thead> <tbody> <tr> <td>MPE/XL</td> <td>ON</td> <td>OFF</td> <td>TurboIMAGE only</td> </tr> <tr> <td>MPE/XL</td> <td>-</td> <td>ON</td> <td>DB2/others RDBMS primary TurboIMAGE secondary</td> </tr> <tr> <td>MPE/XL</td> <td>OFF</td> <td>OFF</td> <td>DB2/others RDBMS only</td> </tr> <tr> <td>HP-UX</td> <td>ON</td> <td>OFF</td> <td>TurboIMAGE only</td> </tr> <tr> <td>HP-UX</td> <td>-</td> <td>ON</td> <td>TurboIMAGE primary DB2/others RDBMS secondary</td> </tr> <tr> <td>HP-UX</td> <td>OFF</td> <td>ON</td> <td>DB2/others RDBMS primary TurboIMAGE secondary</td> </tr> <tr> <td>HP-UX</td> <td>OFF</td> <td>OFF</td> <td>DB2/others RDBMS only</td> </tr> </tbody> </table> <p>For unidirectional data replication and synchronization across an IMAGE database and Eloquence (see figure 2 & 3 above) set</p>	OS	IMAGEMODE	DUALMODE	Comment	MPE/XL	ON	OFF	TurboIMAGE only	MPE/XL	-	ON	DB2/others RDBMS primary TurboIMAGE secondary	MPE/XL	OFF	OFF	DB2/others RDBMS only	HP-UX	ON	OFF	TurboIMAGE only	HP-UX	-	ON	TurboIMAGE primary DB2/others RDBMS secondary	HP-UX	OFF	ON	DB2/others RDBMS primary TurboIMAGE secondary	HP-UX	OFF	OFF	DB2/others RDBMS only
OS	IMAGEMODE	DUALMODE	Comment																														
MPE/XL	ON	OFF	TurboIMAGE only																														
MPE/XL	-	ON	DB2/others RDBMS primary TurboIMAGE secondary																														
MPE/XL	OFF	OFF	DB2/others RDBMS only																														
HP-UX	ON	OFF	TurboIMAGE only																														
HP-UX	-	ON	TurboIMAGE primary DB2/others RDBMS secondary																														
HP-UX	OFF	ON	DB2/others RDBMS primary TurboIMAGE secondary																														
HP-UX	OFF	OFF	DB2/others RDBMS only																														

	<p>OT_DUALMODE to '2PC'. If replication is from HP-UX to HPe3000, then TI_DUALMODE_HOST, TI_DUALMODE_SERVICE, TI_DUALMODE_PRG variables must also be set.</p>
<p>TI_DUALMODE_HOST , TI_DUALMODE_SERVICE , TI_DUALMODE_PGM</p>	<p>Used to connect to TurboIMAGE on HP/3000 from your HP-UX applications. They are used ONLY used during synchronization from HP-UX to HP e3000. The fields are ignored (and can be omitted) when synchronizing from HP e3000 to HP-UX 9000.</p> <p>NOTE: You must start the Listener JLISTNER.PUB.IMAXSOFT from the HP/3000 before running your OPENTURBO applications and utilities on HP-UX in DUAL-MODE or OT_IMAGEMODE.</p>
<p>OT_TRX_THRESHOLD</p>	<p>Reports performance bottlenecks where a SQL transaction executing time exceeds OT_TRX_THRESHOLD limit. By default the threshold is 10 seconds. OPENTURBO logs any execution that exceeds threshold limit and reports it in performance analysis.</p> <p>Your programs must linked with OTXLDBG (HP e3000) or libotdbg (HP-UX) libraries, and OPENTURBO debugging facility must also be set either 'SETVAR LTDBG7 1' or 'export LTDBG7=1'.</p> <p>For most applications, setting OT_TRX_TRHESHOLD=2 is sufficient for detecting potential performance bottlenecks.</p>
<p>OT_WRITE_CACHE</p>	<p>Eloquence uses the OT_WRITE_CACHE only when DUALMODE=2PC. OPENTURBO will synchronize both an OT cache buffer and the primary database on the local and remote systems.</p> <p>There are 3 modes used for synchronization</p> <p>STATEMENT_COMMIT Data is updated to (local) primary database and cache buffer in an ATOMIC fashion (2PC), then the data is subsequently sent to remote side asynchronously <i>after each statement (INSERT, DELETE, UPDATE, etc)</i>. OPENTURBO will perform sync-point check when the CACHE BLOCK (currently set to 30K) is full.</p> <p>BULK_COMMIT (Not Fully Tested) Data is updated to (local) primary database and cache buffer in an ATOMIC fashion (2PC). <i>When the CACHE BLOCK (currently set to 30k) is full</i>, OPENTURBO will perform sync-point check to assure that the transactions are replicated to remote database successfully.</p> <p>OFF Waits for COMMIT Acknowledgement.</p>

OT_READ_CACHE	Only applicable when OT_DUALMODE=2PC and OT_WRITE_CACHE=OFF. This results OT caching of cursors during reads.
OT_LOCKWAIT_CYCLE , OT_NETWORK_COMPRESS , OT_LOCKCOVERAGE , OT_DBLOCK_CONTROL	N/A for Eloquence

Dataset Level Configuration

None of the Dataset Level configurations are applicable for Eloquence. Therefore the only applicable field is

Field	Description
OT_NOOPT	Set to ON since none of the Dataset Level configurations are applicable for Eloquence.

ENCRYPTING/DECRYPTING Passwords

Passwords must be encrypted when using OPENTURBO. OPENTURBO provides a pair of functions to be used for passwords.

Syntax ENCRYPT.BIN.IMAXSOFT <decrypted password>

Example: :ENCRYPT.BIN.IMAXSOFT WRITER
 OPENTURBO ENCRYPT <A.01.00> iMaxsoft Corp. Copyright 2002.
 Encrypted Password = [RMDOZM]

Syntax DECRYPT.BIN.IMAXSOFT <encrypted password>

Example: :DECRYPT.BIN.IMAXSOFT RMDOZM
 OPENTURBO DECRYPT <A.01.00> iMaxsoft Corp. Copyright 2002.
 Original Password = [WRITER]

OPENTURBO Synchronization Modes

Synchronization by default consists of 2 data sources, one on local server and one on remote server. In order to access remote server you must start listener on the remote server. The database level configuration `OT_DUALMODE` must be set to '2PC' for unidirectional data replication and synchronization across an IMAGE database and Eloquence (see figure 2 & 3 above). If replication is from HP-UX to HPe3000, then `TI_DUALMODE_HOST`, `TI_DUALMODE_SERVICE`, `TI_DUALMODE_PRG` variables must also be set.

Starting the OPENTURBO Listener

You must start this daemon process on your HP9000 regardless if you access the database locally or remotely. The listener program accepts DBOPEN requests from your application programs, and then spawns the OPENTURBO server process DBSVR, which performs all subsequent database access calls. The listener is also responsible for OPENTURBO recovery; if DBSVR aborts abnormally, the listener will make sure all dangling database objects that are created by the DBSVR are clean-up properly.

You must provide an unused server port for listener to use, check `/etc/services` file and find an open number: the range is from 1 through 32768. It is highly recommended that you add the newly assigned entry into `/etc/services` file for ease of control.

Here is an example entry:

```
OTB 32608/tcp otb # For OPENTURBO Listener
```

In the client, the listener connection control data is stored in the `CONFIG` file, `OT_HOST`, `OT_SERVICE`, `OT_OS_RDBMS`, `OT_RDB_LOGON`. `OT_SDK_SERVER_PRG` are used to connect to the target host machine, to talk to the listener, to spawn the server program, and to connect to the proper database via proper database logon.

Note: the `OT_RDB_LOGON` is used only when your program login is as the creator of the TurboIMAGE and use semicolon as the password. Otherwise, the DBOPEN password is mapped to its corresponding DB user.

Sample script to start a listener without a configuration file:

```
export LTDBG17=0
export LTDBG18=0
export LTDBG19=0
export LTDBG27=0
export LTDBG28=0
export LTDBGOUT=-
/opt/imaxsoft/OPENTURBO3.7/eloquence/bin/listner 32601
```

You need to start the listener using the proper DB and OPENTURBO setup credentials, which means all environment variables, access paths, library paths, etc. must point to the proper location of your database, OPENTURBO programs, MF-COBOL, and dynamic libraries.

Do not turn on OPENTURBO debugging from listener level, but turn it on through `DBCONTROL`; if you turn on OPENTURBO debugging at listener level, the `LTDBGOUT` file will logs all clients' info. There is no way to isolate individual client trace. This feature is used only in the development environment; you can assign each programmer a listener, then he or she controls his/her own environment. One client per listener; this is the easy way to turn on OPENTURBO trace.

For Eloquence only:

The server program is `DOORELO_dmdrv3` in our standard directory `/opt/imaxsoft/OPENTURBO3.7/eloquence/bin` and it uses libraries in `/opt/imaxsoft/OPENTURBO3.7/eloquence/lib`. The bridge library `libelo3k.sl` must link to the proper eloquence library first and you need to do it only once after installation. The command is:

```
ln -s /opt/eloquence6/lib/pa20_32/libimage3k.sl libelo3k.sl
```

In the listener script you must set `shlib_path` properly in order for program `DOORELO_dmdrv3` to access necessary libraries in `/opt/imaxsoft/OPENTURBO3.7/eloquence/lib`.

Start listener with configuration file:

```
$/opt/imaxsoft/OPENTURBO3.7/eloquence/bin/listner ^conf32601
```

Configuration `conf32601` file format:

```
#####  
# Copyright (c) iMaxsoft Corp. 2006 All Rights Reserved.  
#  
# DEBUG MASK EXAMPLES:  
# 0 3 4 7 8 11 12 15 16 19 20 23 24 27 28 31  
# 0000 0000 0000 0000 0000 0000 0000 0000 0000  
#  
# APP1 40000000 1  
# APP2 20000000 2  
# LAN 00004000 17  
# SOCK 00002000 18  
# NIPC 00001000 19  
# SQLX2 00000010 27  
# SQLX1 00000008 28  
#  
#####  
[ GLOBAL ]  
SERVICE = 32601  
HOME = /tmp  
DBGOUT = /tmp/ltDBGOUT32601  
# DBGMASK = 60003000 APP1 + APP2 + NIPC + SOCK  
DBGMASK = 00000000  
SERVER = 207.0.0.1  
PARM =
```

```

STANDBY = Q1
NICE = /opt/imaxsoft/OPENTURBO3.7/eloquence/config/mynice

[ Q1 ]
SERVER = /opt/imaxsoft/OPENTURBO3.7/eloquence/bin/mypgml
MIN = 2
MAX = 10

```

The configuration has two parts: the GLOBAL and the QUEUE for standby processes.

Global Area

SERVICE	Port of the listener
HOME	Listener's home directory; this where core dump is placed for your application on HP-UX
DBGOUT	Listener's debug output file name
DBGMASK	32-bit mask, from left to right, each bit controls one level of debug trace. The leftmost bit is mapped to LTDBG1, and so on. FFFFFFFF turns all on and 00000000 turns all off.
PARM	SERVER: The default is loop-back 127.0.0.1 or local host name. Mimics HP3000 MPE/XL run command's parm= option
STANDBY	list of standby queue names that are defined in the QUEUE block below; you can specify multiple names here separated by comma, i.e. Q1, Q2, Q3
NICE	Specifies the file name that contains a list of programs that need to be spawned at different nice value. Unless the listener has SU capability, all nice value must be equal or less than listener's. If nice failed due to lack of capability, then the spawned process is set to the same value as the listener. The file format is: /opt/imaxsoft/OPENTURBO3.7/eloquence/myprog1 30 /opt/imaxsoft/OPENTURBO3.7/eloquence/myprog2 19 /opt/imaxsoft/OPENTURBO3.7/eloquence/myprog3 20 /opt/imaxsoft/OPENTURBO3.7/eloquence/myprog4 -10

Queue Area

SERVER	Name of the standby program
MIN	Minimum number of standby programs to be started when listener is initiated
MAX	number of standby programs that are allowed

From HPe3000 to HP-UX

1. Set the FILE equation to CONFIG file. For example:

```
FILE CONFIG=MYCONFIG.GROUP.ACCT
```

2. Start the Listener on the HP-UX.

3. Set values for CONFIG files.

```
OT_TI_DBNAME = DBNAME.GROUP.ACCOUNT
DBNAME.GROUP.ACCOUNT {
DBNAME.GROUP {
DBNAME {
    OT_IMAGEMODE           = OFF
    OT_ROOT_FILE           = DBNAMEti.ti
    OT_RESERVE_WORD_FILE   = RESERVE.ORACLE
    OT_ERROR_FILE          = OTERROR.ORACLE
    OT_HOST                 = 10.1.1.8
    OT_SERVICE              = 32600
    OT_OS_RDBMS             = 521
    OT_RDB_LOGON            = DBNAME_GROUP_ACCOUNT/YWIVHZ
    OT_SDK_SERVER_PGM       = /opt/imaxsoft/OPENTURBOx.x/eloquence/bin/dmdrv
    OT_CIUUPDATE            = ON
    OT_DUALMODE             = 2PC
    OT_TRX_THRESHOLD        = 2
    OT_WRITE_CACHE          = BULK_COMMIT
    OT_READ_CACHE           = ON
OT_DETAILSETNAME = @ [
    OT_NOOPT = ON
]
}
```

From HP-UX to HPe3000

1. Make sure Eloquence and OPENTURBO is set in the \$SHLIB_PATH (for PA-RISC systems) library path
2. Set the OT_CONFIG variable to point to the CONFIG
3. Start the Listener on the HPe3000
4. Set values for the CONFIG files.

```
OT_TI_DBNAME = DBNAME.GROUP.ACCOUNT
DBNAME.GROUP.ACCOUNT {
DBNAME.GROUP {
DBNAME {
    OT_IMAGEMODE           = OFF
    OT_ROOT_FILE           = DBNAMEti.ti
```

```

OT_RESERVE_WORD_FILE = RESERVE.ORACLE
OT_ERROR_FILE        = OTERROR.ORACLE
OT_HOST              = 10.1.1.8
OT_SERVICE           = 32600
OT_OS_RDBMS         = 521
OT_RDB_LOGON        = DBNAME_GROUP_ACCOUNT/YWIVHZ
OT_SDK_SERVER_PGM    = /opt/imaxsoft/OPENTURBOx.x/eloquence/bin/dmdrv
OT_CIUUPDATE        = ON
OT_DUALMODE         = 2PC
TI_DUALMODE_HOST    = 10.1.1.66
TI_DUALMODE_SERVICE = 32602
TI_DUALMODE_PGM     = DMDRV.BIN.IMAXSOFT
OT_TRX_THRESHOLD    = 2
OT_WRITE_CACHE      = BULK_COMMIT
OT_READ_CACHE       = ON
OT_DETAILSETNAME = @ [
  OT_NOOPT = ON
]
}

```

5. Set IMAGE passwords (for access to SUPRTOOL/Query3k). Since the listener on the HP3000 is not in the same group.acct as the IMAGE database, it is not possible to access the IMAGE DB without logging in. This can be done with the following code snippet:

```

export EQ_DBPASSWORD=USER
export EQ_DBUSER=MYPASSWORD
export EQ3K_<dbname>=<dbname>.<group>.<account>

```

Troubleshooting

Here is a list of some common issues:

1. Did you set the correct library path in your environment?

The correct library path must be set so that your application can find both the Eloquence and OPENTURBO libraries. In addition the OPENTURBO library path **MUST** come before the Eloquence6 path. For example:

```
export SHLIB_PATH=/opt/imaxsoft/OPENTURBO3.7/lib:/opt/eloquence6/lib/pa_32
```

2. Did you specify an OT_ERROR_FILE?

A valid OT_ERROR_FILE must exist prior to synchronization.

3. Did you specify an OT_RESERVE_WORD_FILE?

A valid OT_RESERVE_WORD_FILE must exist prior to synchronization.

4. Did you start the listener on the remote server?

Communication on the remote server is handled by the listener. If the listener has not been started then there will be errors communicating with the remote server.

5. *Do you have a valid license?*

HP-UX

Run /opt/leetech/bin/ltvalida
Check product number 1006 and 1007

HPe3000

Run OTVALIDA.PUB.IMAXSOFT
Check product number 2006 and 1688

OPENTURBO Utilities

CROSSREF File

The CROSSREF File is the Cross Reference File used to:

1. Change column name
2. Change type default override
3. Rename of default OPENTURBO internal key column name
4. re-position OPENTURBO internal key columns.

In addition to these core features, CROSSREF also allows you to exclude records based on DATA SEARCH PATTERN exclusion rules and to convert fields from NULLs to BLANK or BLANKS to NULL

Special Notes

iMaxsoft has the option of adding additional columns known IMAXSOFT13_PATH_nn and IMAXSOFT13_SEQ_NO in migrated database to help preserve the IMAGE linked-list internal data structure. If needed (though not recommended), these fields can be used emulated the sequence in the IMAGE database if the sequence or traversal of data is important to your application. The prefix of IMAXSOFT13 was chosen as the default name for this field to avoid name conflicts but it is possible to choose a custom prefix by setting UKEY_NAME and PATH_PREFIX in the CROSSREF file.

CROSSREF Syntax

The CROSSREF syntax consists of 3 columns:

Action	ITEM_NAME	ChangeTo
Action to be performed	The original IMAGE column name.	Value or column type that ITEM_NAME will be converted to.

The values of ITEM_NAME are in IMAGE format where the *usage of underscore is NOT allowed*. For example TOTAL-AMT is valid whereas TOTAL_AMT not.

Types of CROSSREF actions that can be performed include:

Action	Description
COLNAME_CHANGE	Replaces ITEM_NAME strings that match the pattern specified with the ChangeTo string. Rules of ITEM_NAME patterns are: 1) If iMaxsoft encounters a RESERVED word for column name, it appends a default suffix in order to make the name legal. COLNAME_CHANGE allows you

	<p>to append a user-defined suffix for RESERVED words.</p> <p>2) '@' character is used as a wildcard. @ is allowed at beginning or at end but not in the middle of the string.</p> <p>For example, @STRING, STRING@, and @STRING@ are valid search strings but STR@ING is invalid.</p> <p>3) You may concatenate dataset name DATASET.ITEM_NAME, then the name change applies to the specific dataset only</p> <p>NOTE: CROSSREF actions are order specific. It is important to put the more restricted rule first in order to achieve the intended result.</p>
COLTYPE_CHANGE	<p>Changes OPENTURBO default data type. Valid changes include:</p> <p>X_TO_BINARY: VARCHAR to BINARY X_TO_NUMBER: VARCHAR to NUMBER CHAR: VARCHAR to CHAR</p> <p>NOTE: 1. Length cannot be changed. 2. CROSSREF is order specific. If two commands change the same field, then the latter change overwrites the previous change.</p>
OWNER	<p>OPENTURBO will use this user defined owner name as the sole owner for all TurboIMAGE databases referencing this CROSSREF file. The ITEM_NAME is unused in this action and is reserved for future use. The string 'NA' must be entered in the ITEM_NAME column.</p> <p>NOTE The same owner name must be properly set in the run-time CONFIG file for database access.</p>

Sample CROSSREF Entries

Action	ITEM_NAME	ChangeTo	Description
COLTYPE_CHANGE	TOTAL-AMT	X_TO_BINARY	Change TOTAL-AMT column type from VARCHAR to binary
COLNAME_CHANGE	@#@	_NBR	Replace all columns with '#' with '_NBR' in column name
COLTYPE_CHANGE	YMD@	X_TO_NUMBER	Change all columns with the pattern YMD@ from VARCHAR to number

COLTYPE_CHANGE	@	CHAR	Change all columns from VARCHAR to CHAR
UKEY_NAME	IMAXSOFT13_SEQ_NO	IMAGE_RECNR	Changes column name to IMAGE_RECNR

HP3000 Sample Script

You need to use **TILOADAM** to generating your TIFILE, since **TILOAD** doesn't support **-e** option.

```

setvar ltdbg1 0
setvar ltdbg2 0
setvar ltdbg3 0
setvar ltdbg4 0
setvar ltdbg6 0
setvar ltdbgout '$stdlist'
echo
echo 1) use tiloadam, not tiload for TIFile creation, tiload doesn't
echo      support -e CROSSREF file
echo 2) CROSSREF file is crossref.config
echo 3) Must use crossref.config CROSSREF for both tiloadam and otdrv60
echo
input NAME=ot_go;prompt="OK to proceed?"
echo
echo ---- create inventti.ti
echo
tiloadam.bin;info='-d invent.data.sampledb -t inventti.ti &
-r reserve.db2 -e crossref.config'
echo
echo ---- unload and direct load invent18 (SI dataset)
echo
otdrv60.bin;info=' &
-dinvent.data.sampledb -t20 &
-ti inventti.ti &
-v reserve.db2 &
-s si &
-recnum -b DB2 -e crossref.config &
-g 207.92.64.9:32688:INVENT:INVENT:/opt/imaxsoft/db2/otldr2:&
2000:/tmp/'

```

Use **TILOADAM** to generate your TIFILE instead of **TILOAD**. **TILOADAM** allows you to specify a **CROSSREF** file using the **-e** option. The proper loading script and data file will be created according to your **CROSSREF** using **otdrv**. Sample run result:

```

LT928A: MGR. IMS0100(81): gobdir

1) use tiloadam, not tiload for TIFile creation, tiload doesn't
   support -e CROSSREF file
2) CROSSREF file is crossref.config
3) Must use crossref.config CROSSREF for both tiloadam and otdrv60

OK to proceed?

```

```

---- create inventti.ti

OPENTURBO TILoad <A.01.04> iMaxsoft Corp. Copyright 2002.

IMAXSOFT/CSF IMAXSOFT Corp. Copyright 2002-2004, All Rights Reserved.
License No. 000000 DEMO [2005/08/05]

Process Start: 2005-07-16 09:28:50
Process Stop: 2005-07-16 09:28:55

---- unload and direct load invent18 (SI dataset)

OPENTURBO*Pro OTDRVEZ <A.02.01> iMaxsoft Corp. Copyright 2003.

OPENTURBO IMAXSOFT Corp. Copyright 2002-2004, All Rights Reserved.
License No. 000000 DEMO [2005/08/05]

OPENTURBO TIUNLOAD Dataset [SI]
UNLOAD Data File [invent18.OTDATA.IMS0100]
UNLOAD Script File [invent18.OTSCRIPT.IMS0100]
COPY of the Dataset [invent18.OTCOPY.IMS0100]
WORKING File [invent18.OTWORK.IMS0100]

OPENTURBO OTDRV Params:
-dinvent.data.sampledb
-tiinventti.ti
-vreserve.db2
-t20
-rinvent18.OTDATA.IMS0100
-linvent18.OTSCRIPT.IMS0100
-oinvent18.OTCOPY.IMS0100
-winvent18.OTWORK.IMS0100
-sSI
-bDB2
-ecrossref.config
EXCEPTION REPORT = invent18.OTEXCEPT.IMS0100
OTEDIT_OFF
CHRONOLOGICAL_OFF
ASCII_TRANSFER
NON_EXCLUSIVE_MODE
USE_TI_RECNUM and USE_FOR_DOOR
DB2
NORMAL_LOAD_SCRIPT
KEEP_WHITESPACE
GO_STRAIGHT_TO_ORACLE
ORACLE Host =207.92.64.9
OPENTURBO Port =32688
ORACLE User =INVENT
ORACLE Password =INVENT
OPENTURBO Loader=/opt/imaxsoft/db2/otldr2
LOGFILE dir =/tmp/
COMMIT Count =2000

DB2: TOTLen = 4076
DB2: RDBFileRecSize = 4076
DB2: MEDIARECSize = 508

RDBFileRecSize = 4076
MEDIARECSize = 1016
OTGenDetailOutFile: Set(18) HW(124487:1) Row(47690) Reject(0)

Process Start: 2005-07-16 09:28:58
Process Stop: 2005-07-16 10:01:22
Process Elapse: 00:32:24
LT928A: MGR.IMS0100(82):

```

lanutil

Lanutil allows you to view any process that is currently connected to your ORACLE database locally or remotely through OPENTURBO. This utility can be run on the net, as long as you specify the host name and service port number; it directs you all the way to the specific listener and reports the status.

Lanutil is also used to gracefully shut down the listener, command: STOPALL.

```
LANUTIL (A.06.00.00) iMAXSOFT Corp. Copyright 1993-2002, All Rights Reserved.

HOST:[127.0.0.1] APPLICATION:[32601]

Commands: LIST      - shows all connected users.
           KILL id  - kills the specified user.
           STOPALL - terminates listener and all users.
           HOST id  - sets to new host node name.
           APPL id  - sets to new application name.
           SETQ qname #servers
                   - sets # of standby servers for a queue
           EXIT     - ends the LANUTIL.

LANUTIL>>
```

1. LIST command, shows all server processes that are spawned by the listener running on the HOST [127.0.0.1] and waiting on the SERVICE [32601].
2. KILL id command, kills the specific server process via the ID from the LIST command report.
3. STOPALL, is the best way to gracefully shut down this listener along with associated child processes.
4. HOST id command, reconnects Lanutil to another HOST via either an IP address or a DNS name.
5. APPL id command, reconnects Lanutil to another listener via either a SERVICE number or a SERVICE name.
6. SETQ qname command, OPENTURBO listener supports persistent and parallel stand-by modes. At current release of OPENTURBO, only persistent listener is supported; ignore this command.
7. EXIT command, ends Lanutil program.

listner

The listner daemon process must be started on your HP9000 regardless if you access your database locally or remotely. The listener program accepts DBOPEN requests from your application programs, and then spawns the OPENTURBO server process DBSVR, which performs all subsequent database access calls. The listener is also responsible for OPENTURBO recovery; if DBSVR aborts abnormally, the listener will make sure all dangling database objects that are created by the DBSVR are clean-up properly.

You must provide an unused server port for listener to use, check `/etc/services` file and find an open number: the range is from 1 through 32768. It is highly recommended that you add the newly assigned entry into `/etc/services` file for ease of control. Here is an example entry:

```
OTB 32608/tcp otb # For OPENTURBO Listener
```

In the client, the listener connection control data is stored in the CONFIG file, `OT_HOST`, `OT_SERVICE`, `OT_OS_RDBMS`, `OT_RDB_LOGON`. `OT_SDK_SERVER_PRG` are used to connect to the target host machine, to talk to the listener, to spawn the server program, and to connect to the proper database via proper database logon.

Note: the OT_RDB_LOGON is used only when your program login is as the creator of the TurboIMAGE and use semicolon as the password. Otherwise, the DBOPEN password is mapped to its corresponding database user.

Sample script to start a listener without a configuration file:

```
export LTDBG17=0
export LTDBG18=0
export LTDBG19=0
export LTDBG27=0
export LTDBG28=0
export LTDBGOUT=-
$_OTB_BIN/listner 32601
```

You need to start the listener with super user capability and from the login with proper Eloquence and OPENTURBO setup, which means all environment variables, access paths, library paths, etc. must point to the proper places for Eloquence database and OPENTURBO, MF-COBOL, Eloquence dynamic libraries.

Do not turn on OPENTURBO debugging from listener level, but turn it on through `DBCONTROL`; if you turn on OPENTURBO debugging at listener level, the `LTDBGOUT` file will logs all clients' info. There is no way to isolate individual client trace. This feature is used only in the development environment; you can assign each programmer a listener, then he or she controls his/her own environment. One client per listener; this is the easy way to turn on OPENTURBO trace.

HP-UX Special Features

1. If you name your listener to `listenerND`, then the listener will turn itself into a Non Daemon mode.

2. The client process can pass program name with options. For example:

- If `;shlib=/imaxsoft/pub:/imaxsoft/a` is part of your program name, then `/imaxsoft/pub:/imaxsoft/a` will be inserted by listener to the front of `$SHLIB_PATH`
- If `;cwd=/imaxsoft/tmp` is part of your program name, then listener will change your program home directory to `/imaxsoft/tmp`.

listener Configuration File

The configuration has two parts: the GLOBAL and the QUEUE for standby processes.

Global Definition Area

SERVICE	Port of the listener
HOME	Listener's home directory; this where core dump is placed for your application on HP-UX
DBGOUT	Listener's debug output file name
DBGMASK	A 32-bit mask, from left to right, each bit controls one level of debug trace. The leftmost bit is mapped to LTDBG1, and so on. FFFFFFFF turns all on and 00000000 turns all off.
SERVER	Always loop-back 127.0.0.1 or local host name
PARM	Mimics HP3000 MPE/XL run command's parm= option
STANDBY	List of standby queue names that are defined in the QUEUE block below; you can specify multiple names here separated by comma, i.e. Q1, Q2, Q3
NICE	Specify the file name that contains a list of programs that need to be spawned at different nice value. Unless the listener has SU capability, all nice value must be equal or less than listener's. If nice failed due to lack of capability, then the spawned process is set to the same value as the listener. The file format is: <code>/opt/imaxsoft/OPENTURBO3.7/eloquence/myprog1 30</code> <code>/opt/imaxsoft/OPENTURBO3.7/eloquence/myprog2 19</code>

	/opt/imaxsoft/OPENTURBO3.7/eloquence/myprog3 20 /opt/imaxsoft/OPENTURBO3.7/eloquence/myprog4 -10
--	---

Queue Definition Area

SERVER	Name of the standby program
MIN	Minimum number of standby programs to be started when listener is initiated
MAX	Maximum number of standby programs that is allowed

Sample Configuration File

```
#####
# Copyright (c) iMaxsoft Corp. 2006 All Rights Reserved.
#
# DEBUG MASK EXAMPLES:
# 0 3 4 7 8 11 12 15 16 19 20 23 24 27 28 31
# 0000 0000 0000 0000 0000 0000 0000 0000
#
# APP1 40000000 1
# APP2 20000000 2
# LAN 00004000 17
# SOCK 00002000 18
# NIPC 00001000 19
# SQLX2 00000010 27
# SQLX1 00000008 28
#
#####
[ GLOBAL ]
SERVICE = 32601
HOME = /tmp
DBGOUT = /tmp/ltdbgout32601
# DBGMASK = 60003000 APP1 + APP2 + NIPC + SOCK
DBGMASK = 00000000
SERVER = 207.0.0.1
PARM =
STANDBY = Q1
NICE = /opt/imaxsoft/OPENTURBO3.7/eloquence/config/mynice

[ Q1 ]
SERVER = /opt/imaxsoft/OPENTURBO3.7/eloquence/bin/mypgm1
MIN = 2
MAX = 10
```

otDBUTIL

DBUTIL is subset of TurboIMAGE DBUTIL tool. It lists DBOPEN processes and DBLOCK requests from your Eloquence database to the same TurboIMAGE logical database. This utility can be run on the net and uses the same CONFIG file as other OPENTURBO libraries and utilities. The CONFIG can be re-directed via file equation or environment variable OT_CONFIG.

Example:

```
:FILE CONFIG=CONFIG.LEE.IMAXSOFT
:otDBUTIL

OPENTURBO DBUTIL <A.01.02> iMaxsoft Corp. Copyright 2002.

>>help

EXIT                : Leave the program
SHOW (database-name) USERS: Display all current OPENTURBO users
SHOW (database-name) LOCKS: Display all current OPENTURBO locks

>>show invent users

For database INVENT.DATA.MOULTON

PIN    PATH          EXECUTING PROGRAM          JOBNUM  MODE
=====
2227   1                OPENTURBO DBOPEN          1
2254   1                OPENTURBO DBOPEN          1

>>show invent locks

For database INVENT.DATA.MOULTON

LOCKED ENTITY                PIN          PROGRAM
=====
DATA SET      ITMMAST . . . . .          2227 OPENTURBO DBLOCK
ITMDTL:  CC-PARTNO = MOXIMG14047          2254 OPENTURBO DBLOCK

>>
```

recover, reader

OPENTURBO provides a set of recovery tools for disaster recovery. At the core is the `recover` utility that will read a cache log file and roll back the migration back to the original source database. Any additional transactions that have occurred since the original migration point will be applied to the recovered database.

```
OPENTURBO recover<A.03.07> iMaxsoft Corp. Copyright 2005

usage: prerecov LogfileName Host Port Pgm DBpwdFile
      LogfileName - * for all files from current dir
      DBpwdFile record format - DATABASE,PASSWORD,RealBASE
      * RealBASE is the actual server DBname:
      HP3000 recover to HP9000, it is the Eloquence DBname;
      HP9000 recover to HP3000, it is the IMAGE DBname
```

OPENTURBO also provides a utility, `reader`, to read the Log Files used for disaster recovery.

```
OPENTURBO reader<A.03.07> iMaxsoft Corp. Copyright 2005
usage: reader LogfileName
```

Examples

For a detailed overview and examples on how to use the `recover` functionality please refer to the *Disaster Recovery: Recovering from Cache* section of this manual.

tidrvD02dbg_eloquence, tidrvD02dbg_omnidex

`tidrv` is OPENTURBO testing driver program. As its name implies, `tidrvD02dbg_eloquence` is the `tidrv` version for Eloquence and is the `tidrv` version for Omnidex. `tidrv` is the best tool used to verify data migration results, to perform progressive test, and to conduct performance benchmarking analysis. You can enter TurboIMAGE API calls using its straight-forward syntax and verify the formatted results.

`tidrv` can be run in silent mode, which takes inputs from a command file and reports output to an output file or in interactive mode which uses `SDTIN` and `STDOUT` as the input and output files.

`tidrv` can also be run in OPENTURBO mode as well as in TurboIMAGE mode. In OPENTURBO mode, it accesses Eloquence database either locally or remotely. In TurboIMAGE mode, it can be configured to access TurboIMAGE database locally or remotely. The result formats are identical in both modes and you can ‘diff’ them easily.

Running TIDRV

```
OPENTURBO TIDRV <A.01.04> iMaxsoft Corp. Copyright 2002.
```

```
run tidrv.bin;info='-r -i -o -map -turboimage'  
-rRWFile      OPENTURBO Reserve Word File Name  
-iINPUT       TurboIMAGE Command Input File Name  
-oOUTPUT      Output File Name  
-map          Print TurboIMAGE Schema  
-turboimage   Access TurboIMAGE Directly via OPENTURBO Library
```

Option	Description
-r	OPENTURBO Reserve Word File Name. You may use absolute file <code>\$_OTB_ROOT/conf/RESERVE.ORACLE</code> directly, or copy <code>RESERVE.ORACLE</code> into your login MPE account, or use MPE file equation command <code>:FILE RESERVE.ORACLE=</code> to locate the file, or use <code>-r</code> to identify the <code>RESERVE.ORACLE</code> file.
-map	Prints TurboIMAGE schema
-i	Specifies the input command file name <ul style="list-style-type: none">• HP3000: use <code>\$stdin</code> for interactive mode• HP9000: use <code>-</code> for interactive mode
-o	Specifies the output result file name <ul style="list-style-type: none">• HP3000: Use <code>\$stdlist</code> for terminal output• HP9000: Use <code>-</code> for terminal output.

On HP e3000

- Running `TIDRV;XL="OTXL.A.IMAXSOFT"` accesses remote Eloquence database on HP9000.
- Running `TIDRV;XL="XL.PUB.SYS"` accesses local TurboIMAGE database.
- Running `TIDRV;XL="OTXL.A.IMAXSOFT";INFO="-turboimage"` accesses local TurboIMAGE database via OPENTURBO MPE/XL library

On HP9000

- Running `TIDRV` with `libot.sl` accesses local Eloquence database
- Running `TIDRV -turboimage` with `libot.sl` accesses remote TurboIMAGE database on HP e3000 via OPENTURBO HP-UX library

TIDRV Rules and Syntax

- Use `//` or `/*` for comment line
- Use `COMMENT ON` and `COMMENT OFF` for comment block
- Use `&` at end of each command line for denoting command continuation
- All value must be embedded in single quote `'`
- Use `\` for de-reference special character, such as `\`
- Use `,` for parameters separator
- `PRINT ON` and `PRINT OFF` to turn on and off print-result-to-file option
- `DEFINE` is used to declare variables, currently we only support short and int, which are 16-bit and 32-bit interger
- `REPEAT n`, executes the immediate followed DBCall n times, one call only
- `DEBUGOUT filename`, specifies the remote debugging file name (on HP9000)
- `DEBUGn ON` and `DEBUGn OFF` to turn on and off debugging level from 0 through 31; currently supported levels are:

<code>DEBUG0:</code>	Serious Error (no need to turn on)
<code>DEBUG1:</code>	OPENTURBO core level trace
<code>DEBUG2:</code>	OPENTURBO reserved word
<code>DEBUG3:</code>	OPENTURBO mapped error message (TurboIMAGE)
<code>DEBUG4:</code>	OPENTURBO emulator level trace
<code>DEBUG5:</code>	OPENTURBO client SQL statement and CURSOR POOL trace
<code>DEBUG6:</code>	OPENTURBO DUAL MODE diff results
<code>DEBUG7:</code>	OPENTURBO transaction performance trace
<code>DEBUG13:</code>	OPENTURBO call pattern analyzer
<code>DEBUG17:</code>	Network traffic dump in hex and text
<code>DEBUG18:</code>	Network socket trace
<code>DEBUG19:</code>	Network Net/IPC and TCP/IP trace
<code>DEBUG27:</code>	SQL statement dump

DEBUG28: SQL error analyzer
DEBUG29: malloc() and free() trace

- LOADTI,ti-filename and UNLOADTI (obsolete); loads the OPENTURBO root-file into memory for DBCalls syntax and semantic checking
- USETI,tifile-id; sets the active OPENTURBO root-file (tifile) for subsequent DBCalls syntax and semantic checking; the first LOADTI gets 0 for tifile-id, the second LOADTI gets 1 for tifile-id, and so on till 63
- EXIT ends TIDRV program

TIDRV TurboIMAGE Calls

- 1) DBOPEN , TurboIMAGE_Database_Name; , Password; , Mode
 - o The first DBOPEN's baseID = 0
 - o The second DBOPEN's baseID = 1
 - o The third DBOPEN's baseID = 2
 - o . . . and so on
- 2) DBCLOSE , baseID , Dataset-Name | Dataset-Number | None , Mode
 - o The baseID is the number associated to the DBOPEN
 - o None means nothing in between two commas, such as ,,
 - o The Dataset-Name is 16 characters long or terminated by either blank or semi-colon, such as MEMBERDETL;
 - o The Dataset-Number is number only, such as 24
- 3) DBFIND , baseID , Dataset-Name | Dataset-Number , Mode , ITEM='Item-Name | Item-Number' , ARG=Defined-Variable | 'Value'
 - o The ITEM= is TIDRV's key word and is part of command syntax
 - o The Item-Name is the key item, it can be 16 characters long or terminated by either blank or semi-colon, such as MBRNO
 - o The Item-Number is number only, such as 5
 - o The ARG= is TIDRV's key word and is part of command syntax
 - o The Defined-Variable is declared via TIDRV DEFINE command
 - o The Value can be a true value, value with wildcard, or the standard argument as specified in TurboIMAGE manual page 180
 - o OPENTURBO version A.01.00 and above support all modes but 10, which has been implemented but has not been certified by TPI vendors yet.
- 4) DBGET , baseID , Dataset-Name | Dataset-Number , Mode , LIST='Item-Name List | Item-Number List | Special List' , ARG=Defined-Variable | 'Value'
 - o The LIST= is TIDRV's key word and is part of command syntax
 - o Item-Name List is a list of item names separated by comma, such as MBRNO, MBRNAME, . . .
 - o Item-Number List is a list of item numbers separated by comma, such as 24, 5, . . .
 - o Special List has specific meaning, such as @; means all items, *; means same as previous DBCall List, and so on

- o ARG= is used for Manual Master calculated get by key value or direct get by record number
- 5) DBERROR is part of DBEXPLAIN, use DBEXPLAIN instead
 - 6) DBEXPLAIN , baseID
 - 7) DBCONTROL , baseID , QUALIFIER='' , Mode
 - o The QUALIFIER= is TIDRV's key word and is part of command syntax
 - o Supported modes:
 - Mode 5: Enables the critical item update option
 - Mode 6: Disables the critical item update option
 - Mode 7: Allows Dynamic Multiple Database Transaction
 - Mode 88: Turns ON/OFF a remote debugger level, use the first half-word of QUALIFIER= for the debugging level (0 through 31) and the second half-word of QUALIFIER= for the ON(1) and OFF(0) switch
 - Mode 89: Sets the remote debugger file name, such as QUALIFIER='debugger file name'
 - 8) DBINFO , baseID , QUALIFIER='' , Mode
 - o Refer to TurboIMAGE manual for QUALIFIER=, page 190
 - 9) DBLOCK , baseID , QUALIFIER='' , Mode
 - o Refer to TurboIMAGE manual for QUALIFIER=, page 207 shows the detail format of the lock descriptor
 - o CLIENT-LOCK-MANAGER is responsible for checking and enforcing process related lock rules
 - o SERVER-LOCK-MANAGER is responsible for checking and enforcing cross-process lock rules
 - 10) DBUNLOCK , baseID , None , Mode
 - 11) DBPUT , baseID , Dataset-Name | Dataset-Number , Mode , LIST='' , DATA=''
 - 12) DBDELETE , baseID , Dataset-Name | Dataset-Number , Mode
 - 13) DBUPDATE , baseID , Dataset-Name | Dataset-Number , Mode , LIST='' , DATA=''
 - 14) DBXBEGIN , baseID | baseID:baseID:baseID:. . . , Mode
 - 15) DBXEND , baseID | baseID:baseID:baseID:. . . , Mode
 - 16) DBXUNDO , baseID | baseID:baseID:baseID:. . . , Mode

Examples

```
// HP3000 Run script
    parm xl='NOOT'
    setvar xlflag '!xl'
```

```

setvar ltdbg1 0
setvar ltdbg2 0
setvar ltdbg3 0
setvar ltdbg4 1
setvar ltdbg5 1
setvar ltdbg6 1
setvar ltdbg17 0
setvar ltdbg18 0
setvar ltdbg19 0
setvar ltdbgout '$stdlist'
purge outtrxl.output
if (xlflag = 'OT') then
    file config=config.lee.ims
    run tidrv.bin;xl='otxldb.a,tidrv.a,ltxl.pub.imaxsoft'; &
    info='-iintrxl.lee -outtrxl.output'
else
    run tidrv.bin;xl='tidrv.a,ltxl.pub.imaxsoft'; &
    info='-iintrxl.lee -outtrxl.output'
endif

// 1. OT triggers TIDRV to run in OPENTURBO emulator mode
// 2. NOOT triggers TIDRV to run in normal TurboIMAGE mode
// 3. The input command file name is intrxl
// 4. The output result file name is outtrxl

// HP3000 Input command file:

LOADTI,t1l.ti
// ** DBPUT to Detail Dataset - PURCHASE
DBOPEN,INVENT.DATA.MOULTON;,FAVOR;,3
// DBOPEN,INVENT.DATA.MOULTON;;,1
//
COMMENT ON
DEBUGOUT /opt/imaxsoft/OPENTURBO3.7/eloquence/tmp/lee.dbg
DEBUG19 ON
DEBUG18 ON
DEBUG17 ON
DEBUG27 ON
DEBUG28 ON
COMMENT OFF
//
DBCONTROL,0,,7
//
// =====>> Only for OPENTURBO <<=====
// ==>> TIDRV doesn't support mutiple DBOPEN in TurboIMAGE Mode <<==
//
//*** DBXDBGIN Syntax = DBXBEGIN,BASE=id/BASELIST=id:id:id,Mode ***
//*** DBXEND   Syntax = DBXEND,BASE=id/BASELIST=id:id:id,Mode ***
//*** DBXUNDO  Syntax = DBXUNDO,BASE=id/BASELIST=id:id:id,Mode ***
//
DBXBEGIN,BASE=0,1
//
// Dataitem Level LOCK +++++ CC_PNL_C_PO = X[36]
// ----- EQUAL Confiton -----
DBLOCK,0,ARG='1,36,PURCHASE;,CC-PNL_C_PO=,&
CC_PNL_C_PO_88',5
DBPUT,0,PURCHASE;,1,LIST='@;',DATA='CC_PNL_C_PO_88,POR_KEY_88,88,88,88,&
88,88,88,REV_LOT,REQ_NO,COMMENT_IMS,88,USER_IMS,20020526,888888'
DBUNLOCK,0,,1
// ----- <= and >= No ERROR when DBPUT -----
DBLOCK,0,ARG='2,36,PURCHASE;,CC-PNL_C_PO,<=,&
CC_PNL_C_PO_95',36,PURCHASE;,CC-PNL_C_PO,>=,&
CC_PNL_C_PO_70',5
DBPUT,0,PURCHASE;,1,LIST='@;',DATA='CC_PNL_C_PO_89,POR_KEY_89,89,89,89,&
89,89,89,REV_LOT,REQ_NO,COMMENT_IMS,89,USER_IMS,20020526,999999'
DBUNLOCK,0,,1
//
// Rewind Dataset PURCHASE
// ** Backward Serial DBGET 1 Records from PURCHASE and DBDELETE it
//
// Rewind Dataset PURCHASE and DBGET the Last 4 Records

```

```

//
DBCLOSE,0,PURCHASE,2
DBGGET,0,PURCHASE;,3,LIST='@;',ARG=''
DBGGET,0,PURCHASE;,3,LIST='@;',ARG=''
//
// DBXEND,BASE=0,1
DBXUNDO,BASE=0,1
//
DBCLOSE,0,,1
UNLOADTI
//

// HP3000 Output result file:

=>> Repeat[1] TICCommand[LOADTI,t11.ti]
=>> Repeat[1] TICCommand[DBOPEN,INVENT.DATA.MOULTON;,FAVOR;,3]
DBOPEN,INVENT.DATA.MOULTON;,FAVOR;,3,status[1]=0,status[2]=1
=>> Repeat[1] TICCommand[DBCCONTROL,0,,7]
DBCCONTROL: -----
mode=[7]
status[1]   =[0]
status[2]   =[1]
status[3-4] =[0]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICCommand[DBXBEGIN,BASE=0,1]
DBXBEGIN: -----
mode=[1]
status[1]   =[0]
status[2]   =[1]
status[3-4] =[0]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICCommand[DBLOCK,0,ARG='1,36,PURCHASE;,CC-PNLC-PO,=,CC_PNLC_PO_88
',5]
DBLOCK: -----
base=[INVENT.DATA.MOULTON;]
mode=[5]
status[1]   =[0]
status[2]   =[1]
status[3]   =[0]
status[4]   =[0]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICCommand[DBPUT,0,PURCHASE;,1,LIST='@;',DATA='CC_PNLC_PO_88,POR_KEY
Y_88,88,88,88,88,88,88,REV_LOT,REQ_NO,COMMENT_IMS,88,USER_IMS,20020526,888888']
DBPUT: -----
base=[INVENT.DATA.MOULTON;]
dset=PURCHASE;[15]
mode=[1]
list=@;
DATA BUFFER Begin =====
CC-PNLC-PO      1X36  =>CC_PNLC_PO_88
POR-KEY        1X20  =>POR_KEY_88
VEND-NO        1I2   =>+0000000088
QTY-ORD        1I2   =>+0000000088
DATE-ORD       1I2   =>+0000000088
DATE-PROM      1I2   =>+0000000088
DATE-LAST      1I2   =>+0000000088
REV-LOT        1X10  =>REV_LOT
REQ-NO         1X16  =>REQ_NO
COMMENT        1X30  =>COMMENT_IMS
UNIT-COST      1I4   =>+000000000000000088
USER           1X8   =>USER_IMS
DATE-UPDT      1I2   =>+0020020526
TIME-UPDT      1I2   =>+0000888888
DATA BUFFER End   =====
status[1]   =[0]

```

```

status[2]   =[80]
status[3-4] =[500]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICommand[DBUNLOCK,0,,1]
DBUNLOCK: -----
base=[ INVENT.DATA.MOULTON; ]
mode=[1]
status[1]   =[0]
status[2]   =[1]
status[3-4] =[500]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICommand[DBLOCK,0,ARG='2,36,PURCHASE; ,CC-PNLC-PO,<=,CC_PNLC_PO_95
,36,PURCHASE; ,CC-PNLC-PO,>=,CC_PNLC_PO_70
',5]
DBLOCK: -----
base=[ INVENT.DATA.MOULTON; ]
mode=[5]
status[1]   =[0]
status[2]   =[1]
status[3]   =[0]
status[4]   =[500]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICommand[DBPUT,0,PURCHASE; ,1,LIST='@; ',DATA='CC_PNLC_PO_89,POR_KEY
Y_89,89,89,89,89,89,89,REV_LOT,REQ_NO,COMMENT_IMS,89,USER_IMS,20020526,999999']
DBPUT: -----
base=[ INVENT.DATA.MOULTON; ]
dset=PURCHASE;[15]
mode=[1]
list=@;
DATA BUFFER Begin =====
CC-PNLC-PO      1X36  =>CC_PNLC_PO_89
POR-KEY         1X20  =>POR_KEY_89
VEND-NO        1I2   =>+0000000089
(93/193) Continue?
QTY-ORD        1I2   =>+0000000089
DATE-ORD       1I2   =>+0000000089
DATE-PROM      1I2   =>+0000000089
QTY-RECD       1I2   =>+0000000089
DATE-LAST      1I2   =>+0000000089
REV-LOT        1X10  =>REV_LOT
REQ-NO         1X16  =>REQ_NO
COMMENT        1X30  =>COMMENT_IMS
UNIT-COST      1I4   =>+000000000000000089
USER           1X8   =>USER_IMS
DATE-UPDT     1I2   =>+0020020526
TIME-UPDT     1I2   =>+0000999999
DATA BUFFER End   =====
status[1]   =[0]
status[2]   =[80]
status[3-4] =[502]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICommand[DBUNLOCK,0,,1]
DBUNLOCK: -----
base=[ INVENT.DATA.MOULTON; ]
mode=[1]
status[1]   =[0]
status[2]   =[0]
status[3-4] =[502]
status[5-6] =[0]
status[7-8] =[0]
status[9-10]=[0]
=>> Repeat[1] TICommand[DBCLOSE,0,PURCHASE,2]
DBCLOSE,INVENT.DATA.MOULTON;[0],PURCHASE[15],2,db_status[1]=0

```

```

=>> Repeat[1] TICommand[DBGET,0,PURCHASE;,3,LIST='@;',ARG='']
DBGET: -----
base=[INVENT.DATA.MOULTON;]
dset=PURCHASE;[15]
mode=[3]
list=@;
DATA BUFFER Begin =====
CC-PNLC-PO      1X36  =>CC_PNLC_PO_89
POR-KEY         1X20  =>POR_KEY_89
VEND-NO         1I2   =>+0000000089
QTY-ORD         1I2   =>+0000000089
DATE-ORD        1I2   =>+0000000089
DATE-PROM       1I2   =>+0000000089
QTY-RECD       1I2   =>+0000000089
DATE-LAST       1I2   =>+0000000089
REV-LOT         1X10  =>REV_LOT
REQ-NO          1X16  =>REQ_NO
COMMENT         1X30  =>COMMENT_IMS
UNIT-COST       1I4   =>+00000000000000089
USER           1X8   =>USER_IMS
DATE-UPDT       1I2   =>+0020020526
TIME-UPDT       1I2   =>+0000999999
DATA BUFFER End   =====
status[1]      =[0]
status[2]      =[80]
status[3-4]    =[502]
status[5-6]    =[0]
status[7-8]    =[0]
status[9-10]  =[0]
=>> Repeat[1] TICommand[DBGET,0,PURCHASE;,3,LIST='@;',ARG='']
DBGET: -----
base=[INVENT.DATA.MOULTON;]
dset=PURCHASE;[15]
mode=[3]
list=@;
DATA BUFFER Begin =====
CC-PNLC-PO      1X36  =>CC_PNLC_PO_88
POR-KEY         1X20  =>POR_KEY_88
VEND-NO         1I2   =>+0000000088
QTY-ORD         1I2   =>+0000000088
DATE-ORD        1I2   =>+0000000088
DATE-PROM       1I2   =>+0000000088
QTY-RECD       1I2   =>+0000000088
DATE-LAST       1I2   =>+0000000088
REV-LOT         1X10  =>REV_LOT
REQ-NO          1X16  =>REQ_NO
COMMENT         1X30  =>COMMENT_IMS
UNIT-COST       1I4   =>+00000000000000088
USER           1X8   =>USER_IMS
DATE-UPDT       1I2   =>+0020020526
TIME-UPDT       1I2   =>+0000888888
DATA BUFFER End   =====
status[1]      =[0]
status[2]      =[80]
status[3-4]    =[500]
status[5-6]    =[0]
status[7-8]    =[0]
status[9-10]  =[0]
=>> Repeat[1] TICommand[DBXUNDO,BASE=0,1]
DBXUNDO: -----
mode=[1]
status[1]      =[0]
status[2]      =[80]
status[3-4]    =[500]
status[5-6]    =[0]
status[7-8]    =[0]
status[9-10]  =[0]
=>> Repeat[1] TICommand[DBCLOSE,0,,1]
DBCLOSE,INVENT.DATA.MOULTON;[8224],[-1],1,db_status[1]=0
=>> Repeat[1] TICommand[UNLOADTI]

```

```
// HP9000 Run script:
```

```
export LTDBGOUT=-  
$_OTB_BIN/tidrv -iintrxl -oouttrxl
```

```
// (Input command file: Same as HP3000)
```

```
// (Output result file: Same as HP3000)
```

tiload, tiloadam

Generates the OPENTURBO TurboIMAGE (TI) File for the target database. tiloadam is identical to tiload, except that it *requires* a Cross Reference File.

Syntax

```
tiloadam.bin;info='-d -t -r -v -m -s -g -n -i'  
tiloadam.bin;info='-d -t -r -v -m -s -g -n -i -e '
```

-dDBName	TurboIMAGE Database Name
-tTIFile	OPENTURBO Root-File Name
-rRWFile	OPENTURBO Reserve Word File Name. By default, TI looks for the file "Reserve" in current directory.
-v	OPTIONAL: The command will print on the terminal the TI version and checksum of the specified TIFile and the current IMAGE DB.
-mOUTFile	OPTIONAL: Prints OPENTURBO Root-File Structure to OUTFile
-sOUTFile	OPTIONAL: Generates TurboIMAGE Schema to OUTFile
-gSCHEMA	OPTIONAL: Generates OPENTURBO Root-File from a TurboIMAGE Schema File (SCHEMA). TurboIMAGE name is not allowed to qualify its group and account in the SCHEMA, so the -d DBName is used instead. This process performs TurboIMAGE schema syntax validation first and then creates TIFile.
-nItemList	OPTIONAL: Item List File contains a list of TurboIMAGE X-type Data Item Name(s) to be converted (i.e. X to K and mapped to RAW in Eloquence for NLS or X to double bytes in field)
-iFOSET	OPTIONAL: If set, OPENTURBO will display IMAGE dataset statistics from the TIFile FOSET on stdout.
-eCROSSREF	tiloadam ONLY: Cross Reference File for column name and type default override, rename of default OPENTURBO internal key column name and re-position OPENTURBO internal key columns. For DOOR: you may specify record exclusion rules which are DATA SEARCH PATTERNS, and rule to convert column's NULL byte to BLANK.

NOTES

1. Use `-m` option to get a TurboIMAGE and RDBMS mapping
2. Use `-s` option to regenerate your original TurboIMAGE database schema from an existing TIFILE.
3. Use `-g` option to recreate TIFILE directly from TurboIMAGE database schema.
4. Use `-v` to check version of the TIFILE
5. The database name is stored in the TIFILE for reference and internal use only. It is mainly used for our tool to cross-check the integrity of a TIFILE.
6. OWNER name is not stored since it is a run-time setting in RDBMS. Tables with the same name are differentiated by OWNER at runtime depending on the supplied UNIX login and schema name to the application using OPENTURBO.

Examples

1. Generating TI FILE

```
tiloadam.bin.imaxsoft;info='-dINVENT.DATAE.MOULTON -tINVENTTI &  
-rRESERVE.ORACLE.IMAX'
```

2. Comparing Version of TI File to IMAGE DB

```
tiloadam.bin.imaxsoft;info='-v -tINVENTTI -rRESERVE.ORACLE.IMAX'
```

3. Outputting IMAGE DB to File

```
tiloadam.bin.imaxsoft;info='-tINVENTTI -mOUT1 -rRESERVE.ORACLE.IMAX'
```


Debugging Capabilities

OPENTURBO supports 32 debugging levels starting from 0 through 31. Currently the following levels are defined:

LTDBG0	- OPENTURBO Internal Core ERROR
LTDBG1	- OPENTURBO Core Library Call Trace
LTDBG2	- OPENTURBO Reserved Words
LTDBG3	- OPENTURBO Error Messages
LTDBG4	- OPENTURBO Emulator Call Trace
LTDBG5	- OPENTURBO SDK Call Trace and CURSOR POOL Size
LTDBG6	- OPENTURBO DUAL MODE Diff Results
LTDBG7	- OPENTURBO Transaction Performance Trace
LTDBG13	- TurboIMAGE Call Flow Trace
LTDBG17	- Network Traffic Dump in Hex and Text formats
LTDBG18	- Socket Information
LTDBG19	- Net/IPC Information
LTDBG27	- Dynamic SQL Statement Preparation Trace
LTDBG28	- SQL Statement Execution Error
LTDBG29	- MALLOC, CALLOC and FREE Tracing

Example: Setup debugging from Server LISTENER Process.

The following setup will trigger all server DBSVR processes share the same debugging output file /tmp/2002-07-16.dbg with same debugging levels, 4, 17, 18, 19, 27, 28, and 29.

```
EXPORT LTDBG4=1
EXPORT LTDBG17=1
EXPORT LTDBG18=1
EXPORT LTDBG19=1
EXPORT LTDBG27=1
EXPORT LTDBG28=1
EXPORT LTDBG29=1
EXPORT LTDBGOUT=/tmp/2002-07-16.dbg
./listener OTB
```

Example: Setting up HP e3000 DUAL-MODE Differ Option to Validate Database migration.

You can turn on internal DUAL-MODE option from OPENTURBO HP3000 emulator library by setting OT_DUALMODE = ON in the CONFIG file, and you must set the followings environment variables in order to view diff results:

```
SETVAR LTDBG6 1
SETVAR LTDBGOUT "difffile.group.account"
RUN yourpgm
```

Example: Setting up HP-UX DUAL-MODE Differ Option to Validate Application migration.

DMDRV.PUB.IMAXSOFT is the HP3000 DUAL-MODE driver program that connects to your HP9000 programs for handling TurboIMAGE native APIs remotely.

On HP3000, you must stream the listener job first, JLISTNER.PUB.IMAXSOFT, which spawn child process DMDRV.PUB.IMAXSOFT to handle all remote TurboIMAGE calls from your HP9000 program.

JLISTNER File

```
!job listener,mgr.imaxsoft
!COMMENT
!COMMENT *****
!COMMENT *   IMAXSOFT LISTENER - for DUAL-MODE from HPUX   *
!COMMENT *****
!COMMENT
!file hosts.net.sys=hosts.pub.imaxsoft
!file services.net.sys=services.pub.imaxsoft
!purge DMDIFF.LEE > $NULL
!build DMDIFF.LEE;rec=-80,,f,ascii;disc=100000;msg
!setvar ltdbg6 1
!setvar ltdbgout "DMDIFF.LEE"
!run listner.bin.imaxsoft;info="DBA"
!eoj
```

On HP9000, you must set the followings in the CONFIG file:

```
...
OT_DUALMODE           = ON
TI_DUALMODE_HOST      = 207.92.64.66
TI_DUALMODE_SERVICE   = 32600
TI_DUALMODE_PGM       = DMDRV.BIN.IMAXSOFT
...
```

Disaster Recovery: Recovering from Cache

The OPENTURBO Cache Option is implemented for performance enhancement in a synchronized database environment such as IMAGE and ELOQUENCE. Therefore the disaster recovery plan is focused only **‘recovering from the CACHE when OPENTURBO synchronization fails’**. The CACHE design is based upon deferred UPDATE transactions and asynchronous network transmission. Each UPDATE transaction (DBPUT, DBUPDATE or DBDELETE) is cached first and then transmitted to mirror database asynchronously. If any failures occur then the source cache is used by the OPENTURBO recovery process to recover when the system is back on-line.

Special notes:

- 1) OPENTURBO Recover process recovers OPENTURBO layer failure only.
- 2) OPENTURBO Recover process needs to be used in conjunction with DB Recovery and OS Transaction Recovery.
- 3) OPENTURBO Recover is not a HA (High Availability) solution and is not a DB Recovery replacement.

Assumptions:

- 1) Eloquence cannot be terminated if there are any connected users.
- 2) Eloquence recovers ‘System Abort or Process Abort un-committed transactions’ automatically.
- 3) Network Disconnect – all Eloquence connected processes are hung and we need to kill them manually. The impact is that OPENTURBO server process stops in the middle of DBPUT/DBUPDATE/DBDELETE; in the BULK_COMMIT scenario, not all transmitted updates are committed in Eloquence when network cable is un-plugged.
- 4) SYSTEM ABORT – graceful shutdown has no impact to the whole recovery, it requires no recovery at all. It is very difficult to create a SYSTEM ABORT on both machines; and due to central UPS protection, it is also very difficult to POWER-OFF the machine suddenly. Our best SYSTEM ABORT emulation is either ‘kill OPENTURBO listener’, ‘stop Eloquence’, or ‘un-plug network cable’.
- 5) We are able to cause a HP3000 SYSTEM ABORT by running an older version of GLANCE, the result was IMAGE test database corruption and a restore was required.

STATEMENT_COMMIT Mode Recoverability Matrix

HP3000 is the source machine:

Scenarios (HP3000 to HP9000)	Recoverable
HP3000 SYSTEM ABORT	Partially tested – NO
Process abort via BREAK/ABORT (can be blocked via UDC)	NO
Process ends without proper DBCLOSE()	YES - auto and real-time
Process abort (not via BREAK/ABORT)	YES - auto and real-time

Network Disconnect and Re-connect	YES – auto and real-time
Network Disconnect and Wait for Complete Time-out	YES
HP9000 OPEBTURBO Server Process abort (kill 15)	YES
HP9000 OPENTURBO Server Process abort (kill 9)	YES
HP9000 Eloquence STOP	YES
HP9000 SYSTEM ABORT	YES

HP9000 is the source machine:

Scenarios (HP9000 to HP3000)	Recoverable
HP9000 SYSTEM ABORT	Partially tested – NO
Process abort via kill 9 (can be blocked via SYS_ADMIN)	NO
Process ends without proper DBCLOSE()	YES – auto and real-time
Process abort (kill 15)	YES – auto and real-time
Network Disconnect and Re-connect	YES – auto and real-time
Network Disconnect and Wait for Complete Time-out	YES
HP3000 OPEBTURBO Server Process abort (not via BREAK/ABORT)	YES
HP9000 OPENTURBO Server Process abort (via BREAK/ABORT)	YES
HP3000 SYSTEM ABORT	YES

BULK_COMMIT Mode Recoverability Matrix

HP3000 is the source machine:

Scenarios (HP3000 to HP9000)	Recoverable
HP3000 SYSTEM ABORT	Patial tested - NO
HP9000 Eloquence STOP	NO
Process abort via BREAK/ABORT (can be blocked via UDC)	NO
HP9000 OPENTURBO Server Process abort (kill 9 – can be blocked via SYS_ADMIN)	NO
Process ends without proper DBCLOSE()	YES - auto and real-time
Process abort (not via BREAK/ABORT)	YES - auto and real-time
Network Disconnect and Re-connect	YES – auto and real-time
Network Disconnect and Wait for Complete Time-out	YES
HP9000 OPEBTURBO Server Process abort (kill 15)	YES
HP9000 SYSTEM ABORT	YES

HP9000 is the source machine:

Scenarios (HP9000 to HP3000)	Recoverable
HP9000 SYSTEM ABORT	Partial tested - NO
Process abort via kill 9 (can be blocked via SYS_ADMIN)	NO
HP9000 OPENTURBO Server Process abort (via BREAK/ABORT – can be blocked via UDC)	NO

Process ends without proper DBCLOSE()	YES – auto and real-time
Process abort (kill 15)	YES – auto and real-time
Network Disconnect and Re-connect	YES – auto and real-time
Network Disconnect and Wait for Complete Time-out	YES
HP3000 OPEBTURBO Server Process abort (not via BREAK/ABORT)	YES
HP3000 SYSTEM ABORT	YES

Recovery Process

Currently the OPENTURBO recovery process is based on user group. If there are 10 groups with only 2 performing database updates, then there will only be 2 recovery files created when a failure occurs. This behavior is consistent with the OT caching strategy of only caching DB modifications steps.

Improvements Added in OPENTURBO v3.7

There have been improvements made to disaster recovery and to improve system availability during these situations.

1. On the remote server, OPENTURBO has added a FLAG to ignore errors. On error, OPENTURBO will log the errors but continue to processing change updates from source server.
2. In the event of failure on requests to remote servers, OPENTURBO will switch to LOG_FOR_EVER mode. In this mode, all change requests are logged to local disc file in directory specified by OT_CACHE_LOG_DIR. Each process will have its one unique log file.

When the errors on the remote server are resolved, it is now possible to synchronize the two databases without having to shutting down any users on the main server by running the OPENTURBO recover process. This will cause OPENTURBO to read the local cache log files and send the cached request to the remote server. When OPENTURBO finishes the synchronization process, it will turn off LOG_FOR_EVER mode.

New Requirements for Recovery Improvements

1. The source server now needs large amounts of free disk space to handle potentially huge amounts of cache log data.
2. There will be some performance costs caused by the need to write each request to file during the LOG_FOR_EVER mode. This could have the effect of slowing overall performance. This is a reasonable tradeoff for continued system availability in the event of synchronization failures.

HP-UX as Source Server

Log files on HP-UX will have the name: <AnyAcct>.OTLOG and will be located by default in /tmp. To set a custom directory, set the environment variable OT_CACHE_LOG_DIR to the desired directory. On failure use the following utility:

```
/opt/imaxsoft/OPENTURBO3.7/eloquence/bin/recover
```

```
FUNCTION: recover
```

```
USAGE:      recover LogfileName Host Port Pgm DBpwdFile
            LogfileName - * for all files from current dir
            Host - Remote HPe3000 address
            Port - Remote HPe3000 port
            Pgm - Server program to run on remote HPe3000
            DBpwdFile record format - DATABASE,PASSWORD
```

```
EXAMPLE:   recover ACCT.OTLOG 10.1.2.3 1000 dbprgm passfile
```

HPe3000 as Source Server

Log files on HPe3000 will have the name: OTLOG.<AnyAcct> and will be located by default in the CACHE.IMAXSOFT directory. To set a custom directory, set the environmental variable OT_CACHE_LOG_DIR to the desired directory. On failure, use the recover.bin.imaxsoft for recovery.

```
FUNCTION: recover.bin.imaxsoft
```

```
USAGE:      recover LogfileName Host Port Pgm DBpwdFile
            LogfileName - @ for all files from current group
            Host - Remote HPe3000 address
            Port - Remote HPe3000 port
            Pgm - Server program to run on remote HP-UX
            DBpwdFile record format - DATABASE,PASSWORD
```

```
EXAMPLE:   recover;info="OTLOG.ACCT 10.1.1.1 10000 dbprog passfile"
```

Recovering From Unsupported Failures

Both BREAK/ABORT and kill 9 are not trappable, therefore OPENTURBO has no chance to dump all CACHE to disk for recovery. The same applies to source system abort. Possible resolutions to these cases are to:

1. Re-load target (mirror) database,
2. Restore from backup and re-enter data from previous sync-point,

APPENDIX

Appendix A: OPENTURBO Error Messages

```
400:: GENERAL
0      = SUCCESSFUL EXECUTION - NO ERROR
-1     = NO SUCH DATABASE
-11    = BAD DATABASE NAME OR PRECEDING BLANKS MISSING
-12    = DATABASE MUST BE IN LOGON GROUP AND ACCOUNT
-13    = NOT ALLOWED; MUST BE CREATOR OF ROOT FILE OR DATABASE
-21    = BAD PASSWORD
-22    = MAINTENANCE WORD REQUIRED
-23    = USER (CLASS) LACKS WRITE ACCESS TO DATA SET
-31    = DBGET MODE ILLEGAL FOR DETAIL DATA SET
-32    = UNOBTAINABLE ACCESS MODE
-33    = MODE 7 DIAGNOSIS NOT ALLOWED
-34    = DATABASE MUST BE RECOVERED BEFORE ACCESS IS ALLOWED
-51    = LIST TOO LONG OR NOT PROPERLY TERMINATED
-52    = ITEM SPECIFIED IS NOT AN ACCESSIBLE SEARCH ITEM IN THE SPECIFIED SET
-53    = DBPUT LIST IS MISSING A SEARCH OR SORT ITEM
-82    = CIUPDATE IS SET TO DISALLOWED; CANNOT USE CRITICAL ITEM UPDATE
-90    = ROOT FILE BAD
-99    = UNSUPPORTED FEATURE
-121   = ILLEGAL LOCK DESCRIPTOR COUNT
-123   = ILLEGAL RELATIONAL OPERATOR
-124   = DESCRIPTOR LENGTH ERROR; MUST BE 9 OR MORE
-125   = ILLEGAL SET NAME OR NUMBER IN DESCRIPTOR
-126   = ILLEGAL ITEM NAME OR NUMBER IN DESCRIPTOR
-127   = ILLEGAL ATTEMPT TO LOCK ON A COMPOUND ITEM
-128   = VALUE FIELD TOO SHORT FOR THE ITEM SPECIFIED
-129   = P28 IS LONGEST P-TYPE ITEM THAT CAN BE LOCKED
-130   = ILLEGAL DECIMAL DIGIT IN TYPE 'P' DATA VALUE
-131   = LOWERCASE CHARACTER IN TYPE 'U' DATA VALUE
-132   = ILLEGAL DIGIT IN TYPE 'Z' DATA VALUE
-133   = ILLEGAL SIGN CHARACTER IN TYPE 'Z' DATA VALUE
-134   = TWO LOCK DESCRIPTORS CONFLICT IN SAME REQUEST
-135   = DBLOCK CALLED WITH LOCKS ALREADY IN EFFECT IN THIS JOB/SESSION
-136   = DESCRIPTOR LIST LENGTH EXCEEDS 4094 BYTES
-137   = USER ABOUT TO WAIT FOR SELF
-139   = INVALID NUMBER OF BASE IDS
-140   = BAD BASE ID LIST
-151   = TEXT LENGTH GREATER THAN 512 BYTES
-198   = TOTAL DBOpen COUNT PER USER EXCEEDS LIMIT
-212   = DATABASE CORRUPTION DETECTED
-229   = CANNOT DELETE MANUAL MASTER WITH EMPTY CHAINS
-258   = INVALID ARGUMENT FOR INDEX
-259   = INVALID MODE FOR INDEX
-260   = NO PREVIOUS LIST OF QUALIFIED DATA ENTRIES
-305   = INVALID DATA SET NUMBER
-306   = INVALID DATA SET TYPE
-307   = INVALID RECORD NUMBER FOUND
-420   = FEATURE NOT IMPLEMENTED
-421   = BTE: UNKNOWN QUALIFIER VALUE FOR DBCONTROL MODE 13
-422   = BTE: DATE SET # NOT IN VALID RANGE
-423   = BTE: B-TREE ALREADY EXISTS
-424   = BTE: FAILED TO CREATE B-TREE
-425   = BTE: DB NOT OPENED EXCLUSIVELY
-426   = BTE: B-TREE DOESN'T EXIST
-429   = BTE: DBFIND ARGUMENT VERSION IS BAD
-430   = BTE: DBFIND (mode 4/24) ARGUMENT TYPE IS BAD
```

```

-431 = BTE: DBFIND (mode 4/24) ARGUMENT #1 LENGTH IS BAD
-432 = BTE: WILDCARD NOT ASCII
-433 = BTE: DBFIND (mode 4/24) ARGUMENT #2 LENGTH IS BAD
-434 = DATASET DETAIL INSTEAD OF MASTER
-436 = BTE: FAILED TO EXTRACT DATA FROM ROOT FILE
-437 = BTE: FAILED TO CONVERT @c TO [ ] DBFIND
-439 = BTE: CONVERSION OF KEY FROM EXTERNAL TO INTERNAL FORMAT FAILED
-444 = BTE: DBFIND ON NON-KEY FAILED OF MASTER
-446 = BTE: ARGUMENT 2 SPECIFIED FOR RELOP OF (</<= />=>)
-452 = BTE: KEY LENGTH GREATER THAN 252 BYTES (MAXIMUM INDEX KEY SIZE)
-458 = DBOPEN FAILED. OUT OF DISK SPACE
10 = BEGINNING OF FILE
11 = END OF FILE
12 = DIRECTED BEGINNING OF FILE
13 = DIRECTED END OF FILE
14 = BEGINNING OF CHAIN
15 = END OF CHAIN
16 = THE DATA SET IS FULL
17 = THERE IS NO CHAIN FOR THE SPECIFIED SEARCH ITEM VALUE
18 = BROKEN CHAIN - FORWARD AND BACKWARD POINTERS NOT CONSISTENT
20 = DATABASE CURRENTLY LOCKED SETS OR ENTRIES LOCKED WITHIN DATABASE
22 = DATA SET ALREADY LOCKED
23 = CANNOT LOCK SET DUE TO LOCKED ENTRIES WITHIN IT
24 = ENTRIES CURRENTLY LOCKED USING DIFFERENT ITEM
25 = CONFLICTING DATA ENTRY LOCK ALREADY IN EFFECT
26 = IMMINENT DEADLOCK
41 = DBUPDATE ATTEMPTED TO MODIFY VALUE OF CRITICAL ITEM: KEY/SEARCH/SORT
42 = DBUPDATE WILL NOT ALTER A READ-ONLY DATA ITEM
43 = DUPLICATE KEY VALUE N MASTER
44 = CAN'T DELETE A MASTER ENTRY WITH NON-EMPTY DETAIL CHAINS
49 = ILLEGAL BUFFER ADDRESS
50 = USER'S BUFFER IS TOO SMALL FOR REQUESTED DATA
60 = DATABASE ACCESS DISABLED
61 = PROCESS HAS THE DATABASE OPEN 63 TIMES; NO MORE ALLOWED
69 = BAD DATABASE
401:: DBOPEN
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-13 = MUST BE CREATOR OF ROOT FILE OR DATABASE.
-21 = BAD PASSWORD.
-22 = MAINTENANCE WORD REQUIRED.
-31 = BAD MODE.
-32 = UNOBTAINABLE MODE.
-34 = DATABASE MUST BE RECOVERED BEFORE ACCESS IS ALLOWED.
-90 = ROOTFILE BAD.
-94 = DATABASE BAD.
60 = DATABASE ACCESS DISABLED.
61 = THIS DATABASE OPENED MORE THAN 63 TIMES BY THE SAME PROCESS.
402:: DBINFO
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-21 = BAD DATA SET REFERENCE.
-31 = BAD MODE.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
49 = ILLEGAL BUFFER ADDRESS.
50 = BUFFER IS TOO SMALL.
403:: DBCLOSE
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-21 = BAD DATA SET REFERENCE.
-31 = BAD MODE.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
-232 = ILLEGAL DBCLOSE MODE 2 USED DURING AN ACTIVE DYNAMIC TRANSACTION.

```



```

-235 = DYNAMIC TRANSACTION ABORTED DUE TO DBCLOSE MODE 1; DATABASE CLOSED.
-420 = FEATURE NO IMPLEMENTED.
404:: DBFIND
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-21 = BAD DATA SET REFERENCE.
-31 = BAD MODE.
-51 = BAD LIST LENGTH.
-52 = BAD ITEM.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
-258 = INVALID ARGUMENT FOR INDEX.
-259 = INVALID MODE FOR INDEX.
-260 = NO PREVIOUS LIST OF QUALIFIED DATA ENTRIES.
17 = NO MASTER ENTRY.
405:: DBGET
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-21 = BAD DATA SET REFERENCE.
-31 = BAD MODE.
-51 = BAD LIST LENGTH.
-52 = BAD LIST OR BAD ITEM.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
10 = BEGINNING OF FILE.
11 = END OF FILE.
12 = DIRECTED BEGINNING OF FILE.
13 = DIRECTED END OF FILE.
14 = BEGINNING OF CHAIN/QUALIFIER ENTRIES.
15 = END OF CHAIN/QUALIFIER ENTRIES.
17 = NO ENTRY.
18 = BROKEN CHAIN.
49 = ILLEGAL BUFFER ADDRESS.
50 = BUFFER IS TOO SMALL.
406:: DBUPDATE
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-12 = NO LOCK COVERS THE DATA ENTRY TO BE ADDED.
-14 = ILLEGAL INTRINSIC IN CURRENT ACCESS MODE.
-21 = BAD DATA SET REFERENCE.
-31 = BAD MODE.
-51 = BAD LIST LENGTH.
-82 = CIUPDATE IS SET TO DISALLOWED; CANNOT USE CRITICAL ITEM UPDATE.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
17 = NO ENTRY.
41 = DBUPDATE ATTEMPTED TO MODIFY VALUE OF CRITICAL ITEM: KEY/SEARCH/SORT.
42 = READ ONLY ITEM.
49 = ILLEGAL BUFFER ADDRESS.
50 = BUFFER TOO SMALL.
407:: DBPUT
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-12 = NO LOCK COVERS THE DATA ENTRY TO BE ADDED.
-14 = ILLEGAL INTRINSIC IN CURRENT ACCESS MODE.
-21 = BAD DATA SET REFERENCE.
-23 = DATA SET NOT WRITABLE.
-24 = OPERATION NOT ALLOWED ON AUTOMATIC MASTER DATA SET.
-31 = BAD MODE.
-51 = BAD LIST LENGTH.
-52 = BAD LIST OR BAD ITEM.
-53 = MISSING SEARCH OR SORT ITEM.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
16 = DATA SET FULL.
18 = BROKEN CHAIN.
43 = DUPLICATE KEY ITEM VALUE.

```

```

408:: DBDELETE
0      = SUCCESSFUL EXECUTION - NO ERROR
-11    = BAD DATABASE REFERENCE.
-12    = NO LOCK COVERS THE DATA ENTRY TO BE DELETED.
-14    = ILLEGAL INTRINSIC IN CURRENT ACCESS MODE.
-21    = BAD DATA SET REFERENCE.
-24    = DBDELETE NOT ALLOWED ON AUTO MASTER.
-31    = BAD MODE.
-222   = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
17     = NO ENTRY.
44     = CANNOT DELETE MASTER ENTRY WITH NON-EMPTY DETAIL CHAINS.
409:: DBLOCK
0      = SUCCESSFUL EXECUTION - NO ERROR
-11    = BAD DATABASE REFERENCE.
-31    = BAD MODE VALUE.
-121   = DESCRIPTOR COUNT ERROR.
-123   = ILLEGAL RELOP IN A DESCRIPTOR.
-124   = DESCRIPTOR TOO SHORT. MUST BE GREATER THAN OR EQUAL TO 9.
-125   = BAD SET NAME/NUMBER.
-126   = BAD ITEM NAME/NUMBER.
-127   = ATTEMPT TO LOCK USING A COMPOUND ITEM.
-128   = VALUE FIELD TOO SHORT IN A DESCRIPTOR.
-129   = P-TYPE ITEM LONGER THAN P28 SPECIFIED.
-130   = ILLEGAL DIGIT IN A P-TYPE VALUE.
-131   = LOWERCASE CHARACTERS IN TYPE U VALUE.
-132   = ILLEGAL DIGIT IN TYPE Z VALUE.
-133   = ILLEGAL SIGN IN TYPE Z VALUE.
-134   = TWO DESCRIPTORS CONFLICT.
-135   = DBLOCK CALLED WHEN LOCKS ALREADY IN EFFECT.
-136   = DESCRIPTOR LIST EXCEEDS 4094 BYTES.
-222   = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
20     = DATABASE LOCKED OR CONTAINS LOCKS.
22     = DATA SET LOCKED BY ANOTHER PROCESS.
23     = ENTRIES LOCKED WITHIN SET.
24     = ITEM CONFLICTS WITH CURRENT LOCKS.
25     = ENTRY OR ENTRIES ALREADY LOCKED.
26     = LOCK NOT PERFORMED SINCE DEADLOCK WOULD OCCUR.
410:: DBUNLOCK
0      = SUCCESSFUL EXECUTION - NO ERROR
-11    = BAD DATABASE REFERENCE.
-31    = BAD MODE.
-222   = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
411:: DBCONTROL
0      = SUCCESSFUL EXECUTION - NO ERROR
-11    = BAD DATABASE REFERENCE.
-14    = ILLEGAL INTRINSIC IN CURRENT ACCESS MODE.
-31    = BAD MODE.
-222   = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
-224   = DBCONTROL MODE 1 NOT ALLOWED INSIDE A DYNAMIC TRANSACTION.
412:: DBBEGIN
413:: DBEND
414:: DBMEMO
418:: DBEXPLAIN
419:: DBERROR
420:: DBXBEGIN
0      = SUCCESSFUL EXECUTION - NO ERROR
-11    = BAD DATABASE REFERENCE.
-31    = BAD (UNRECOGNIZED) DBXBEGIN MODE.
-139   = INVALID NUMBER OF BASE IDS.
-140   = BAD BASE ID LIST.
-151   = TEXT LENGTH GREATER THAN 512 BYTES.
-152   = DBXBEGIN CALLED WHILE A TRANSACTION IS IN PROGRESS.
-217   = DBOPEN MODE INCOMPATIBLE WITH DYNAMIC ROLLBACK.

```

```
-221 = CANNOT BEGIN TRANSACTION WHEN A DYNAMIC TRANSACTION IS ACTIVE.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
421:: DBXEND
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-31 = BAD (UNRECOGNIZED) DBXBEGIN MODE.
-140 = BAD BASE ID LIST.
-151 = TEXT LENGTH GREATER THAN 512 BYTES.
-217 = DBOPEN MODE INCOMPATIBLE WITH DYNAMIC ROLLBACK.
-222 = ONLY DBXUNDO ALLOWED WHEN A DYNAMIC TRANSACTION ENCOUNTERS AN ERROR.
-223 = CANNOT DBXEND OR DBXUNDO A TRANSACTION WHICH WAS NOT ACTIVE.
-238 = MDBX DBXBEGIN, DBXEND MODE MISMATCH.
422:: DBXUNDO
0 = SUCCESSFUL EXECUTION - NO ERROR
-11 = BAD DATABASE REFERENCE.
-31 = BAD (UNRECOGNIZED) DBXBEGIN MODE.
-140 = BAD BASE ID LIST.
-151 = TEXT LENGTH GREATER THAN 512 BYTES.
-223 = CANNOT DBXEND OR DBXUNDO A TRANSACTION WHICH WAS NOT ACTIVE.
-238 = MDBX DBXBEGIN, DBXEND MODE MISMATCH.
-240 = MDBX MODE MISMATCH.
```

APPENDIX B: TurboIMAGE Data Conversion Reference

TurboIMAGE Data Types

Type	Description
E	ieee floating point. sub-item length is in halfwords
I	signed integer, sub-item length is in halfwords
J	signed integer, but conforms to COBOL standards (i.e. s9999 has max value 9999). sub-item length is in halfwords
K	unsigned integer, no negative value. 1 halfword = 0-65K, 2 halfwords= 0-2 Billion, sub-item length is in halfwords
P	packed decimal, sub-item length is in nibbles, 2 to 28, with one digit used for the sign (note: TurboIMAGE will let you create a P48 or even larger, but COBOL will not process it)
R	classic HP 3000 floating point, old, 2 halfwords or 4 halfwords
U	uppercase ASCII chars, sub-item length is in bytes
X	any ASCII characters, sub-item length is in bytes
Z	zoned decimal number. sub-item length is in bytes

NOTES: The size of the entire data item must be a multiple of halfwords (16 bits). Therefore, P types normally come in multiples of 4 and U/X/Z types come in multiples of 2.

TurboIMAGE Conversion Types

Xn	Character, n bytes, define as Character in FORTRAN, X(n) in COBOL.
Un	Uppercase Character, n bytes, define as Character in Fortran, A(n) in COBOL
E2	Floating-Point, 4 bytes, define as Real in Fortran, not supported in HP COBOL
E4	Floating-point, 8 bytes, define as Double Precision in Fortran, not supported in HP COBOL
I1/J1	Integer, 2 bytes, define as Integer*2 in Fortran, S9 to S9(4) Comp in COBOL
I2/J2	Integer, 4 bytes, define as Integer*4 in Fortran, S9(5) to s9(9) Comp in COBOL
I4/J4	Integer, 8 bytes, define as S9(10) to S9(18) Comp in COBOL, not supported in Fortran
K1	Logical, 2 bytes, define as Logical in Fortran, not supported in COBOL
Zn	Zoned-Decimal, n bytes, s(n) Display in COBOL, overpunched
P4	Packed-Decimal, 2 bytes, s9(3) Comp-3 in COBOL, not supported in Fortran.

P8	Packed-Decimal, 4 bytes, s9(7) Comp-3 in COBOL, not supported in Fortran.
Pn	Packed-Decimal, n/2 bytes, s9(n-1) Comp-3 in COBOL, not supported in Fortran. Maximum N in HP COBOL is 19 (18 digits plus a sign).
Zn	Numeric Display, n bytes, s9(n) Display in COBOL, with sign "overpunched" in the units position (unless you specify SIGN IS SEPARATE, then there are only n-1 digits in the value).



3 easy ways to reach us for support:

- Visit us on the Web
- E-mail us: support@imaxsoft.com
- Call us: 408.253.1987 (24/7, 365)

iMaxsoft Corp.

20410 Town Center Lane, Ste. 295
Cupertino, CA 95014

Phone (408) 253-8808
Fax (408) 253-4008

www.imaxsoft.com