

Oracle Module Version 2.6

Installation and User's Guide

STOREWAY DPA



STOREWAY DPA

Oracle Module Version 2.6

Installation and User's Guide

Software

March 2008

**BULL CEDOC
357 AVENUE PATTON
BP.20845
49008 ANGERS CEDEX 01
FRANCE**

**REFERENCE
DPA_ORACLE_V2_6_EN**

The following copyright notice protects this book under Copyright laws which prohibit such actions as, but not limited to, copying, distributing, modifying, and making derivative works.

Copyright © Bull SAS 2008

Printed in France

Suggestions and criticisms concerning the form, content, and presentation of this book are invited. A form is provided at the end of this book for this purpose.

To order additional copies of this book or other Bull Technical Publications, you are invited to use the Ordering Form also provided at the end of this book.

Trademarks and Acknowledgements

We acknowledge the rights of the proprietors of the trademarks mentioned in this manual.

All brand names and software and hardware product names are subject to trademark and/or patent protection.

Quoting of brand and product names is for information purposes only and does not represent trademark misuse.

The information in this document is subject to change without notice. Bull will not be liable for errors contained herein, or for incidental or consequential damages in connection with the use of this material.

Table of Contents

Chapter 1. Product overview	7
Main features	7
Supported environments	7
Chapter 2. Oracle Architecture	8
The Oracle database	9
Tablespaces	10
Oracle Code.....	12
Datafiles.....	13
Redo log files.....	14
Active redo logs	14
Archived Redo logs	14
Redo log configuration	16
Simple configuration (3 disks)	16
Complex configuration (4 disks)	16
Control files	17
Initialization parameter file	18
Oracle trace files.....	19
Chapter 3. StoreWay DPA configuration prerequisites	20
ARCHIVELOG mode Parameters	21
Parameters Location	21
ARCHIVELOG mode transition	23
Elimination of Oracle meta-characters "?" and "@"	25
Chapter 4. Installation of the Oracle module	26
Installation procedure and initialisation of the Oracle module	27
Creation of the Oracle application	29
Chapter 5. Hot backup	32
Overview.....	33
Backed up elements	33
Backup profile and objects to back up	33
Objects to back up	33
Optimization of backups	35
Hot backup	35
Restoration and recovery	35
Chapter 6. Restoration and Recovery	36
Restoration and recovery prerequisites	37
General restoration procedures	38
Restoration window	38
Full database restoration	39
Restoration of a TableSpace	41
Missing non-critical datafile: hot restoration	42
Restoration and recovery of a non-critical datafile	42
Missing datafile system, cold restoration	44
Several datafiles missing, cold restoration.....	45

Missing archived redo log: hot restoration.....	46
Restoration of the ControlFile.....	48
Loss of a control file, cold restoration	48
Loss of all control files, cold restoration	49
Loss of active members redo log, hot restoration	49
Loss of all active redo logs, cold restoration	50
Chapter 7. Oracle Commands	52
Alter the Tablespace OFFLINE	52
Alter the Tablespace ONLINE	52
Stopping/restarting a database	52
List of files containing the different tablespaces	52
List of Datafiles files	52
Select the list of control files	52
Select the list of redologs files	52
Set tablespace in backup mode	53
Return a tablespace to normal mode	53
Force the switch of a current log	53
Index	55

Chapter 1. Product overview

The **StoreWay DPA Backup Module for Oracle** is an optional module which enables you to back up and restore Oracle type databases in complete security.

The StoreWay DPA for Oracle agent enables you to visualize database storage objects (tablespaces, active and archived redo logs, control files). It allows you to configure and plan your backups using an intuitive graphical user interface. The agent also exploits the power of the StoreWay DPA to locate and restore Oracle objects.

You can control the smooth backup performance thanks to the alarms generated by the StoreWay DPA.

This guide describes the features provided by the StoreWay DPA for Oracle agent.

Main features

- > Graphical representation database storage objects (tablespaces, active and archived redo logs, control files).
- > Full or partial ONLINE backup.
- > High performance through compression.
- > Troubleshooting assistance.
- > Restoration using time navigation.
- > Archived redo log file management.

Supported environments

Please refer to the compatibility guide.

Chapter 2. Oracle Architecture

The most important Oracle files are as follows:

- > The Oracle code.
- > The datafiles.
- > The active redo log files (online redo log files).
- > The archive redo log files.
- > The control files.
- > The initialisation configuration file.
- > Oracle trace files.

We advise you to understand a certain number of database concepts concerning Oracle and its architecture before planning backups.

See:

["The Oracle database" page 9](#)

["Tablespaces" page 10](#)

["Oracle Code" page 12](#)

["Datafiles" page 13](#)

["Redo log files" page 14](#)

["Redo log configuration" page 16](#)

["Control files" page 17](#)

["Initialization parameter file" page 18](#)

["Oracle trace files" page 19](#)

The Oracle database

An Oracle database is identified by its SID (*System Identification*) or system identifier. A SID is a database identifier which represents an Oracle Instance.

An instance corresponds to one or more tablespaces, storage zones in which the Oracle core stores data in the database.

A database can contain a number of datafiles, control files and parameter files. A given database instance is defined by a unique INIT file, and by other optional files.

NOTE: An Oracle database can use physical disk partitions (raw device) or standard Unix or Windows files for certain objects.

Tablespaces

Database data are stored in the tablespaces. A tablespace is a logical entity which corresponds to one or several files of physical data (datafiles) on the disk(s). The database is divided into one or more tablespaces. Each tablespace can contain one or several physical data files.

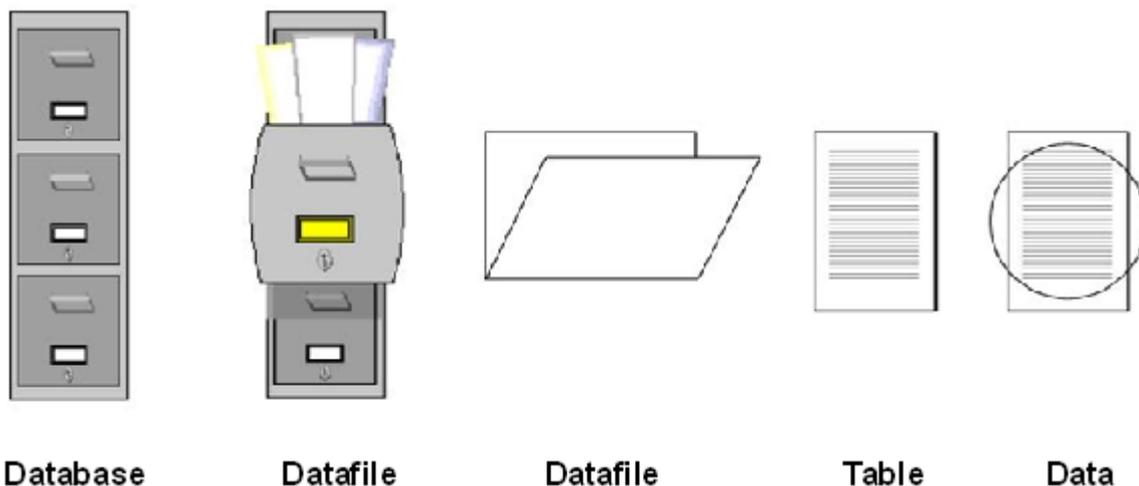
Before adding data to an Oracle database, you need to create a table in a tablespace.

Example: Creation of the table structure

```
create table client
(name varchar2(15),
prename varchar2(15),
postal_code number,
telephone_number_ number)
tablespace users;
```

The Oracle database structure can be compared to a filing cabinet.

The figure below represents the different database levels. This logical data group regroupes data which allows greater database flexibility during data operations.



Tablespaces are generally organized in terms of the type of the information they contain. The table below describes all types of Oracle tablespace.

Tablespace	Contains
System	The required tablespace which contains all system processes and datafiles. Can contain everything. The SYSTEM tablespace is always ONLINE when the database is open.
Temp	<p>All temporary tables. This tablespace is used to store temporary information.</p> <p>If the database is very active, several "temp" tablespaces can coexist.</p> <p>When multiple sorts executed concurrently are too large to fit in memory, designated sort space then a temporary segment can be utilised. A temporary tablespace is dedicated to sorting and is therefore more efficient</p>

Tools	The objects necessary for database maintenance (such as Oracle Forms).
Users	Information concerning the users.
Index (INDX)	<p>This tablespace to store all the indexes in the database. Indexes are database objects which enable Oracle to rapidly locate data stored in a table.</p> <p>Note: Putting an index tablespace into OFFLINE mode does not inhibit querying of the database only slow that querying down</p>
Rollback (RBS)	<p>Rollback segments (rollbacks or rbs) are database segments which stock information before data snapshots when a transaction modifies a block. They record all actions on the database and allow restoration back to a previous version of the database.</p> <p>The information contained in the rollback segment has several rollback entries called <i>undo</i>. The rollback segments guarantee the undo retention periods for a transaction.</p> <p>This tablespace is used for read consistency of transaction rollbacks during transaction retrievals.</p>

Oracle Code

When you install Oracle on a machine, many sub-directories and files are created. Installation procedures vary depending on the operating system.

On Unix or Windows systems, all Oracle sub-directories and files are created in the main ORACLE_HOME directory. The sub-directories contain files such as the Oracle executables and various SQL scripts which are essential for the administration and smooth running of the database. These are generally grouped under the terms *Oracle code*.

NOTE: We advise you to back up Oracle code at each upgrade to new version.

Datfiles

Datfiles comprise the physical support, or content, of all database data.

They are divided into several logical entities the smallest of which is the Oracle block.

At the time of the database creation, Oracle creates the SYSTEM tablespace which contains the system tables known as the *Oracle Data Dictionary*.

Two types of data are stored in database files:

- > **User data** which the user enters into the database.
- > **System data**: information required by the database to manage user data and manage itself. For example, thanks to system data, Oracle knows when a field is mandatory and when it must contain numbers, which user is authorised to access this database, how many files are in the database, where these files are located etc.

This table provides examples of *user* and *system* data:

Data type	Contains information on
User data	
Product Information	Product name, color, price, etc.
Client Information	Company name, phone number, contact, etc.
System data	
Tables	Table fields and the type of information they contain.
Size	Physical space used by the database objects.
Users	Names, passwords and privileges.
Datfiles	Number, location, last access.

Redo log files

Active redo logs

The *redo log files* are used by Oracle to record modifications to the database in normal usage. These files are open, or *online*, during the normal usage of the database. They are generally called *online redo log files* or *active redo log files*.

Each Oracle database instance is associated with a redo log which enables you to protect the database in the event of instance failure. An online redo log comprises at least two preallocated files which store all modifications and which are used in circular mode. Writing to these files is sequential and is performed by the Oracle Log Writer (ora_lgwr) background process each time the buffer zone reaches its limit or that a transaction is executed.

Example: Circular Mode

A database contains two redo log files: redolog1 and redolog2. If modifications, or transactions, are performed on the database, they are first written to the redolog1 file. When redolog1 is full, the following transactions are written to the redolog2 file. When the redolog2 is full, following transactions are once again written to redolog1.

The redo log files work in circular mode, when the redolog1 file is reused, all information it contained previously is overwritten.

Archived Redo logs

Oracle DBAs (*Database Administrators*) can choose to run the database in ARCHIVELOG mode or in NOARCHIVELOG mode. These modes are in direct correlation with the redo log files.

ARCHIVELOG Mode

The contents of the online redo log files are copied to an archive zone by a background process. These archived files are called redo log files archived (archived redo).

The redo log files work in circular mode. They are always copied before being overwritten. If the database needs to change the redo log files before the copy is finished, Oracle delays this operation to finish the copy. Oracle never authorizes the loss of the contents of a redo log file. Thanks to redo log files (online and archived), the database is protected against failures, manipulation errors or disk failure. It is the safest operational mode.

NOTE: The ARCHIVELOG mode allows you to perform hot backups of your Oracle database.

This table lists the advantages and drawbacks of the ARCHIVELOG mode:

Advantages	Drawbacks
All transactions can be relocated because modifications to the database are stored in the redo log files.	The DBA must set aside an archive space to copy the archived redo log files and ensure these files are also copied to the cartridges.
If datafiles are lost after a breakdown or defective media, the physical backup and the archived redo log files can be used to restore the database. Data security is guaranteed.	If disk space is insufficient, the database is locked. As long as online redo log files are saved, the database cannot continue the operations.
The tablespaces can be stopped immediately.	The copy of archived redo log files requires more disk space.
Online backups can be performed without needing to prevent user access to the database.	
Retrieving the database can be spread over the network if all database nodes work in ARCHIVELOG mode.	

NOARCHIVELOG Mode:

Logging is circular and the redo log files are not archived . The oldest redo logs are overwritten by the new without any copy being made. This means there is no guarantee that the data files backup copies can be rolled forward to synchronize with other data files. If a data file is lost, a return to the last full backup is the most reliable solution. This mode is only suitable for databases where all updates have been performed by batch and can be repeated if necessary.

The database is only protected against power cuts and not against disk failure. This mode is used by default.

The table below lists the advantages and drawbacks of the NOARCHIVELOG mode:

Advantages	Drawbacks
Reduced DBA tasks.	The database is unavailable during backups because it needs to close. Online backups are therefore impossible.
The database and its redo log files need less disk space	Tablespaces cannot be stopped immediately. The DBA can only retrieve the database as it was at the time of the last offline backup prior to the data loss. Offline backups must therefore be more frequent.

The NOARCHIVELOG mode is often considered to be more powerful than the ARCHIVELOG mode. However, in NOARCHIVELOG mode, you only avoid disk to disk copies and not information update writes to the redo logs. In fact, this operation is performed in every case because the transaction information is essential to execute a database recovery when the database starts after an incident, whatever the database archive mode.

Redo log configuration

If a database is subject to intensive transactions, its performances are reduced, because the read-write head will move more. This is why it is preferable to avoid storing the redo log files on the same disk as the other data files.

When the current redo log is full, Oracle switches to the next redo log and, in ARCHIVELOG mode, starts to archive (copy) this redo log. Ideally, it is preferable that these redo logs and file system archives are not on the same disk.

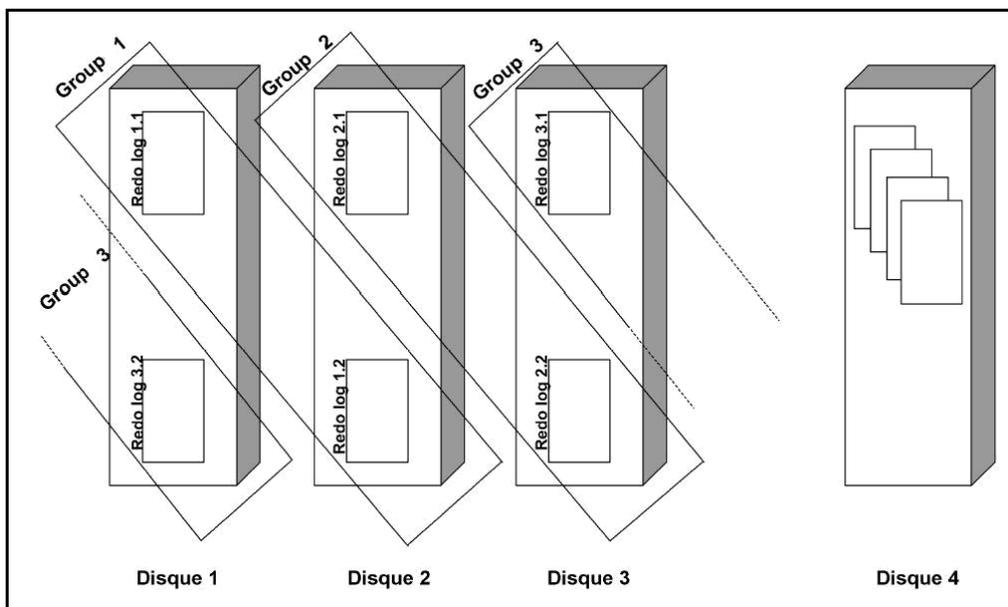
Simple configuration (3 disks)

- > Create two redo logs, one on disk 1 and the second on disk 2. The file system archives are on disk 3.
- > When the disk 2 redo log is active, the ora_arch process reads disk 1 and writes to disk 3.
- > When the disk 1 redo log becomes active again, the process reads the disk 2 and writes to disk 3.

Complex configuration (4 disks)

A more sophisticated configuration uses 4 disks (see the diagram below).

- > You can create a redo logs mirror of two separate disks to ensure the security of the transactions.
- > The redo log 1 group contains elements on disk 1 and disk 2. The redo log 2 group contains elements on disk 2 and disk 3 whereas the group 3 elements are on disk 3 and disk 1. System files archives are on disk 4.
- > When the redo log 2 group is active, the ora_lgwr process writes simultaneously to disks 2 and 3. The ora_arch process reads the first element of group 1 on disk 1 and copies the contents to disk 4.



NOTE: The redo log files (online and archived) are essential for the database retrieval because they contain information on all database modifications. We recommend you back up regularly the redo log files archived on tape.

Control files

The *control file* contains the database schema. It comprises essential information on the state of other database files. Several types of information are stored in this file: the names, locations, statuses and states of all datafiles and online redo log files. It can be used to identify the redo log files which are useful for the retrieval process. Each Oracle database has at least one control file.

When the database starts, Oracle reads the control file to locate datafiles and online log files. If the information contained in the control file is not valid, the database cannot start.

Each time a checkpoint is created or the database structure is modified, the control file is updated.

NOTE: The control file is essential to the smooth running of an Oracle database. If a media failure means the file is inoperative, the database is unavailable. It is recommended to keep at least two copies of the control file on separate disks and mounted with different controllers.

Initialization parameter file

When the software is delivered, Oracle provides an initialization parameter file called INIT.ORA. This file contains Oracle system parameters and must be used by the DBA to adapt the RDBMS configuration (Relational Databases Management System) to specific site needs.

Oracle reads the file when the database starts to know, among other things, the global system size, find control files etc.

NOTE: Given the importance of the control file and of the INIT.ORA file when Oracle starts up, they must be backed up regularly. The size of these files being extremely small, it is also recommended to keep online copies.

Oracle trace files

To analyze incidents and application parameters, Oracle creates text files called *trace files*. Each background Oracle process can write to an associated trace file if necessary. These files are typically called *background trace files*. The user process can also create trace files, and these files are called *user trace files*.

You can specify the location of the background and user trace files by setting the appropriate `INIT.ORA` parameter. All background trace files are created in a directory specified by the `BACKGROUND_DUMP_DEST` parameter. The `USER_DUMP_DEST` parameter specifies the location of the user trace files.

Oracle automatically creates trace files when Oracle experiences internal errors. Moreover, a DBA can force Oracle to create trace files by configuring different diagnostic events in the `INIT.ORA` file on Unix/Linux or `SPFILE.ORA` on Windows.

The trace directory must be examined daily to check if Oracle has not created significant trace files. The DBA must save trace files which are important and delete the others. It is recommended to regularly archive trace files to tape cartridges. Some DBAs automate these procedures.

Chapter 3. StoreWay DPA configuration prerequisites

The configuration of the StoreWay DPA ORACLE module requires:

- > The database to be in ARCHIVELOG mode.
- > The ORACLE DBA password (user system).
- > The complete Oracle installation path.

The StoreWay DPA for Oracle agent is designed to perform hot backup of your Oracle database. You can therefore perform backups of your database without stopping it or affecting users.

To perform hot backups of your database, it must be in ARCHIVELOG mode. The redo log files are therefore archived before being overwritten. In the case of restoration, it will be possible to replay all redo log files to retrieve the datafiles in their former state before requiring restoration.

NOTE: To perform a database backup, Oracle must be in automatic ARCHIVELOG mode.

By default, most data databases are in NOARCHIVELOG mode after creation. Switching to ARCHIVELOG mode implies the allocation of disk space to archive redo logs, the creation of an adequate directory owned by the database owner, and the modifications of parameters.

Environments supported

Please refer to the compatibility guide.

See:

["ARCHIVELOG mode Parameters" page 21](#)

["ARCHIVELOG mode transition" page 23](#)

["Elimination of Oracle meta-characters "?" and "@"" page 25](#)

ARCHIVELOG mode Parameters

Use the following parameters to switch the database to ARCHIVELOG mode. The four first parameters are Oracle and the last is a StoreWay DPA parameter.

log_archive_dest

Destination directory to which are added the prefix of the name given to the archived redo logs. By default, `$ORACLE_HOME/dbs/arch` on Unix and `%ORACLE_HOME%\database\arch` on Windows.

WARNING: Many Oracle versions do not allow you to specify a directory in this parameter: you must add a part of the name given to the archived redo log files.

log_archive_dest_1..._5 (oracle 8i)

These parameters only concern Oracle 8i and Oracle 9i and are used to create up to 5 mirror copies of the log files. They replace the standard `log_archive_dest` parameter. The path allocated to `log_archive_dest_1` must point to the local disk. If the opposite is true, the StoreWay DPA will not find the directory.

EXAMPLE: `"destination=ORACLE_HOME/dbs/arch/"`

log_archive_format

This parameter sets the name of the archived redo log files, for which there is a thread number and a replacement sequence number. The default value is `"%t_%s.dbf"`.

Any file extension can be used : `".dbf"` or `".arc"` or `".log"`. `%t` represents the thread number and `%s` the sequence number.

WARNING: Some of the "log_archive_format" values, although accepted by Oracle, are ambiguous and are not accepted by the StoreWay DPA for Oracle. For example, the value `"arch%t%s.log"` is ambiguous, because the file `"arch1129.log"` could represent either the sequence log 129 and thread 1, or the sequence 29 and thread 11. This is why Bull recommends you use the format `%t_%s`.

We recommend the use of the character `"%S"` for the sequence number, and not the variant `"%s"` this avoids any sorting problems during the display of the archived redo log destination directory (for example `"arch1_9999.log"` placed after `"arch1_10000.log"`). The variant `"%S"` will generate a 10-figure field such as `"arch1_0000009999.log"`.

We recommend the use of a different suffix from that of the datafiles for OFA (Optimal Flexible Architecture) compatible databases. They will enable you to eliminate the file system backup, the database datafiles without eliminating the redo log files (it's always a good idea to keep as many versions of these files as possible).

WARNING: Avoid overloading your configuration by using formats such as `"archive_thread_%t_sequence_%S.log"`. They take up a lot of room in the parent directory and slow down the access to the file. A format such as `"%t_%S.log"` is quite sufficient.

log_archive_start

This parameter enables the system to perform or not archiving operations. By default, `"false"`.

Parameters Location

- > On Unix systems, Oracle parameters are generally in the **`$ORACLE_DATABASE/admin/$ORACLE_SID/pfile/initSID.ora`** file for the OFA (Optimal Flexible Architecture) compatible databases or in the **`$ORACLE_HOME/dbs/directory`**.

They can also be located in a secondary file contained in the instruction "ifile" (a configSID.ora file for example). In this case, SID represents the instance identifier as with the ORACLE_SID environment variable.

- > On Windows, consult the **%ORACLE_HOME%\database** directory to locate these files. Oracle has no rules concerning the location of parameter files because a file can be specified by the pfile password when the database starts. The parameter files must be backed up with the rest of the Oracle, with a regular backup of the file system.

NOTE: When you modify parameters, the new values are not loaded until you start the database. Oracle does not update parameters files for changes made online. To fully install the StoreWay DPA for Oracle and perform hot backups, you need to completely stop the database for a short moment.

ARCHIVELOG mode transition

Switch the Oracle database to ARCHIVELOG mode

Modify the default `configSID.ora` file by adding the log mode parameters.

The file below corresponds to the file `configSID.ora` after modification:

```
:::::::::::
configDB80.ora.sample
:::::::::::
#
# $Header: cnfg.ora 1.1 95/02/27 12:14:25 wyim Osd<unix> $ Copyr
(c) 1992 Oracle
#
# cnfg.ora - instance configuration parameters

control_files = (/home01/oradata/DB80/control01.ctl,
/home02/oradata/DB80/control02.ctl,
/home03/oradata/DB80/control03.ctl)
# Below for possible future use...
#init_sql_files = (?/dbs/sql.bsq,
# ?/rdbms/admin/catalog.sql,
# ?/rdbms/admin/expvew.sql)
background_dump_dest = /export/home/oracle/admin/DB80/bdump
core_dump_dest = /export/home/oracle/admin/DB80/cdump
user_dump_dest = /export/home/oracle/admin/DB80/udump
log_archive_dest = /export/home/oracle/admin/DB80/arch/arch
log_archive_format = %t_%s.log
log_archive_start = true
#db_block_size = <blocksize>

db_name = DB80
```

Enter the Oracle Server Manager commands:

```
SVRMGR> show parameters log_archive
NAME TYPE VALUE
-----
-
log_archive_buffer_size integer 64
log_archive_buffers integer 4
log_archive_dest string ?/dbs/arch
log_archive_duplex_dest string
log_archive_format string %t_%s.dbf
log_archive_min_succeed_dest integer 1
log_archive_start booleanFALSE
SVRMGR> archive log list
Database log mode No Archive Mode
Automatic archival Disabled
Archive destination ?/dbs/arch
Oldest online log sequence 572
Current log sequence 574
SVRMGR> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
SVRMGR> host
charlie.oracle(1) [ ~/admin/DB80/pfile ] mv configDB80.ora
configDB80.ora.orig
```

```

charlie.oracle(2) [ ~/admin/DB80/pfile ] mv configDB80.ora.sample
configDB80.ora
charlie.oracle(3) [ ~/admin/DB80/pfile ] exit
SVRMGR> startup mount
ORACLE instance started.
Total System Global Area 4864192 bytes
Fixed Size 47296 bytes
Variable Size 4325376 bytes
Database Buffers 409600 bytes
Redo Buffers 81920 bytes
Database mounted.
SVRMGR> alter database archivelog;
Statement processed.
SVRMGR> alter database open;
Statement processed.
SVRMGR> show parameters log_archive

```

```

NAME TYPE VALUE
-----
log_archive_buffer_size integer 64
log_archive_buffers integer 4
log_archive_dest string /export/home/oracle/admin/DB80
log_archive_duplex_dest string
log_archive_format string %t_%S.log
log_archive_min_succeed_dest integer 1
log_archive_start boolean TRUE
SVRMGR> archive log list
Database log mode Archive Mode
Automatic archival Enabled
Archive destination /export/home/oracle/admin/DB80/ arch/arch
Oldest online log sequence 572
Next log sequence to archive 574
Current log sequence 574
SVRMGR>

```

Elimination of Oracle meta-characters "?" and "@"

Some parameters which declare physical paths such as `log_archive_dest` or `user_dump_dest` or control file paths can contain old Oracle meta-characters: "?" for `ORACLE_HOME` and "@" for the `ORACLE_SID` value. These meta-characters are rejected by the StoreWay DPA and must be replaced by the appropriate environment variables in all parameter files.

NOTE: Oracle replaces the environment variable parameter values in all versions from 7.1, and therefore this operation should not be problematic.

All referenced variables must be defined in the environment when the database is started.

Once you have changed Oracle parameter, you must stop and restart the database using special sequence to change the database log on mode before it is opened and available for users. To do this, use the command line version of the server managers or **SQL*Plus**.

Chapter 4. Installation of the Oracle module

See:

["Installation procedure and initialisation of the Oracle module" page 27](#)

["Creation of the Oracle application" page 29](#)

Installation procedure and initialisation of the Oracle module

The use of this module requires a valid license.

WARNING: If Oracle is not installed on your server, do not select Oracle in the choice of applications to install, the standard backups will fail when the executables call the Oracle libraries without finding them.

If the client agent is already installed without the StoreWay DPA for Oracle option or if the Oracle version has been upgraded on your system, you need to uninstall the agent before reinstalling it again.

Installation initialisation process.

- 1 Insert the **StoreWay® DPA Initial Setup (Windows), Agents (Windows, Linux, Netware, MAC OS), Graphical User Interface Setup, ASM & Disaster Recovery Agents Windows) Version X.X.XXX** CD-ROM on the Windows server you want to protect. Click the StoreWay DPA backup agent link to start the Setup program
- 2 Select the language.
- 3 Follow the setup for these steps:
 - Select the installation directory.
 - Select the StoreWay DPA.
 - Choose the application to hot backup: **Oracle** in our case.
 - Choice of application version (Oracle 8, 9, or 10).
 - Continue until you reach the summary page before installation.
 - Click **Finish** to go to the next step.

Configuring the application: Password Manager Application

- > To connect to the Oracle server, the **StoreWay DPA** uses a login and therefore needs to know the associated password. This login and password occurs on the SQL server thanks to the Password Manager program which is in the list of your programs in the **StoreWay DPA** directory.
- > The programme requests the instance name. Set login and password. Entering the password is invisible on screen but the programme requests a confirmation.

NOTE: If the server where you installed the StoreWay DPA agent has several instances, you must save as many entries as you have instances (configured in ARCHIVELOG mode of course).

Password Manager application parameters:

- > Instance name: Oracle system identifier for the instance to use to log onto the database.
- > User name: User allowed to log onto the database. Typically, this is SYS, but can also be any user with sysdba privileges.
- > Password: Oracle password for user access. This password enables **StoreWay DPA** for Oracle to log onto the database as an administrator.

NOTE: The passwords are stored encrypted in the StoreWay DPA and the backups can be performed without the backup administrator.

> Path Oracle Home : Specifies the directory where Oracle is installed.

- On Unix : ORACLE_HOME=/export/home/oracle/product/ 9.0.1
- On Windows : ORACLE_HOME=d:\oracle\product\9.0.1

WARNING: The ORACLE_HOME variable must have exactly the same value as the ORACLE_HOME variable used for starting the Oracle instance.

Creation of the Oracle application

- 1 Open the **settings** menu, click **Systems**:
- 2 Click the **Create a new application**  icon.
- 3 Select the type of application to create from the list.
- 4 Click **Create**.

Application: Edit

Description

Name:

Type: Oracle

Linked to: sql

Status: Active 

Comments:

Settings

Path to file PWD.ora :

* Path to 'spfile' or 'init.ora' file :

* Oracle instance :

Oracle server name (if different from the system name) :

** Mandatory parameters*

Backup

Data to backup

Add:

Type	Data path
Directories	<input type="text"/>

Profiles checked for this application

Activated	Name	Associated data to backup
<input checked="" type="checkbox"/>	Critical applications	All data
<input type="checkbox"/>	Normal applications	All data

Parameters:

- > **Path to the PWD.ora file** (optional variable): indicates the file access path containing the Oracle password, if you want to back up the Oracle system identifier file to the instance to use to log onto the database.
- > **Path to spfile or init.ora file** (mandatory variable): indicates the Oracle database «ora.ini» file access path containing the database configuration parameters.

- > **Oracle Instance** (mandatory variable): this variable is absolutely necessary to enable navigation in the Oracle database. It must contain the Oracle instance name (SID) on which you wish to navigate and backup. This variable is case-sensitive and must be the same as the definition in the Password Manager application).
- > **Oracle server name** (optional variable): if it is different from the system name where the application is hosted. This optional variable is to be used when the server name indicated in Oracle is not the same as the system name (hostname).

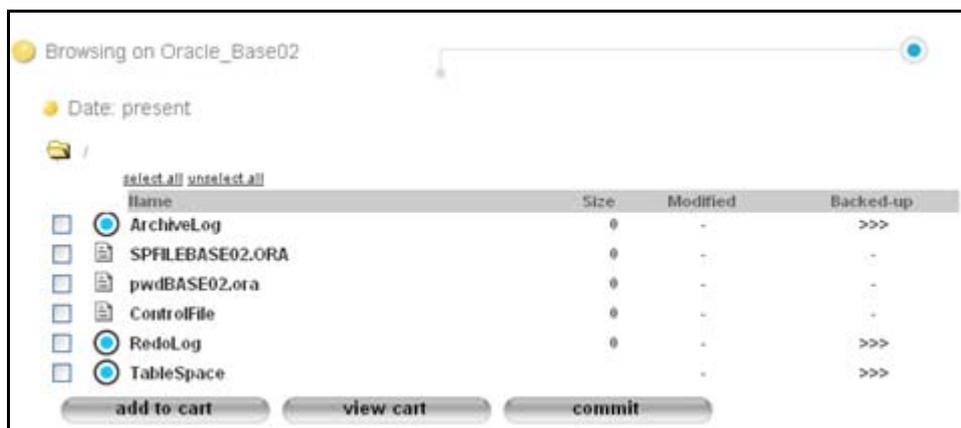
Backup:

- > **Data to backup:** Specify the data to back up for this application; enter, for example / in the **Data path** field to back up all data.

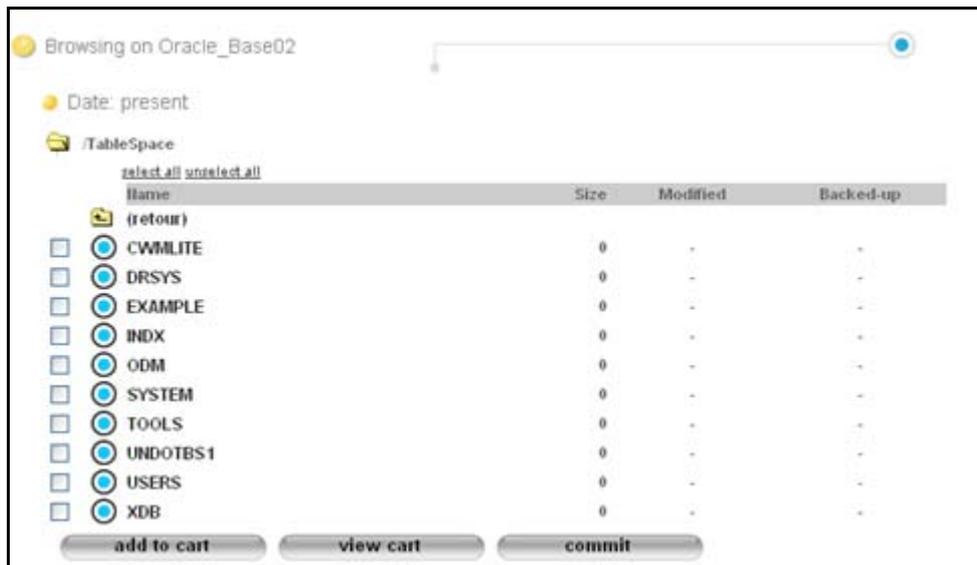
NOTE: It is recommended to associate the application with an existing profile which backs up all data («/»). In this case leave the **Data to backup** field empty.

Validate this configuration

- 1 To validate the access to the Oracle instance (parameters and application logs), click **browse** in the application in edit mode to open the instance file tree structure:



- 2 At this stage, you must display metafiles and meta-objects. To access TableSpaces, click the meta object (link) to open the file tree structure:



Composition of the metatree: metafiles: ControlFile, PWDtest.ora and SPFILETEST.ORA.

The **StoreWay DPA** application meta-trees comprise objects which can be backed up or restored individually. These objects are represented as metafiles.

Composition of the metatree: metaobjects: Archivelog, RedoLog and TableSpace

	Name
<input type="checkbox"/>	<input checked="" type="radio"/> ArchiveLog
<input type="checkbox"/>	<input type="radio"/> SPFILEBASE02.ORA
<input type="checkbox"/>	<input type="radio"/> pwdBASE02.ora
<input type="checkbox"/>	<input type="radio"/> ControlFile
<input type="checkbox"/>	<input checked="" type="radio"/> RedoLog
<input type="checkbox"/>	<input checked="" type="radio"/> TableSpace

	Name
<input type="checkbox"/>	<input type="radio"/> (up)
<input type="checkbox"/>	<input checked="" type="radio"/> CWMNLITE
<input type="checkbox"/>	<input checked="" type="radio"/> DRSYS
<input type="checkbox"/>	<input checked="" type="radio"/> EXAMPLE
<input type="checkbox"/>	<input checked="" type="radio"/> INDX
<input type="checkbox"/>	<input checked="" type="radio"/> ODM
<input type="checkbox"/>	<input checked="" type="radio"/> SYSTEM
<input type="checkbox"/>	<input checked="" type="radio"/> TOOLS
<input type="checkbox"/>	<input checked="" type="radio"/> UNDOTBS1
<input type="checkbox"/>	<input checked="" type="radio"/> USERS
<input type="checkbox"/>	<input checked="" type="radio"/> XDB

Meta-directory TABLESPACES

This meta-directory comprises a fixed name and all database tablespaces. Each tablespace is represented by an object with the name of the tablespace containing the corresponding datafile, for example USERS.

Meta-directory _ARCHIVE_LOGS

This meta-directory contains a meta-file for each archived redo log file detected on the disk file system.

This directory is defined by the Oracle parameters log_archive_dest or log_archive_dest1.

Meta-directory _REDO_LOGS

This meta-directory contains a meta-file for each active redo log group known to the database plus a private MetaLink giving the private physical address of each member of the group. A MetaLink collector points towards the directory of destination archived redo logs in which Oracle archives the full redo logs and where the administrator switches the logs.

NOTE: For transactional databases: ensure your active redo log files are mirrored to separate disks, if possible with different controllers. This will avoid any transaction loss.

Chapter 5. Hot backup

See:

["Overview" page 33](#)

["Optimization of backups" page 35](#)

Overview

Backed up elements

- > Backup at the start and end of the control file (CONTROLFILE) → generation of 2 files (begin & end).
- > Parallel backup of the TABLESPACES.
- > Backup of REDOLOGS (after current log switch).
- > Parallel backup of the ARCHIVELOGS.
- > Backup of the initialisation file (initSID.ora).
- > Backup of the PASSWORD ORACLE file (If defined as a parameter).

It is, of course, essential to back up the elements contained in the meta-tree.



Backup profile and objects to back up

You have to associate your Oracle application with a backup profile, and indicate the objects to back up.

The **StoreWay DPA** backup profiles concept unites administration needs and a shared file system and application interface. If you know how to administrate files systems backup with **StoreWay DPA**, you know how to administrate Oracle backups.

Objects to back up

You must back up the meta-directory root and use this for the full database backup. The **StoreWay DPA** backs up all your database and ensures overall consistency.

To back up the entire database, indicate / (character slash) as the object to back up.

Prefer the backup of the entire application using the application profile:

● Data to backup

Add:

Type	Data path
Directories	<input type="text"/>

● Profiles checked for this application

Activated	Name	Associated data to backup
<input checked="" type="checkbox"/>	Critical applications	All data
<input type="checkbox"/>	Normal applications	All data

ControlFile Backup:

The "ControlFile" file is backed up before and after each full backup. The **StoreWay DPA for Oracle** module backs it up via a metafile "ControlFile_begin" and "ControlFile_end". This metafile does not actually exist; it does not appear in the present. You need to activate the field depth for this file to appear.

The "ControlFile" file is also backed up incrementally: it is not renamed.

In the event of a restoration, it will be restored to the StoreWay DPA agent installation directory, you need to copy it to its original directory.

Multiple copies of the ControlFile and ArchiveLogs:

Oracle enables you to mirror ControlFile and ArchiveLogs files (using the ARCHIVE_LOG_DEST_# variable). If these features are activated, and in the event of a missing ControlFile or ArchiveLog file, you can retrieve its mirror copy, without having to restore it.

Operations performed

- > SQL Begin and End Backup command.
- > Purge of ARCHIVELOGS files after the second backup of the same file.

Incremental Backup

An incremental backup systematically backs up the following objects whatever objects are chosen in the interface: ArchiveLog, RedoLog and ControlFile.

Pre- and post-processing management

If an administrator wants to perform backup processing of an Oracle application, he can create his own script and execute it in post-processing in the backup profile associated with the Oracle application:

```
"path to script\my script.bat" arg1 ... argn
```

Optimization of backups

The following section provides additional information which may be useful for the optimization of backup restoration recovery timeslots.

Hot backup

Each Oracle datafile contains a heading with essential information: the size of the datafile and the block, the associated tablespace, the creation date and the SCN value corresponding to the latest checkpoint performed on this datafile.

During the hot database backup, the StoreWay DPA sets each tablespace in turn to "**BEGIN BACKUP**" mode and logs it into the backup event log. The datafiles belonging to the tablespace are subject to a checkpoint (all current modifications are flushed from the memory) and their heading updated.

During the backup of these datafiles, Oracle stops logging database checkpoint occurrences in the headers. When a tablespace is backed up, the StoreWay DPA sets the mode to "**END BACKUP**", and logs it in the event log. Oracle then updates datafiles headings to reflect the current checkpoint.

Restoration and recovery

When a datafile is restored, it contains information on the checkpoint prior to the hot backup, but none of the checkpoints performed during this backup. It is for this reason that Oracle searches for all the redo logs necessary for the recovery and replays all modifications performed during the backup.

As soon as you open a database or attempt to alter a tablespace or datafile online, Oracle checks that the information contained in the datafile heading correspond to those in the control file. If this is not the case, a recovery is necessary.

The time required for a restoration depends on the number of cartridges to insert and the volume of data to restore. The length of the recovery depends on the number of redo logs to apply to the datafiles which need a recovery.

It is by applying the redo logs to the datafiles and not reading these files which takes the longest. Applying the redo log files all modifications to the different datafiles to be found which leads to a lot of disk head movement. Reading sequential redo log files is much quicker and therefore not penalizing, for example, to read a long sequence of redo log files to restore an old datafile which has few modifications.

On the other hand, the application of redo logs to a datafile which is constantly modified because it contains rollback segments, for example, will take longer.

- 1 Open the Tablespace meta-directory, click the tablespace system, then select the datafile and click **add to the cart**.
 - **Cart contents** Window: click **restore**.
 - **Restoration options** Window: click **restore** to start the process.
- 2 Use a file manager or any other appropriate tool to check that the datafile was correctly restored to its location with the date of the previous backup.
- 3 Retrieve the datafile using the Oracle or SQL*Plus Server Manager. Oracle will not allow you to specify a tablespace recovery command if the database is closed.

Chapter 6. Restoration and Recovery

See:

["Restoration and recovery prerequisites" page 37](#)

["General restoration procedures" page 38](#)

["Full database restoration" page 39](#)

["Restoration of a TableSpace" page 41](#)

["Missing non-critical datafile: hot restoration" page 42](#)

["Missing datafile system, cold restoration" page 44](#)

["Several datafiles missing, cold restoration" page 45](#)

["Missing archived redo log: hot restoration" page 46](#)

["Restoration of the ControlFile" page 48](#)

Restoration and recovery prerequisites

You must have access to a database to perform the tests on which the following scenarios will be reproduced.

Do not wait for a corrective operation to be required on your operational database to read this chapter.

Ensure that you have backed up the database several times before voluntarily destroying datafiles. Each description of object destruction and restoration procedure is documented with the corresponding Oracle Server Manager or SQL*Plus commands.

If you are used to restoring lost or corrupt files from file system backups using the StoreWay DPA, your experience will be useful but insufficient to restore applications. The objects from data databases must be synchronized to conserve their integrity.

Database restoration has two steps:

- > Restoration is the procedure which consists of reading a group of objects from media and restores them to disk.
- > Recovery is the procedure which consists of applying replayed information to datafiles to obtain a consistent database.

Restoration is performed using StoreWay DPA graphical interfaces. Recovery is performed using Oracle commands and is also known under the name of media recovery.

Recovery applies exclusively to the database repair and is necessary because the datafiles must be consistent in time, whatever their content (tables, index, stored procedures, triggers, etc.) before being opened and used. If you try to alter a datafile online, without first performing a recovery to the same point in time as the rest of the database, Oracle refuses the operation.

The restoration scenarios in this chapter are followed by examples of commands for Oracle or SQL*Plus Server Manager. These examples list the procedures to follow if you want to try out these restoration scenarios beforehand.

Bull recommends you use the Oracle Server Manager (svrmgrl) for the Oracle 8 and 8i databases and SQL*Plus (sqlplus) for Oracle 9i and 10.

General restoration procedures

Restoration operations are performed from the web restoration interface:

- > Select the operations/Restoration menu.
- > Select the application then Open a new restoration session

Handling objects in the meta-tree

Select objects

Selecting objects allows you to restore them later.

In the file tree, select a meta-directory. All sub-meta-directories and meta-files included in this directory are selected.

In the restoration interface, clicking the name of the object (file or meta-directory) displays the list of backed up versions of this object.

To synchronize a backup date for this object, simply select the version and click choose the version.

NOTE: By default, displaying missing objects is configured with a value which corresponds to the synchronisation of the last ControFile backup.

Restoration window

In each window, select the objects you want to restore, click add to the cart, a padlock appears next to the object which means this object can be restored. The restoration window opens.

NOTE: If you want to remove an object from the cart, click restore to display the list of selected objects. Remove them from the cart by selecting them and clicking remove from cart.

Full database restoration

Cold restoration of the last backup

This method consists of restoring the complete database as it was before the incident. You must first synchronize the MetaTree within a time frame defined by the backup of a control file then restore a complete group of tablespaces using datafile instances including previous backups. You must then perform a database recovery by positioning in the timeframe and recreating the control files.

To test the restoration procedures, you can simulate the loss of a table by removing them from the database or by renaming them after performing a full database backup.

Perform a restoration and recovery of the database

- 1 Use the Oracle or SQL*Plus Server Manager to stop the database. The database cannot be stopped correctly if some control files are missing.
- 2 Open the StoreWay DPA restoration interface on the target Oracle application.
- 3 Click the ControlFile_begin or ControlFile_end metaobject and select the database version you want to restore. All transactions after this date will be lost.
- 4 Select all the files to restore:
 - Redologs
 - Tablespace
 - Controlfiles
- 5 Click add to the cart.
 - **Cart contents:** To restore, click restore.
 - **Restoration option :** Click restore to start the process.
 - For security reasons, the controlFiles are restored to the config directory installation of the StoreWay DPA client agent (e.g.: C:\Program Files\dpa\config) under the name: ctl1ORCL_begin.ora and ctl1ORCL_end.ora
 - Copy one of these files to the original Oracle directory by renaming it **CONTROL01.CTL**.

WARNING: You must also update also the mirrored CONTROL_FILE files if there are any.

Example:

```
Copy of the file:
C:\Program Files\dpa\config\ctl11TESTDB_begin.ora
in C:\oracle\oradata\testdb\testdb, renamed en CONTROL01.CTL,
duplicated for CONTROL02.CTL and CONTROL03.CTL (file mirror)
> Use the Oracle or SQL*Plus Server Manager to perform the application database
recovery operations.
```

Example : Server manager commands

```
SQL>startup mount ;
Instance ORACLE started.
Total System Global Area 135338868 bytes
Fixed Size 453492 bytes
```

```
Variable Size          109051904 bytes
Database Buffers      25165824 bytes
Redo Buffers          667648 bytes
Database mounted.
```

```
SQL>recover database using backup controlfile until cancel;
ORA-00279: change 1543894 generated 08/09/2005 11:19:29 required
for thread 1
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00026.001
ORA-00280: change 1543894 for thread 1 is on sequence n° 26
Indicate the log : {<RET>=suggested | namefile | AUTO | CANCEL}
CANCEL;
SQL>alter database open resetlogs;
```

WARNING: When the latest archivelog is restored, enter CANCEL.

Restoration of a TableSpace

The restoration of an Oracle TableSpace requires four steps:

- 1 Alter the TableSpace offline in Oracle.
- 2 Restore the TableSpace via the StoreWay DPA.
- 3 Perform a « recover tablespace » in Oracle.
- 4 Alter the TableSpace online in Oracle.

Missing non-critical datafile: hot restoration

You can restore and retrieve the datafile in the present, without having to stop the database. This procedure is called hot restoration.

To test restoration procedures, you can simulate the loss of a datafile by removing it from the database, renaming it or compressing it whilst the database is in use. Choose a datafile whose tablespace is not critical (NOT a system tablespace, NOR a tablespace containing active rollback segments).

NOTE: On Windows, the operating system can prevent the destruction of a file if the database is active. You need to alter the datafile or its tablespace offline, stop the database, destroy the datafile then restart the database.

Restoration and recovery of a non-critical datafile

- 1 Open the application restoration interface.
- 2 Click the time navigation button and position a display value of missing datafiles before selecting the date. The meta-tree displays all modifications to this tree within the chosen timeframe.

NOTE: By default, the missing files display is configured with a value corresponding to the synchronization of the last ControFile backup.
- 3 Open the relevant tablespace: the datafile appears with a red cross, which means the datafile in question is missing.
- 4 Select the datafile (or the tablespace : n+1)
- 5 Click add to the cart: the checkbox becomes a padlock which means the object is selected and eligible for restoration.
- 6 Contents of the cart Window: click restore.
- 7 Restoration Option Window: click restore to start the process.
- 8 Use a file manager or any other appropriate tool to check that the datafile has been restored to its initial location with the date of the previous backup.
- 9 Perform a datafile or tablespace recovery using the Oracle or SQL*Plus Server Manager.
- 10 Use the Oracle or SQL*Plus Server Manager to alter the datafile or tablespace online.

```
SQL> connect sys as sysdba
Connected.
SQL> recover tablespace example;
Media recovery complete.
SQL> alter tablespace example online;
Tablespace altered.
SQL>
```

If the database is dismounted, you must run the following command after the recovery:

```
SQL> alter database online;
Database altered.
```

Example: Server manager commands, recovery requiring archive logs.

```
SQL> connect sys as sysdba
Enter password:
Connected.
```

```
SQL> alter tablespace tools online;
alter tablespace tools online
*
ERROR at line 1:
ORA-01113: file 6 needs media recovery
ORA-01110: data file 6:
'C:\Oracle\product\10.1.0\oradata\orcl\tools01.dbf'
SQL> recover tablespace tools;
ORA-00279: change 243957 generated at 11/22/2001 21:50:07 needed
for thread
1
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00026.001
ORA-00280: change 243957 for thread 1 is in sequence #26

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00279: change 244886 generated at 11/23/2001 23:31:43 needed
for thread
1
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00027.001
ORA-00280: change 244886 for thread 1 is in sequence #27
no longer needed for this recovery
Log applied.
Media recovery complete.
SQL> alter tablespace tools online;
Tablespace altered.
```

Missing datafile system, cold restoration

To test restoration procedures, you can simulate the loss of a datafile system by removing the database, renaming or compressing it while the database is in service.

NOTE: On Windows, the operating system may prevent you from destroying a file if the database is in service. You will need to stop the database to destroy the datafile.

It is not possible, however, to hot restore a datafile system because the tablespace system is crucial to the smooth running of the database (it contains, among others, the data dictionary), you must therefore stop the database and cold restore the missing file.

Perform a restoration and recovery of a datafile system

- 1 Use the Oracle or SQL*Plus Server Manager to stop the database.
- 2 Check that the datafile system is really missing. The database is mounted (to prepare the recovery) but cannot be opened.
- 3 Open the StoreWay DPA restoration interface for the target Oracle application. The database is offline, the display of missing objects enables you to see the entire list of « inactive » database application files, namely:
 - The Passwd and spfile or init.ora files.
 - The backup of the controlFile ? 2 versions: at the start and end of the database backup for security reasons.
 - All Tablespaces, Redologs and Archivelogs.
- 4 Open the Tablespace meta-directory, click the tablespace system, then select the datafile and click add to the cart.
 - Cart contents Window: click restore.
 - Restoration options Window: click restore to start the process.
- 5 Use a file manager or any other appropriate tool to check that the datafile was correctly restored to its location with the date of the previous backup.
- 6 Retrieve the datafile using the Oracle or SQL*Plus Server Manager. Oracle will not allow you to specify a tablespace recovery command if the database is closed.

Example: Server manager commands

```
SQL> shutdown abort
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area  171966464 bytes
Fixed Size                  787988 bytes
Variable Size              145750508 bytes
Database Buffers           25165824 bytes
Redo Buffers                 262144 bytes
Database mounted.
ORA-01157: cannot identify/lock data file 1 - see DBWR trace file
ORA-01110: data file 1:
'C:\ORACLE\PRODUCT\10.1.0\ORADATA\ORCL\SYSTEM01.DBF'

SQL> recover datafile
'C:\ORACLE\PRODUCT\10.1.0\ORADATA\ORCL\SYSTEM01.DBF';
Media recovery complete.
SQL> alter database open;
Database altered.
SQL>
```

Several datafiles missing, cold restoration

In the event of missing or the corruption of several datafiles, it is often faster to restore the tablespaces in which they were located rather than try to know exactly which datafiles to restore.

If you use the previous scenario method, you are going to perform a restoration and a recovery. The recovery may be long if there are many archived redo logs. You must then open the database and Oracle warns you that another datafile is missing. You then need to restore it and apply all the archived redo logs a second time.

To test the restoration procedures, you can simulate the loss of several datafiles by removing them from the database or by renaming them or compressing them while the database is in service.

NOTE: On Windows, the operating system may prevent you from destroying a file if the database is in service. You need to stop the database to destroy the datafile.

Perform the restoration and recovery of several datafiles

- 1 Use the Oracle or SQL*Plus Server Manager to stop the database.
- 2 Check that at least one datafile is missing or is corrupt. The database is mounted (to prepare the recovery) but cannot be opened
- 3 Open the **StoreWay DPA** restoration Interface of the target Oracle application .
- 4 Select the TABLESPACES meta-directory (and therefore all tablespaces and datafiles) and select **add to the cart** to restore the most recent backup of each available datafile.
- 5 Contents of the cart: to restore, click **restore**.
- 6 Restoration option: click **restore** to start the process.
- 7 Retrieve the entire database using the Oracle or SQL*Plus Server Manager.

Example: Server manager commands

```
SQL> shutdown abort
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area 171966464 bytes

Fixed Size                787988 bytes
Variable Size             145750508 bytes
Database Buffers         25165824 bytes
Redo Buffers              262144 bytes
Database mounted.
ORA-01157: cannot identify/lock data file 1 - see DBWR trace file
ORA-01110: data file 1:
'C:\ORACLE\PRODUCT\10.1.0\ORADATA\ORCL\SYSTEM01.DBF'
SQL> recover database;
Media recovery complete.
SQL> alter database open;
Statement processed.
SQL>
```

Missing archived redo log: hot restoration

In the event of one or more missing archived redo log disk files, you can restore them without stopping the database.

To test the restoration procedures, you can simulate the loss of a recent archived redo log by removing the database or renaming it. You must also remove a non-critical datafile (which does NOT belong to a tablespace system, NOR a tablespace containing active rollback segments) for which the recovery will require the missing redo log file.

NOTE: On Windows, the operating system may prevent you from destroying a file if the database is running. You must therefore stop the database to destroy the datafile

Perform the restoration and recovery of an archived redo log

- 1 Open the **StoreWay DPA** restoration interface on the target Oracle application then define a « time navigation » value.
- 2 Open the ARCHIVE_LOGS meta-directory and select the missing archived redo log file. You may need to extend the field depth to find the missing archived redo log file, in the case of a database being backed up after the loss of the archived redo log.
- 3 Select the objects to restore and click **add to the cart**.
- 4 Contents of the cart: to restore click **restore**.
- 5 Restoration option: Click **restore** to start the process.
- 6 Check that archived redo log file has been restored to its initial location using a file manager or any other appropriate tool.
- 7 Use the Oracle or SQL*Plus Server Manager to perform a valid recovery of the datafile or the tablespace. Alter the datafile or tablespace ONLINE.

Example: Server manager commands

```
SQL> connect sys as sysdba
Enter password:
Connected.
SQL> alter tablespace tools online;
alter tablespace tools online
*
ERROR at line 1:
ORA-01113: file 6 needs media recovery
ORA-01110: data file 6:
'C:\Oracle\product\10.1.0\oradata\orcl\tools01.dbf'

SQL> recover tablespace tools;
ORA-00279: change 243957 generated at 11/22/2001 21:50:07 needed
for thread
1
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00026.001
ORA-00280: change 243957 for thread 1 is in sequence #26
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
AUTO
ORA-00279: change 244886 generated at 11/23/2001 23:31:43 needed
for thread
1
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00027.001
```

```
ORA-00280: change 244886 for thread 1 is in sequence #27  
no longer needed for this recovery  
Log applied.  
Media recovery complete.  
SQL> alter tablespace tools online;  
Tablespace altered.  
SQL>
```

Restoration of the ControlFile

To prevent handling errors, StoreWay DPA for Oracle restores the ControlFile in the config directory of the StoreWay DPA agent directory installation. You need to recopy it to the directory where you want to restore it.

Loss of a control file, cold restoration

If you lose a control file, the situation is easy to restore providing the control file is mirrored. This is what happens, for example, after a failure of a disk which contains a mirror of the control file.

If your control file is mirrored but on the same disk, you will need to recreate it.

NOTE: It is recommended that your control file be mirrored to a separate disk using, preferably, different controllers. You then restore the missing control file mirror by copying a « healthy » control file from another disk. To test the restoration procedures, you can simulate the loss of a recent archived redo log by removing the database or renaming it.

Perform the restoration and recovery of a control file

- 1 Open **StoreWay DPA** restoration interface on the target Oracle application then define, if necessary, a « time navigation » value.
- 2 Stop the database again and copy a healthy control file to the same location as the missing control file using the appropriate operating system command (such as "cp" or "dd").
- 3 Start the database and come back to the present in the **User Interface**. The MetaLinks of the control file now appear inline.

Example: Server manager commands

```
SQL> shutdown abort
ORACLE instance shut down.
SQL> startup
ORACLE instance started.
Total System Global Area 4438012 bytes
Fixed Size 48260 bytes
Variable Size 3971960 bytes
Database Buffers 409600 bytes
Redo Buffers 8192 bytes
ORA-00205: error in identifying control file, check alert log for
more info'
SQL> shutdown abort
ORACLE instance shut down.
> Copy the file.
SQL> startup
ORACLE instance started.
Total System Global Area 4438012 bytes
Fixed Size 48260 bytes
Variable Size 3971960 bytes
Database Buffers 409600 bytes
Redo Buffers 8192 bytes
Database mounted.
Database opened.
SQL>
```

Loss of all control files, cold restoration

If you lose all control files mirrors, you can easily recreate them if all the datafiles and the active redo log files are intact.

You may also want to recreate your control file to change certain sizing parameters which define the size of a control file.

If you want, for example to increase the maximum number of authorized datafiles in the database or change the size of the archived logs history. Perform a full hot database backup, close the database correctly then use the method described below.

To test the restoration procedures, you can simulate the loss of all your control files by removing them from the database or by renaming them.

Perform a restoration and a recovery of all control files

- 1 Use the **Oracle or SQL*Plus Server Manager** to stop the database. This database cannot be stopped correctly if some control files are missing.
- 2 Open the StoreWay DPA restoration interface on the target Oracle application then define, if necessary, a « time navigation » value.
- 3 Select the ControlFile_begin or ControlFile_end meta-object, or even both, and click **add to the cart**.
- 4 **Contents of the cart:** To restore click **restore**.
- 5 **Restoration option:** Click **restore** to start the process.
- 6 For security reasons, the controlFiles are restored to the config directory installation of the StoreWay DPA client agent (for example: **C:\Program Files\StoreWay DPA\config**) under the name: **ctl1ORCL_begin.ora** and **ctl1ORCL_end.ora**
- 7 Copy one of these files to the original Oracle directory by renaming it **CONTROL01.CTL**.
WARNING: You must also update any mirrored **CONTROL_FILE** files.
- 8 Use the Oracle or SQL*Plus Server Manager to perform a database recovery.

Loss of active members redo log, hot restoration

If you lose an active members redo log, the situation can be easily rectified if the active redo log files are copied to different disks.

If these logs are copied to another disk, you will have to perform a full database restoration. It is possible that you lose transactions, unless you have only lost one member which is not the active log, which has already been archived and which can be deleted using the Oracle or SQL*Plus Server Manager. In this case, do not try to restore it, simply delete it.

NOTE: For transactional databases: ensure your active redo log files are mirrored to separate disks, if possible with different controllers. This precaution will avoid all transaction loss.

This procedure restores an entire set of missing active redo logs by copying a healthy set from another disk.

To test the restoration procedures, you can simulate the loss of active members redo logs by removing them from the database or by renaming them.

Restore a set of active redo log members

- 1 Stop and restart the database. All missing redo logs now have the status "INVALID" but the database works.
- 2 Copy the active redo log files, group by group, to the right destination using the operating system command (such as cp or dd).
 - When the current redo log file is switched, the mirrored copies become useable again by Oracle as the MetaTree refresh shows.

Loss of all active redo logs, cold restoration

If you lose all active redo logs, you have lost your database. You must therefore perform a full database restoration and recovery.

NOTE: Check that your active redo log files are copied simultaneously onto separate disks preferably with different controllers, if the database is transactional, so as to avoid the loss of redo log files.

Cold restoration of the last backup

This method consists of restoring the complete database as it was before the incident. You must first synchronize the MetaTree within a time frame defined by the backup of a control file then restore a complete group of tablespaces using datafile instances including previous backups. You must then perform a database recovery by positioning in the timeframe and recreating the control files.

To test the restoration procedures, you can simulate the loss of a table by removing them from the database or by renaming them after performing a full database backup.

Perform a restoration and recovery of all active redo logs and the database

- 1 Use the Oracle or SQL*Plus Server Manager to stop the database. This database cannot be stopped correctly if some control files are missing.
- 2 Open the StoreWay DPA restoration interface on the target Oracle application.
- 3 Click the **ControlFile_begin** or **ControlFile_end** metaobject and select the database version you want to restore. All transactions after this date will be lost.
- 4 Select all the files to restore:
 - Redologs
 - Tablespace
 - Controlfiles
- 5 Click **add to the cart**.
- 6 Cart contents: To restore, click **restore**.
- 7 Restoration option : Click **restore** to start the process.
 - For security reasons, the controlFiles are restored to the config directory installation of the StoreWay DPA client agent (e.g.: **C:\Program Files\dpa\config**) under the name: **ctl1ORCL_begin.ora** and **ctl1ORCL_end.ora**
- 8 Copy one of these files to the original Oracle directory by renaming it CONTROL01.CTL.

WARNING: You must also update also the mirrored CONTROL_FILE files if there are any.

Example

Copy of the file :

C:\Program Files\dp\config\ctl11TESTDB_begin.ora
in C:\oracle\oradata\testdb\testdb, renamed en CONTROL01.CTL,
duplicated for CONTROL02.CTL and CONTROL03.CTL (file mirror)

> Use the Oracle or SQL*Plus Server Manager to perform the application database recovery operations.

EXAMPLE: Server manager commands

```
SQL>startup mount ;  
Instance ORACLE started.  
Total System Global Area 135338868 bytes  
Fixed Size 453492 bytes  
Variable Size 109051904 bytes  
Database Buffers 25165824 bytes  
Redo Buffers 667648 bytes  
Database mounted.
```

```
SQL>recover database using backup controlfile until cancel;  
ORA-00279: change 1543894 generated 08/09/2005 11:19:29 required  
for thread 1  
ORA-00289: suggestion : C:\ORACLE\ORA92\LOGS\ARC00026.001  
ORA-00280: change 1543894 for thread 1 is on sequence n° 26  
Indicate the log : {<RET>=suggested | namefile | AUTO | CANCEL}  
CANCEL;  
SQL>alter database open resetlogs;
```

WARNING: When the latest archive log is restored, enter **CANCEL**.

Chapter 7. Oracle Commands

This chapter describes the main Oracle commands.

Alter the Tablespace OFFLINE

```
SQL> alter tablespace « name of the tablespace » offline [normal  
| immediate] ;
```

WARNING: Requires immediate redo logs archiving.

Alter the Tablespace ONLINE

```
SQL> alter tablespace « name of the tablespace » online;
```

Stopping/restarting a database

```
SQL> shutdown [abort | immediate | normal | transactional]
```

Abort	Immediate shutdown by closing the database instance
Immediate	Cancels the active transactions and disconnects all connected users
Normal	Waits for all currently logged on users to log off from the database
Transactional	Disconnects all connected users once the transactions are terminated

The shutdown command will:

- > Close the database
- > Dismount the database
- > Stop the instance

List of files containing the different tablespaces

```
SQL> select * from dba_data_files
```

List of Datafiles files

```
SQL> select * from v$datafile;
```

Select the list of control files

```
SQL> select * from v$controlfile;
```

Select the list of redologs files

```
SQL> select * from v$logfile;
```

Set tablespace in backup mode

```
SQL> alter tablespace « name of the tablespace » begin backup
```

Return a tablespace to normal mode

```
SQL> alter tablespace « name of the tablespace » end backup
```

Force the switch of a current log

```
SQL> alter system switch logfile
```

Index

A

- architecture
 - Oracle 8
- archived redo log 46
 - hot restoration 46
- ARCHIVELOG mode 14
 - parameters 21
 - transition 23

B

- backup
 - cold restoration 39
 - ControlFile 34
 - hot 32
 - optimization 35
 - profile 33

C

- Circular Mode 14
- code
 - Oracle 12
- cold restoration
 - last backup 39
- ControlFile
 - backup 34
 - restoration 48

D

- database
 - full restoration 39
 - Oracle 9
- datafile
 - cold restoration 44
 - hot restoration 42
- Datafiles 13

H

- hot backup 32

I

- Index (INDX) tablespace 11
- INIT.ORA 18
- init.ora

- path 29
- Initialisation parameter file 18
- Installation
 - Oracle module 26

M

- meta-characters
 - Oracle 25
- meta-tree
 - handle objects 38

N

- NOARCHIVELOG Mode 15

O

- Oracle
 - architecture 8
 - code 12
 - commands 52
 - create StoreWay DPA application 29
 - Data Dictionary 13
 - database 9
 - meta-characters 25
 - trace files 19
- Oracle module
 - installation 26
 - overview 7

P

- Password Manager Application 27
- prerequisites
 - restoration 37
- profile
 - backup 33
- PWD.ora file path 29

R

- RDBMS configuration 18
- Redo log
 - archived files 14
 - configuration 16
 - files 14
- restoration 46
 - cold of last backup 39

- ControlFile 48
- datafile 44
- full database 39
- prerequisites 37
- procedures 38
- Tablespaces 41
- window 38

Rollback (RBS) tablespace 11

S

- SID (System Identification) 9
- spfile file path 29
- StoreWay DPA
 - Backup Module for Oracle overview 7
 - configuration prerequisites 20
 - create application 29

T

- Tablespaces
 - overview 10
 - restoration 41
- trace files 19

Technical publication ordering form

To order additional publications, please fill in a copy of this form and send it via mail to:

BULL CEDOC 357 AVENUE PATTON B.P.20845 49008 ANGERS CEDEX 01 FRANCE	Telephone: +33 (0) 2 41 73 72 66 FAX: +33 (0) 2 41 73 70 66 E-Mail: srv.Duplicopy@bull.net
--	---

Reference:	Designation:	Qty
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
__ - - - - - [__]		
[__] The latest revision will be provided if no revision number is given.		

NAME: DATE:
COMPANY:
ADRESS:

TELEPHONE: FAX:
E-MAIL:

For Bull Subsidiaries:

Identification:

For Bull Affiliated Customers:

Customer Code:

For Bull Internal Customers:

Budgetary Section:

For Others: Please ask your Bull representative.

**BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE
REFERENCE
DPA_ORACLE_V2_6_EN**

