Backing Up Oracle Using Tivoli Storage Management

Protecting your Oracle databases with TSM

Using hardware to back up Oracle databases

Implementing practical recovery scenarios

ibm.com/redbooks
Backing Up Oracle
Using Tivoli Storage Management

October 2001
Contents

Figures ......................................................................................................... xi
Tables .......................................................................................................... xiii
Preface ........................................................................................................ xv
The team that wrote this redbook .............................................................. xvi
Comments welcome ................................................................................... xvii

Part 1. Tivoli Storage Manager and database primer ................................. 1

Chapter 1. Tivoli Storage Manager for database administrators ............ 3
  1.1 Introducing Tivoli Storage Management ........................................ 3
     1.1.1 Tivoli Storage Manager backup/archive client ....................... 5
     1.1.2 Tivoli Data Protection for Oracle ........................................ 6

Chapter 2. Overview of relational databases and the Oracle product ..... 9
  2.1 Fundamentals of Relational Database Management Systems .......... 9
     2.1.1 Databases ........................................................................ 9
     2.1.2 Tables ........................................................................... 9
     2.1.3 Indexes ......................................................................... 10
     2.1.4 Tablespaces ................................................................. 11
     2.1.5 Partitioning options ....................................................... 13
     2.1.6 Log Files ...................................................................... 13
     2.1.7 Control files .................................................................. 15
     2.1.8 Configuration parameters ............................................. 15
  2.2 Oracle8i overview ........................................................................... 16
     2.2.1 Instances ...................................................................... 16
     2.2.2 Databases ..................................................................... 16
     2.2.3 Tablespaces .................................................................. 16
     2.2.4 Datafiles ..................................................................... 17
     2.2.5 Redo log. ...................................................................... 17
     2.2.6 Control files .................................................................. 18
     2.2.7 Initialization parameter files ......................................... 18
     2.2.8 Password file .................................................................. 18
     2.2.9 Partitioning option ....................................................... 18
     2.2.10 Parallel server option .................................................. 19

Chapter 3. Planning considerations ......................................................... 21
  3.1 Backup requirements ................................................................. 21
     3.1.1 Types of events ............................................................ 22
     3.1.2 Speed of recovery ......................................................... 24
Section 3.1.3 Backup windows ........................................... 24
Section 3.1.4 Recovery points ........................................ 25
Section 3.1.5 Units of recovery ........................................ 25
Section 3.2 Backup techniques ........................................ 25
Section 3.2.1 Disk mirroring .......................................... 26
Section 3.2.2 Offline backup .......................................... 27
Section 3.2.3 Online backup .......................................... 27
Section 3.2.4 Database export ....................................... 27
Section 3.2.5 Full database backup .................................. 28
Section 3.2.6 Partial database backup ................................. 29
Section 3.2.7 Incremental backup ..................................... 29
Section 3.2.8 Log file backup (simulated incremental) .............. 29
Section 3.2.9 LAN-free backup ........................................ 30
Section 3.2.10 Backup using split mirror features ................. 30
Section 3.2.11 Backup of RDBMS supporting files ................. 31

Chapter 4. TSM server considerations ................................ 33
Section 4.1 Initial requirements ......................................... 33
Section 4.2 Tivoli Storage Manager terminology .................... 33
Section 4.3 Similarities between TSM and database software ....... 34
Section 4.4 TSM server base functionality explained ............... 35
Section 4.4.1 Registering a node with the TSM server ............. 35
Section 4.4.2 Installing the TSM client ................................ 36
Section 4.4.3 Configuring the TSM client .............................. 36
Section 4.5 Managing data objects .................................... 37
Section 4.5.1 Archive or data object .................................. 37
Section 4.5.2 Description of data object ............................... 37
Section 4.5.3 Management class to bind object to .................. 40
Section 4.5.4 Life cycle of TSM data objects ........................ 44
Section 4.6 TSM server considerations for Oracle backups ......... 47
Section 4.6.1 How TDP for Oracle stores data objects ............ 47
Section 4.7 Policy management considerations ....................... 48
Section 4.7.1 Domain considerations .................................. 48
Section 4.7.2 TSM management class considerations ............... 49
Section 4.7.3 TSM client include-exclude option .................... 50
Section 4.8 Node considerations ........................................ 52
Section 4.8.1 Choosing a nodename .................................... 52
Section 4.8.2 Choosing a password .................................... 52
Section 4.8.3 Setting the BACKDELETE option ..................... 52
Section 4.8.4 Specifying the domain ................................... 53
Section 4.8.5 Setting the MAXNUMMP ................................ 53
Section 4.9 Storage pool considerations ................................ 53
Section 4.10 Our TSM server setup ..................................... 55
Part 2. TDP for Oracle on the UNIX platforms .................................................. 59

Chapter 5. Configuration and setup of TDP for Oracle on UNIX .......... 61
5.1 TDP for Oracle .................................................................................. 61
  5.1.1 Install TDP for Oracle ................................................................. 61
  5.1.2 Define environment variables ..................................................... 62
  5.1.3 Edit the client options files ....................................................... 64
  5.1.4 Registering with the TSM server ................................................. 66
  5.1.5 Initialize the password .............................................................. 66
  5.1.6 Setting up RMAN ................................................................. 67

Chapter 6. Oracle backup considerations on UNIX ......................... 69
6.1 RMAN: Oracle’s Recovery Manager .................................................. 69
  6.1.1 RMAN system components ..................................................... 69
6.2 Creating a database for the RMAN catalog ........................................ 73
  6.2.1 Creating the RMAN catalog .................................................... 73
  6.2.2 Registering the target database .............................................. 75
6.3 Back up the database using TDP for Oracle and RMAN ................. 76
  6.3.1 Full backup ............................................................................ 77
  6.3.2 Whole database backup ......................................................... 78
  6.3.3 Incremental backup ............................................................. 78
  6.3.4 Image copies ........................................................................ 80
6.4 Backup examples ............................................................................ 81
  6.4.1 Consistent whole database backup .......................................... 81
  6.4.2 Archived redo log backup .................................................... 83
  6.4.3 Incremental level 0 backup ..................................................... 87

Chapter 7. Day-to-day monitoring of Oracle backups on UNIX ........... 89
7.1 Automating database tasks .............................................................. 89
  7.1.1 Starting SQL*Net and RDBMS services at system reboot ............ 89
  7.1.2 Automating RMAN backups ................................................... 90
7.2 RMAN report, list, and crosscheck utilities ...................................... 91
  7.2.1 Report utility ...................................................................... 92
  7.2.2 List utility ........................................................................ 96
  7.2.3 Crosscheck utility ............................................................ 101
7.3 Automatic deletion of old backups ................................................ 101
7.4 List objects in the TSM database .................................................. 102

Chapter 8. Recovering Oracle Databases on UNIX ............................. 107
8.1 Restore operations ........................................................................ 107
  8.1.1 Consistent database recovery ................................................. 107
  8.1.2 Inconsistent database recovery ............................................. 107
8.2 Recovery Operation ................................................................. 108
8.3 Recovery examples ................................................. 109
  8.3.1 Datafile recovery ........................................... 109
  8.3.2 Complete recovery ......................................... 110
  8.3.3 Incomplete recovery ...................................... 112
  8.3.4 Duplicate database ........................................ 114

Part 3. TDP for Oracle on Windows 2000 .............................. 119

Chapter 9. Installation of TSM software on Windows 2000 .......... 121
  9.1 Registering a node on the TSM Server for TDP for Oracle backups . 121
  9.2 Stopping Oracle services ....................................... 122
  9.3 Installing TDP for Oracle for Windows .......................... 123
  9.4 Configuring the TDP for Oracle client options file .............. 130
  9.5 Generating the encrypted password ............................. 132
  9.6 Verify a TDP for Oracle for Windows RMAN backup .............. 133
     9.6.1 Starting RMAN from a command prompt .................... 133
     9.6.2 Viewing the orcagent.log ................................ 136
     9.6.3 Listing the backup piece from RMAN ...................... 137
     9.6.4 Where the backup is stored on the TSM server .......... 138
     9.6.5 Viewing the backup object in greater detail .......... 139
     9.6.6 Deleting the backup piece from RMAN .................... 139
  9.7 Installing the TSM backup-archive client ....................... 141
     9.7.1 Registering a node with the TSM server .................. 141
     9.7.2 Installing the TSM backup-archive client .............. 141
     9.7.3 Configuring the backup-archive client options file (dsm.opt) . 149

  10.1 RMAN: Oracle Recovery Manager .............................. 153
     10.1.1 RMAN system components ............................... 153
     10.1.2 Create the RMAN catalog ............................... 157
     10.1.3 Registering the target database ....................... 158
  10.2 Back up the database using TDP for Oracle and RMAN .......... 159
     10.2.1 Full backup .............................................. 159
     10.2.2 Whole database backup .................................. 160
     10.2.3 Incremental backup ..................................... 160
     10.2.4 Image copies ............................................. 162
  10.3 Backup examples .............................................. 163
     10.3.1 Consistent whole database backup ..................... 163
     10.3.2 Archived redo log backup ............................. 165
     10.3.3 Incremental level 0 backup ............................ 166
  10.4 File level backup using TSM ................................... 167
Backing Up Oracle using Tivoli Storage Management
Figures

1. User interaction with tables ........................................ 10
2. Using an index in a table ........................................... 11
3. Tables, tablespaces, and data storage ................................. 12
4. Log files .................................................................... 14
5. Tablespaces and datafiles .............................................. 17
6. Table partitioning .......................................................... 19
7. Parallel environment ...................................................... 20
8. Native support or Tivoli Data Protection interfacing with the TSM API ................................. 35
9. Archive object life cycle ................................................ 44
10. Backup object life cycle ................................................... 45
11. AIX and Sun Solaris over ethernet ....................................... 61
12. Relationship environment variable — file dsm.opt — file dsm.sys ........................................ 65
13. RMAN and TSM system components ................................... 72
14. backup sets and backup pieces created by RMAN ...................... 77
15. Non-cumulative incremental backups .................................. 79
16. Cumulative incremental backups ........................................ 80
17. Whole consistent backup, commandfile backup_offline_paris ......................... 82
18. Multiple log groups, archive log destinations, different tape drives in TSM ............................. 85
19. Windows Services tool ..................................................... 122
20. Stopping the Oracle services ............................................ 123
21. Downloading TDP for Oracle 2.1.10 if needed ......................... 124
22. First screen of TDP for Oracle for Windows 2.1.10 installation .......... 124
23. Type of install ............................................................. 125
24. Select destination directory ............................................. 126
25. Choosing the installation directory .................................... 127
26. Changed destination directory ......................................... 127
27. Select program folder ................................................... 128
28. Confirmation screen ....................................................... 129
29. Installation completed ................................................... 130
30. Open dsm.opt with a text editor ....................................... 131
31. Editing the client options file ......................................... 132
32. Generating the encrypted password .................................... 133
33. First steps to test a RMAN backup .................................... 134
34. TSM backup-archive client version and release ...................... 142
35. Levels of TSM backup-archive client code ........................... 142
36. Files available for latest level of TSM backup-archive client .................. 142
37. Location to Save Files screen ......................................... 143
38. Language selection ....................................................... 144
39. Welcome screen of InstallShield ...................................... 144
40. Installation directory selection ......................................... 145
41. Complete or Custom ................................................................. 146
42. Selecting the API SDK and Administrative client files ............... 147
43. Installation confirmation screen ............................................. 148
44. Successful installation confirmation ....................................... 149
45. Creating a new text document ............................................... 150
46. Editing the client options file using notepad.exe ........................ 151
47. RMAN and TSM system components ...................................... 156
48. Differential incremental backups .......................................... 161
49. Cumulative incremental backups .......................................... 162
50. Whole consistent backup, RMAN command ............................ 163
51. Whole consistent backup, commandfile backup_offline_beach ....... 164
52. Backup-archive client GUI ................................................... 167
53. Selecting the database files and starting the archive ............... 168
54. Viewing the archive ............................................................ 169
55. Changing the scheduler service to use a different account .......... 176
56. c:\oracle\admin\scripts\autodelete.cmd ................................ 185
57. c:\oracle\admin\scripts\delete.sql .................................... 186
58. Using Windows search to check orasbt.dll files .................... 200
59. FlashCopy concept ........................................................... 228
60. Implementation of split mirror ............................................ 229
61. Typical TDP for ESS configuration ...................................... 230
62. Showing hidden devices in Device Manager ............................ 249
63. Opening Non-Plus and Play Drivers entry ............................. 249
64. AdsmScsi Properties .......................................................... 250
Tables

1. Default installation directories ......................................... 62
2. TDP for Oracle environment variables ................................. 63
3. Nodes and servers ............................................................. 66
4. Default installation directories .......................................... 212
5. System environment ......................................................... 231
Preface

Tivoli Storage Manager is a full-function storage software product that addresses the challenges of complex storage management across distributed environments. It protects and manages a broad range of data, from the workstation to the corporate server environment. Tivoli Storage Manager provides:

- Centralized administration for data and storage management
- Efficient management of information growth
- Customized backup solutions for major groupware and database products

Using Tivoli Data Protection products, Tivoli Storage Manager provides data protection for a wide variety of applications, databases, and groupware, ensuring that data is safe and secure no matter where it is located or how it is stored. These products interface directly with the applications using their backup-certified utilities and interfaces, simplifying online backup and restore procedures. Among these is the Tivoli Data Protection for Oracle application.

This IBM Redbook is designed to assist Oracle Database administrators and system/storage administrators with Oracle backup solutions using Tivoli Storage Manager V4.1. The primary tool we used for backup and recovery is Tivoli Data Protection (TDP) for Oracle V1.1, and the book covers setup and configuration of TDP as well as day-to-day management examples. Operating environments covered by the project are AIX, Sun Solaris, and Microsoft Windows 2000. We provide an overview of relational database management systems and how to plan for protecting them.

This book will help you install, tailor, and configure Tivoli Storage Manager and Tivoli Data Protection for Oracle on UNIX and Microsoft Windows 2000 platforms in order to accomplish backup and restore of Oracle8i databases. RMAN commands and setup are covered in detail, and emphasis is placed on practical recovery scenarios.

Also featured are new solutions utilizing Tivoli’s hardware integration features for IBM’s ESS intelligent storage subsystem, as well as an update on using TDP for Oracle V2.2 new features.

Note: This redbook replaces relevant sections of the earlier redbook, Using ADSM to Back Up Databases, SG24-4335-03, by updating and re-branding the content of that book.
The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization San Jose Center.

Deirdre Hewgill is a Project Leader with the IBM International Technical Support Organization, San Jose Center, specializing in Tivoli Storage Manager. Before joining the ITSO this year, Deirdre provided EMEA level 2 technical support for Tivoli Storage Manager and related products. For the last 13 years, she has worked for IBM assisting Enterprise customers, primarily in support, but also in service roles.

Art Balingit is a DBA Technical Specialist with IBM, New Zealand, where he installs and supports Oracle and DB2 databases. He has 17 years of IT experience using a wide range of mainframe and client/server platforms. His experience includes 4 years of database design, implementation, administration, and support of Oracle and DB2 databases on UNIX systems and Windows NT. He holds a degree in Psychology from the Ateneo de Manila University (Philippines). His areas of expertise include transaction processing systems, analysis and design, and programming.

Martin Bruegger is an Advisory IT Specialist in Switzerland where he is working in the outsourcing department. He has more than 10 years of experience as a DBA in a production environment. He has worked at IBM for 11 years. His areas of expertise include installation and support of Oracle databases on UNIX and S/390 servers.

Wolfgang Postl is an I/T Specialist with IBM Austria, where he installs and supports RS/6000 and SP systems. He has 6 years of experience with DB2 and the RS/6000 products line. Wolfgang holds a degree in Mathematics from the University of Graz (Austria). His areas of expertise include system management, storage servers, and relational databases.

James Thompson is a Tivoli Storage Manager Level 2 Technical Support Engineer with IBM in Tucson, Arizona. He has 7 years of IT experience across multiple platforms and applications. He graduated Cum Laude with a degree in Computer Science from Utah State University. He is currently a Technical Support Team Lead for the TSM Tivoli Data Protection and API products. His areas of expertise include Tivoli Storage Manager, Windows, Novell, AIX, DB2, Oracle, and Storage Area Networks.
Thanks to the following people for their invaluable contributions to this project:

Charlotte Brooks  
Pat Randall  
International Technical Support Organization, San Jose Center  

Seema Guraraj  
Jawed Iqbal  
Neil Rasmussen  
Kavita Shah  
Thiha Than  
Cora Yang  
Tivoli Systems, San Jose  

Comments welcome  

Your comments are important to us!  

We want our Redbooks to be as helpful as possible. Please send us your comments about this or other Redbooks in one of the following ways:  

• Fax the evaluation form found in “IBM Redbooks review” on page 273 to the fax number shown on the form.  
• Use the online evaluation form found at ibm.com/redbooks  
• Send your comments in an Internet note to redbook@us.ibm.com
Part 1. Tivoli Storage Manager and database primer
Backing Up Oracle using Tivoli Storage Management
Chapter 1. Tivoli Storage Manager for database administrators

Tivoli Storage Manager (TSM) is an enterprise-wide storage management application for the network. This chapter gives functional details of the core elements of the TSM product, and serves as an introduction for database administrators who are considering using TSM as their database backup application.

1.1 Introducing Tivoli Storage Management

Tivoli Storage Manager is the core application of the Tivoli Storage Management solution set. It provides automated storage management services (including backup and restore, archive and retrieve hierarchical space management and disaster recovery) to multivendor workstations, personal computers, mobile laptops and servers of all sizes and operating systems, which are connected via WAN, LAN, and SAN. Tivoli Storage Manager includes these components:

- **Server**
  
  The server provides backup, archive, and space management services to its defined clients. The server maintains its own database and recovery log for information about Tivoli Storage Manager resources, users, and user data including all backed-up, archived and migrated files. The client data itself is stored in server-controlled entities called storage pools. These are groups of random and sequential access media that store backed-up, archived, and space-managed files.

  The Tivoli Storage Manager server is responsible for maintaining the integrity of client sessions, reliably receiving client data, storing client data in storage pools, and efficiently managing that data internally so that it can be restored or retrieved when required. You can set up multiple servers in your enterprise network to balance storage, processor, and network resources. Tivoli Storage Manager allows you to manage and control multiple servers from a single interface that runs in a Web browser (the enterprise console).

- **Administrative interface**
  
  This interface allows administrators to control and monitor server activities, define management policies for client files, and set up schedules to provide services at regular intervals. Administrative functions are available from an administrative client command line and from a Web browser interface. A server console is also available.
• **Backup/archive client**
  This allows users to maintain backup versions of their files, which they can restore if the original files are lost or damaged. Users can also archive files for long-term storage and retrieve the archived files when necessary. A command line interface, native GUI interface, and Web browser interface are available for the backup/archive clients.

• **Application program interface (API)**
  This allows users to enhance existing applications with backup, archive, restore, and retrieve services. When users install the Tivoli Storage Manager API client on their clients, they can register as client nodes with a Tivoli Storage Manager server.

The Tivoli Storage Management solution set also includes the following client programs:

• **Tivoli Data Protection for applications (application clients)**
  This allows users to perform online backups of data that is used by particular applications such as database programs. After the database initiates a backup or restore, the application client uses the API to interface to Tivoli Storage Manager. The Tivoli Storage Manager server then applies its storage management functions to the data. The application client can perform its functions while users are working, with minimal disruption. Tivoli Data Protection clients are available for Oracle, Informix, SAP R/3, Lotus Notes R4, Lotus Domino R5, MS Exchange and MS SQL Server.

• **Tivoli Space Manager (hierarchical storage management client)**
  This provides space management services for clients on some platforms. Tivoli Space Manager users can free client storage by migrating less frequently used files to server storage. These migrated files are also called space-managed files. Users can recall space-managed files automatically simply by accessing them as they would normally. You can learn more about Tivoli Storage Manager in *Tivoli Storage Management Concepts*, SG24-4877.

• **Tivoli Disaster Recovery Manager**
  This automatically generates a disaster recovery plan containing the information, scripts, and procedures needed to automate restoration to help ensure quick recovery of your data after a disaster. It automatically manages and tracks the media on which your data is stored, whether on-site, in-transit, or off-site in a vault, so your data can be easily located if disaster strikes.
1.1.1 Tivoli Storage Manager backup/archive client

The Tivoli Storage Manager backup/archive client is designed to back up and restore, archive, and retrieve client file system data. Therefore, the client can back up any non-database and database files. Tivoli Storage Manager clients use standard operating system functions to access files within file systems, but they are not required to understand any logical structure that might exist within a file. This is not a limitation specific to Tivoli Storage Manager; all other client backup products that work at the file system level operate similarly.

The way TSM sees Oracle and other database systems affects how they are backed up. Each database appears as an individual file on the server or client file systems. A Tivoli Storage Manager backup/archive client running on an Oracle server or client can back up and restore, archive, and retrieve entire Oracle databases. It cannot back up smaller increments such as tables.

Other than the issues of size and replication, using a Tivoli Storage Manager backup/archive client to back up Oracle databases is straightforward. Each database is a self-contained data file that is backed up and restored. Tivoli Storage Manager restore a database in its entirety because it is just a file for Tivoli Storage Manager. If a database is deleted or corrupted, it is a simple task for Tivoli Storage Manager to restore the most recent or any previous backup version of this database from the Tivoli Storage Manager server to the Oracle server or client.

The Tivoli Storage Manager backup/archive client, however, does not meet all requirements for an ideal storage management solution in a Oracle environment. Following are some of the drawbacks you might experience when using the Tivoli Storage Manager backup/archive client:

- Consider a 5 GB database that changes everyday. The Tivoli Storage Manager backup-archive client will back up the full 5 GB even if only a 2 MB document has changed. You can waste a lot of time and storage space using this strategy.
- Many databases need to operate twenty four hours a day, seven days a week, so they are in use all the time and a consistent backup cannot be taken. The alternative is to quiesce the Oracle database and take backups, but this would result in server unavailability, which is not good for business.
1.1.2 Tivoli Data Protection for Oracle

The TDP for Oracle application client provides an integrated solution for performing full backup and restore operations on Oracle databases. It is a client application that provides full backup of online databases and restore of full databases to the original or different location. It is supported on these platforms:

- Microsoft Windows NT and Windows 2000 (on Intel)
- IBM AIX
- HP-UX
- Sun Solaris

TDP for Oracle is not intended as a substitute for the standard Tivoli Storage Manager backup/archive client. TDP for Oracle cannot be used to back up or restore any non-database data, such as history files or any other system configuration files. Those files need to be backed up by the Tivoli Storage Manager backup/archive client. Therefore, the two client types work together to provide full data protection for your Oracle environment.

The TDP for Oracle application client and the Tivoli Storage Manager backup/archive client can run simultaneously on the same Oracle server, however, they are totally separate clients as far as the Tivoli Storage Manager server is concerned.

1.1.2.1 TDP for Oracle and RMAN

The Oracle Recovery Manager (RMAN) provides consistent and secure backup, restore, and recovery performance for Oracle databases. While the Oracle RMAN initiates a backup or restore, TDP for Oracle acts as the interface to the TSM server (Version 3 or above). The TSM server then applies administrator-defined storage management policies to the data.

- TDP for Oracle provides the following actions and operations:
  - Full backup function for the following while online or offline:
    - Databases
    - Tablespaces
    - Datafiles
    - Archive log files
    - Control files
  - Full database restore while offline
  - Tablespace and datafile restore while online or offline
  - Change of TDP client password
The TDP for Oracle application client provides a command line interface for performing backups and restores. The application client commands are issued from a command prompt.
Chapter 2. Overview of relational databases and the Oracle product

In this chapter, we explain Relational Database Management System (RDBMS) concepts. Our discussion is directed towards system administrators or TSM administrators who need to know the basics of RDBMS in order to implement an RDBMS backup strategy with their database administrators.

Fundamentals of relational databases common to all RDBMSs are discussed first. This is then followed by concepts specific to the Oracle8i server products.

2.1 Fundamentals of Relational Database Management Systems

RDBMSs share a common set of principles. The purpose of this section is to explain the basic principles that a systems administrator or TSM administrator needs to understand in order to design a backup and recovery system for a relational database. Please note that, although all RDBMS products are based on the same set of principles, not all use the same terminology or structures. For example, the tablespace concept does not exist on some RDBMSs.

2.1.1 Databases

A database presents data as a collection of tables, where a table consists of a defined number of columns and any number of rows. A database can include a data dictionary or a set of system tables that describe the logical and physical structure of the data, a configuration file containing the parameter values allocated for the database, and a recovery log with ongoing transactions and archivable transactions. Some RDBMSs use control files as an extension of the data dictionary.

2.1.2 Tables

A table consists of data logically arranged in columns and rows. Figure 1 shows that tables are assigned to tablespaces and that users interact with tables. Table data is accessed through Structured Query Language (SQL), a standardized language for defining and manipulating data in a relational database. The data in the table is logically related, and relationships can be defined between tables.
2.1.2.1 Data dictionary
It is common for RDBMS to maintain a data dictionary in a set of system tables. These tables describe the logical and physical structure of the data. They are like any other tables, but are owned by the database administrator or by the database. They are created either when the database is created or when the database administrator runs a set of scripts supplied by the RDBMS. These tables contain information about the definitions of database objects, such as user tables and indexes, as well as security information about the authority that users have on these objects.

2.1.3 Indexes
An index is a set of keys, each pointing to rows in a table. For example, table A in Figure 2 has an index based on the employee numbers in the table. This key value provides a pointer to the rows in the table: employee number 19 points to employee KMP. An index allows more efficient access to rows in a table by creating a direct path to the data through pointers. In certain circumstances, the data storage of an index can be larger than that of the table it refers to.
2.1.4 Tablespaces

Tables and indexes are assigned to tablespaces as shown in Figure 3, which also shows that one or more data storage units can be allocated to tablespaces. Tablespaces cannot share data storage. Indexes of tables they refer to can be assigned to a separate tablespace for efficiency purposes. Normally, the data dictionary tables reside in their own tablespace.
Tables, tablespaces, and data storage

Tables, tablespaces, and data storage

Figure 3. Tables, tablespaces, and data storage

Tablespaces are simply logical concepts utilized by RDBMSs. They provide a convenient way of separating the user’s view of data from some of the practical considerations associated with storing that data on disk. For example, a database administrator can make more disk space available to several tables by adding disk space to the appropriate tablespace, thus ensuring that tables do not run out of space and that disk space is used efficiently. Furthermore, the tablespace concept means that neither users nor application programs need to be aware of the fact that the database administrator has made more disk space available.

Data storage in tablespaces can be implemented using either data files or directories on file systems or raw devices. (For information on file systems and raw devices, please see your operating system documentation.)

Tablespaces provides the link between logical views and data storage. Here are some points to note:

- The data for a table or index may be contained in only one data storage.
- Alternatively, the data for a table or index may be spread over several data storage.
- Each of the data storage may contain data for one or more tables in the tablespace.
- Each data file or directory may reside in a separate file system.

The significance of these alternatives is that the only way to back up or recover individual tables is by using the facilities that the RDBMS provides.
Normally, you would back up or restore tablespaces instead of the individual data storage of the tables. This ensures that all data storage for a tables is backed up consistently with the same timestamp.

You would use tablespace backup instead of full database backup depending on the volatility and/or importance of data. You have the option of backing up tablespaces which have more update activity more often than tablespaces, which have less activity.

The tablespace where the data dictionary tables resides is the most important tablespace. You must ensure that this tablespace is backed up successfully. Corruptions in the data dictionary can cause the database to become unusable.

### 2.1.5 Partitioning options

RDBMSs may provide partitioning options to handle very large amounts of data. This will allow workload parallelization of very large objects, and will also allow for the manipulation of subsets of these large objects. You should investigate whether partitioning is used in your database and know how partitioning options are implemented by your RDBMS. This can affect your backup strategy.

### 2.1.6 Log Files

As shown in Figure 4, most RDBMSs maintain details of the updates to databases in log files. If, for some reason, a transaction that makes a change to the database fails to complete successfully, the RDBMS' recovery procedure will use the log file to detect that an update may be only partially complete and to undo any changes that the transaction had made to the database.
Some RDBMSs support the use of log files to perform forward recovery. Forward recovery takes advantage of the fact that log files hold details of all changes that have been made to the database, and therefore you do not necessarily need to undo changes but instead can reapply changes. With forward recovery, the recovery process can:

- Restore a database to the state it was in at the time the last backup was taken
- Use the log files to reapply the changes that had been made since the last backup was taken
- Back out (undo) any partially completed changes.

A standard RDBMS concept related to log files is the checkpoint process. All RDBMSs use buffers in memory to hold changes to the database and log files. The purpose of buffers is to improve the operational performance of the RDBMS. However, the use of buffers means that most changes to databases and log files do not get written to disk until some time after the RDBMS has indicated to the user application that the update has been made successfully.

Checkpoints ensure that all database and log file changes held in the RDBMS' buffers are flushed out to disk. This shortens the time it takes to recover a database after a system crash because the number of redundant log records processed during the recovery is reduced. All RDBMSs support checkpoints and issue them automatically at intervals.

It is recommended that log files be mirrored or duplexed.
2.1.7 Control files

Some RDBMSs maintain control files to hold additional information about the physical structure of the database, such as which physical files are used by each tablespace and which one is the current log file.

For those RDBMSs that use control files, you need to define policies for backing up those files. It is recommended that these files also be mirrored or duplexed.

2.1.8 Configuration parameters

All RDBMSs provide a range of options. Some are set permanently, and others can be modified even when a database is in use (running). Some options allow you to tune the performance of the database; others allow you to specify how you want logging, for example, to be implemented. Depending on the RDBMS, you can either change the configuration parameters using database commands or modify them in an initialization file. The configuration parameters may be stored in a file, as in the case of an initialization file.

For RDBMSs which use initialization files, a database may have multiple initialization files. One reason for having multiple files for a single database might be to optimize performance for different circumstances. For example, you may decide to allocate one set of values when the database is used for batch processing and another set when it is used for online transactions. Although some of the options are set differently for each situation, many will be the same. Some RDBMSs allow you to specify options that are common to multiple initialization files in configuration files. Instead of repeating all options and their values in each of the initialization files you can select the configuration file that contains the options that you want to use.

You need to define policies for backing up both initialization files and configuration files.
2.2 Oracle8i overview

The previous section has provided a generic overview of databases for system administrators responsible for designing a backup and recovery system for data held on a relational database. The purpose of the current section is to introduce Oracle version 8.1.7 concepts that can help system administrators or TSM administrators to implement a backup strategy that is specific to Oracle.

Common concepts like tables and indexes already mentioned in 2.1, “Fundamentals of Relational Database Management Systems” on page 9 are no longer mentioned in this section. Rather, we concentrate on concepts that are new or specific to Oracle.

2.2.1 Instances

An instance, when started, allocates its own memory called the system global area (SGA) and starts a set of background processes. The SGA is used for database information that is shared by database users. Typically, there is one instance per database. In a parallel environment, the database will be shared by multiple instances. See 2.2.10, “Parallel server option” on page 19 for more discussion on parallel processing.

2.2.2 Databases

A database would consist of one or more logical units called tablespaces. On the physical layer, the database will have datafiles, control files, and optionally a password file.

2.2.3 Tablespaces

When a database is created, a data dictionary is created in the system tablespace. Although it is not required to create additional tablespace, additional tablespaces are recommended for user data. The data dictionary is critical to the operation of the database because it records, verifies, and conducts ongoing work. The system tablespace is always online and cannot be taken offline because the data dictionary must always be available to Oracle. We recommend that you reserve the system tablespace for use by Oracle; do not create any user tables in this tablespace.

You must ensure that the system tablespace is backed up successfully every time you do a database backup.
### 2.2.4 Datafiles

A tablespace is a logical grouping of data storage called datafiles. A datafile can be a file or a raw device. A tablespace can have a mixture of both files and raw devices as datafiles. Figure 4 shows the relationship between tablespaces and datafiles.

Backup can be performed on a logical level (database and tablespaces) or on a physical level (datafiles).

#### Figure 4. Tablespaces and datafiles

### 2.2.5 Redo log

The redo log is set of files that record all changes made to the database so that in the event of a failure database updates are not lost. The redo log consists of two parts: the online redo log and the archived redo log. The online redo log consists of the current log files that are being used to record database changes. The online redo logs are reused in a circular fashion. When the last log file is full, the first log file is reused.

Optionally, filled online redo logs can be archived before being reused. If you have archived redo logs, the database can be recovered from both instance and disk failures (instead of only instance failures), and the database can be backed up while it is open. To use archived redo logs, you must run with the `ARCHIVELOG` mode set to on.
Online redo logs are commonly multiplexed at the database level where at least two copies can be maintained on different disks. Backing up the online redo logs is only required when the database is not in **ARCHIVELOG** mode.

### 2.2.6 Control files
Control files keep information about the physical structure of the database and log files. They are commonly multiplexed and are defined in the initialization parameter files. You should keep at least three copies on separate disks. Just like the data dictionary, it is important to make backups of your control files regularly. Losing the control files makes recovery much more complicated.

### 2.2.7 Initialization parameter files
The initialization parameter files are text files that contain instance configuration parameters, such as how much memory to use and what to do with filled online redo logs. You should back up the initialization parameter files whenever configuration parameters change.

### 2.2.8 Password file
A database can optionally have a password file. The password file is used during remote administration of a database server. The password file needs to be backed up when there are changes or additions of administrative users.

### 2.2.9 Partitioning option
The Enterprise Edition of Oracle has a partitioning option for tables and indexes. When a table or index is partitioned, the table or index can be assigned to more than one tablespace. The table or index is distributed across multiple tablespaces using a partitioning key or a hashing algorithm. An example can be a history table shown in Figure 6 where current data which is often queried is placed on a tablespace assigned to a faster data storage device, and old data placed on tablespaces assigned to slower devices.
When planning for tablespace backup with partitioned tables or indexes, consider the dependencies involved with the other tablespaces that contain the partitioned tables or indexes.

### 2.2.10 Parallel server option

With the Enterprise Edition, parallel processing is possible with the use of the Oracle Parallel Server (OPS) option. OPS employs the use of a collection of interconnected hardware called clusters. A cluster can have two or machines or nodes. Each node would run an instance, and each instance would have its own redo log. The datafiles and control files should be accessible to all nodes because all instances access the same datafiles. Figure 7 shows an example using OPS.
You can do offline or online backup with OPS from any node. If not in ARCHIVELOG mode, you can only do offline backup, and all instances must be shut down to close the database.
Chapter 3. Planning considerations

Planning is one of the most important areas for consideration before beginning to use Tivoli Storage Manager for database backups. It is important that the database administrator and the Tivoli Storage Manager administrator work together to anticipate the circumstances in which recovery will be required, as well as the resource and configuration requirements. These ideas apply to all types of databases.

This chapter will include details of some possible data recovery situations. We also cover the factors which should be weighed against one another in planning for recovery, for example, type of database, backup windows and relative speed of backup and recovery methods. Finally, we introduce various backup methods, which are covered in more detail in later chapters.

3.1 Backup requirements

A backup strategy is only one part of your overall data management plan. You must consider how important your data is to the function (or even existence) of your organization. The less time that your organization can function without its data the more important that data is to you. Your system must be designed in such a way that it keeps important data available when a failure occurs. Reliance on backups is not necessarily sufficient. You should also consider the following:

- Redundant Arrays of Inexpensive Disks (RAID) devices
- Dual access paths
- Dual I/O controllers
- Dual power supplies
- Backup or standby processors
- Uninterruptable power supplies

None of these on their own can guarantee the availability of your data but in combination they can reduce the impact of a failure.

Before you design a backup strategy, you need to define the requirements that the strategy must satisfy. These are some factors to be considered when defining the requirements for your backup strategy:

- Types of events (the categories of incidents that may occur)
- Speed of recovery (how quickly you need to be able to recover)
- Backup windows (the periods of time at which backups can be performed)
- Recovery points (to which points in time you need to be able to recover)
• Units of recovery (which other tables and files need to be recovered to the same point in time)

Let us look at each of these factors in more detail.

3.1.1 Types of events
We identify five categories of events that may require data recovery:

• User error
• Statement failure
• Transaction failure
• Media failure
• Disaster

Let us look at each category in more detail.

3.1.1.1 User error
There is considerable opportunity for a user to make an error that causes data to be lost. For example, a user might inadvertently delete or update rows in a table or accidentally drop an entire table; or a programmer might make a logic error that results in data loss or corruption.

RDBMSs provide facilities that reduce the risk or impact of user errors. For example, you can use RDBMS security to restrict the data that individual users can access or update. However, it is not possible to eliminate the risk entirely, and you need to consider how to handle such situations.

One approach is to say that it is the user's responsibility to recover from such errors. This approach may not be acceptable to users or their management, however. Another approach is to restore the entire database to the point in time at which the last backup was taken. This may not be satisfactory to other users who will lose the updates that they have made to the database since the last backup.

A third approach is to restore the table space that contains the damaged table. This approach is likely to be more acceptable than the other two because:

• It removes the responsibility for data recovery from the users.
• It may impact fewer users. The number of users impacted will depend partly on the number of tables included in the affected table space.

You may, however, need to be able to restore individual tables, in which case you need to have backed up the tables individually.
3.1.1.2 Statement failure
SQL statements that are syntactically correct may fail because, for example, the database is full. RDBMSs will usually detect such problems, roll back the effects of the failing statement, and report the problem to the user. Once the fundamental cause of the problem has been resolved, the user can retry the statement and continue to work. There is normally no need to take any special action to recover from SQL statement failures.

3.1.1.3 Transaction failure
Transactions may fail for a variety of reasons:
- Programming errors
- Network failures
- Failures of the operating system or RDBMS
- Power failures

The actions required to recover from these situations vary according to the particular circumstances. However, the RDBMS will ensure that the integrity of the data it manages is preserved. You do not need to restore data to recover from transaction failures.

3.1.1.4 Media failure
RDBMSs normally use magnetic disk as the medium on which they store the data that they manage. If a disk volume is physically damaged or destroyed, at a minimum, you need to restore the data files that have been lost to the state they were in when they were last backed up.

3.1.1.5 Disaster recovery
Many organizations have developed plans for recovery from disasters such as floods, fires, accidents, earthquakes, and terrorist attacks. You need to ensure that your strategy for backing up and recovering data fits in with any such plans. For example, you may need to arrange for backups to be made to a removable medium and stored off site. Disaster recovery is too broad a subject to address in this book and is not discussed in any more detail.

Almost all RDBMSs provide the facilities necessary to bring databases up to date by applying log files. They also provide the facilities necessary to undo changes made by partially completed transactions. This means that designers of database backup and recovery solutions do not need to concern themselves with recovering database data after statement failures or transaction failures.
For each type of event that may occur, designers of a database backup and recovery solution must:

- Ensure that operational procedures specify who needs to do what, in order to recover from loss or corruption of data used by the RDBMS.
- Ensure that the data files that the RDBMS recovery routines use are available when needed.
- Ensure that any data which the RDBMS does not manage can be recovered to a state that is consistent with the database.

### 3.1.2 Speed of recovery

If you ask users how quickly they would like you to be able to recover lost data, they usually answer "immediately". In practice, however, recovery takes time. The actual time taken depends on a number of factors, some of which are outside your control (for example, hardware may need to be repaired or replaced). Nevertheless, there are certain things that you can control and that will help to ensure that recovery time is acceptable:

- Develop a strategy that strikes the right balance between the cost of backup and the speed of recovery.
- Document the procedures necessary to recover from the loss of different groups or types of data files.
- Estimate the time required to execute these procedures (and do not forget the time involved in identifying the problem and the solution).
- Set user expectations realistically, for example, by publishing service levels that you are confident you can achieve.

### 3.1.3 Backup windows

Some RDBMSs do not allow databases to be backed up while they are in use. In such cases, you need to shut down the database before the backup starts, and you cannot restart the database until after the backup has completed.

Shutting down a database often means that users cannot use applications. You need to ensure that the times at which databases are shut down and unavailable are acceptable to your users.

Even if you can perform backups while the database is operational, you need to ensure that any load on processors or networks caused by the backup process does not result in performance or responses that are unacceptable to your users.
3.1.4 Recovery points

You need to define the points in time to which you will restore data. For example, you may need to recover the data to the state it was in when the last transaction was completed. Alternatively, it may be acceptable to restore the data to a consistent state that is no more than 24 hours old. In addition to either of these, you may be required to restore individual tables to the state they were in at any particular date within the last 30 days.

Whatever your situation, you need to consider recovery points and define a policy that is both achievable and acceptable to your user community.

3.1.5 Units of recovery

In some circumstances, it may not be sufficient to restore individual tables (or even entire databases) to the state they were in at some point in the past. Sometimes, in order to maintain data consistency, you may need to restore data held in tables or files that have not been lost or damaged. This undamaged data needs to be restored to the same point in time as the damaged data.

In developing your backup strategy, you need to understand the relationships between the data objects on which user applications rely. Many applications rely on relationships that extend beyond the data held in a single database. For example, an engineering database application holds references to documents that exist as independent file system files. If the engineering database is lost and restored to the point in time at which the last backup was taken, references to documents may be lost. Alternatively, the medium on which some of the documents are stored may be damaged. If the data files used to hold the documents are restored to the point in time at which the last backup was taken, the engineering database may contain references to documents that do not exist.

There are many other situations where you need to ensure that data consistency is preserved. The key point is that your backup and recovery strategy must take into account the needs of the applications that use the data.

3.2 Backup techniques

You can use any number of techniques to back up data managed by RDBMSs. These techniques are, at least at a conceptual level, common to most RDBMSs. The purpose of this section is help you understand the techniques well enough to design an appropriate backup strategy.
Often, a combination of techniques is used. The techniques we consider are:

- Disk mirroring
- Offline backup
- Online backup
- Database export
- Full database backup
- Partial database backup
- Incremental backup
- Log file backup
- LAN-free backup
- Backup using splitcopy features
- Backup of RDBMS supporting files

### 3.2.1 Disk mirroring

Disk mirroring is a useful technique for maximizing the availability of your database. Mirroring is the process of writing the same data to multiple storage devices at the same time. This is done either sequentially, when data is only written to the mirror once the master write is successful or in parallel when both master and mirror writes occur at the same time. The first method is slower but you are more likely to have at least one good copy of the data if a failure occurs.

When reading from a mirrored logical volume, AIX will read from either the master or the mirror, whichever is the quickest at the time.

If a media failure occurs, operations are automatically switched to the good copy and AIX marks the faulty copy as stale. Mirroring allows your users to continue working even though a media failure has occurred. Mirroring can be implemented in either software or hardware.

However, mirroring does not remove the need to back up databases. For example, disk mirroring will not allow you to restore a table that has been lost or damaged as a result of user error. Also, although disk mirroring dramatically reduces the impact of media failures, there is still a risk of damage to both sides of the mirror. If a database is held on one set of physical volumes, and a mirror image of the same database is maintained on a separate set of physical volumes, it is possible for both sets of physical volumes to be damaged or destroyed. This could happen as a result of a disaster or it could just be bad luck. In such instances, it will be necessary to recover the database from backup copies.
Oracle provides multiplexing of redo logfiles and control files, and allow for multiple destinations of archive log files. It is recommended to use this feature as an alternative to mirroring these files.

### 3.2.2 Offline backup

Offline backup involves shutting the database down before you start the backup and restarting the database after backup is complete.

Offline backups are relatively simple to administer. However, they suffer from the obvious but significant disadvantage that neither users nor batch processes can access the database while the backup is taking place. You need to schedule sufficient time to perform the backup to ensure that the periods when the database will be unavailable are acceptable to your users.

Some RDBMSs provide a "single-user mode" or "quiesced mode". You can think of this as an "almost offline" mode. A database administrator can still use the database, but general users cannot.

### 3.2.3 Online backup

Some (but not all) RDBMSs allow backups to be performed while the database is started and in use. Clearly, if a database is being backed up while users are updating it, it is likely that the data backed up will be inconsistent ("fuzzy"). The RDBMSs that support online backup use log files during the recovery process to recover the database to a fully consistent state. This approach requires that you retain the RDBMS log files and indicate to the RDBMS when you are about to start the backup and when you have completed the backup.

Some RDBMSs allow you to quiesce activity on portions of the database (for example, a particular table space) so that a set of complete tables is temporarily "frozen" in a consistent state. You then can back up the set of tables that has been "frozen". Once the backup has completed, you can reactivate the table space.

### 3.2.4 Database export

All RDBMSs provide export and import utilities. These utilities operate on logical objects as opposed to physical objects. For example, you can use an export command to copy an individual table to a file system file. At some later time, you might want to restore the table, in which case you would use the import command. Although export and import can be used for backup and restore operations, they are really designed for moving data, for example, for workload balancing or migration.
Most other utilities operate on the physical data files that RDBMSs use to store their databases. Therefore, other utilities cannot normally be used to back up and restore a single table because:

- A single physical data file may contain data belonging to several tables.
- The data contained in a single table may be spread across multiple data files.

Thus, the only way to gain access to the set of data contained in a single table is through the RDBMS itself.

Export utilities are usually slower than most other utilities and should be used only when you need access to database objects or raw devices.

### 3.2.5 Full database backup

Full database backups involve making copies of:

- Data files used to hold user data
- Data files that hold tables used by the RDBMS itself
- RDBMS log files
- Any control files and parameter files that the RDBMS uses

Many RDBMSs allow you to perform full database backup when the database is either online or offline. However, the technique for full database backup when the database is online can be quite different from offline.

To perform offline backup, you can use the operating system utilities, RDBMS utilities, or TSM to back up the data files that constitute the database. To perform online backup, you need to use an RDBMS utility to create data files containing a copy of the database. You can then use TSM to back up these data files along with the parameter files that you use to start up the RDBMS.

The simplest approach to database backup is to perform only full, offline backups at regular intervals. This approach is relatively easy to administer, and recovery is relatively straightforward. However, it may not be practical to take databases offline for the period of time that is necessary to perform full backups at the frequency you need. You may have to adopt a more flexible approach.

---

**Note**

You should assume that import will work only with files that have been created by the same RDBMS's export utility.
3.2.6 Partial database backup

Many RDBMSs allow partial database backups when the database is online or offline.

Partial database backups involve backing up a subset of the full database (such as the data files that make up a table space). Make sure that the subset you back up (as part of a partial backup) represents a complete logical unit of recovery from the point of view of the application. You may also need to back up data files that the RDBMS does not manage.

You must also ensure that the unit of recovery is consistent from the point of view of the RDBMS. If you have added a new data file to a table space, you must ensure that any control file that the RDBMS uses to define the relationship between data files and table spaces is also backed up.

3.2.7 Incremental backup

Some RDBMS provides backing up those data that has changed since the last offline or online database backup. This will save tape or disk space but may or may not reduce the time to do a backup because the RDBMS still need to read the data blocks to determine if it has changed since the last backup. When recovery is needed, the database backup and incremental backups are required to fully recover the database. This can take more time to recover a database. Incremental backups are useful when saving space or when saving bandwidth when backing up over the network.

3.2.8 Log file backup (simulated incremental)

For some applications, the units of recovery are too large to be backed up on a daily basis. Sometimes the constraining factor is the elapsed time that is available (the backup window). Sometimes the load that the backup would place on the network would have an unacceptably bad impact on other processes and users.

In such situations it may be possible to capture only the changes to the database by backing up the RDBMS' log files. Some database vendors refer to log file backup as incremental backup, but it is really a "simulated" incremental backup as opposed to a "true" incremental backup. A true incremental backup backs up changed database blocks or pages, whereas a simulated incremental backup backs up the database transactions. Recovery from a simulated incremental can be much longer than from a true incremental because you must reapply all of the transactions in the logs.
This is how to recover from a log file or simulated incremental backup:

1. Restore the database from a full database backup (in some circumstances, restoring from a partial backup may be sufficient).
2. Restore the log files.
3. Apply the log files to the restored database.

### 3.2.9 LAN-free backup

Normally, a database backup has to go over the LAN to the storage destination and may impact the users or applications using the same LAN. One way to overcome this problem is to use a dedicated LAN for backup so that the data transfer will no longer interfere with the work of other users and applications.

However, even when using a dedicated LAN all the data and metadata (file permissions, owner...) must still be handled by the application that will receive the backup data. This application will need resources such as CPU, memory or disk space to buffer and manage data an metadata. To free the application from the work needed for handling the backup data itself a LAN-free solution can be used.

LAN-free means that a group of machines are able to share the same storage devices over a high performance connection. LAN-free also provides an easy way of defining storage devices to machines without much cabling effort, as the devices all communicate over the same high performance connection.

In our context, LAN-free can be used as follows: The clients can send their backup data files directly to the tape library (LAN-free) and only send the metadata information on where the file resides to the storage application.

### 3.2.10 Backup using split mirror features

A backup may potentially degrade the performance of a production system. In a 24x7 environment or with very large databases, it is particularly hard to schedule a backup such that it will not interfere with the normal operation. In order to free the production system from the overhead of backup, it is valuable to have a copy or mirror of the database for backup, reporting, or other purposes.

Some intelligent storage servers, such as IBM ESS, support the split mirror feature. Split mirror means that identical and independent copies of disk volumes can be established within those storage servers. These copies can normally be established in a very short time (a maximum of 5-20 seconds dependent on vendor).
If the database resides on such a storage server that supports the split mirror feature, a copy of the disk volumes can be established and assigned to another (backup) machine. On the backup machine, the (backup) database can be accessed exclusively for backup or other purposes.

One important requirement is that the data on the disk volumes is consistent during the creation of the copy volumes. One way to establish this is to shut down the database and to synchronize all the data that may reside in the memory of the operating system to disk. After the split mirror is established the database can be started again.

If the database cannot be stopped, then the database itself must provide features to ensure that the data on the disk will be in a consistent state at the time of establishing the split mirror volumes.

### 3.2.11 Backup of RDBMS supporting files

Most RDBMSs require certain files to operate, but do not back them up when using their backup utilities. These files can be initialization parameter files, password file, files that define the environment, or network configuration files. They are external files and are not part of the database because they must be accessible for reading or editing, even when the database is down. As in the case of the password file, it provides authentication in order to administer a database, especially for starting up a database from a remote site.

You must ensure that these files are also backed up using operating system or third party tools like TSM.
Chapter 4.  TSM server considerations

Tivoli Storage Manager server is an advanced storage management tool. The TSM server provides data storage and retrieval services to TSM client programs, which can be native TSM backup-archive client, Tivoli Data Protection clients, or other 3rd party products that interface to the TSM server using the TSM Application Program Interface (API). In this chapter, we discuss all the steps, concepts, and considerations necessary to configure your TSM server to properly manage Oracle backups.

4.1 Initial requirements

The chapter assumes that you already have a TSM server installed and want to perform the steps to configure it to correctly manage Oracle database backup objects. This chapter also assumes that you have storage available and know how to define it to the TSM server (tape or disk) or that you are going to use existing storage pools to store the Oracle backups.

If this is a completely new installation, you will first need to perform the initial configuration of the TSM server. Assistance with this can be found in the TSM server product documentation and other ITSO Tivoli Storage Manager redbooks. Once this is complete, you can configure your TSM server to enable backup of the Oracle databases by defining your storage policies and registering nodes.

4.2 Tivoli Storage Manager terminology

The following terminology is going to be used.

**TSM server:** A computer where the TSM server program has been installed. The TSM server is responsible for managing data objects sent by a TSM client. A TSM Administrator allocates disk and tape storage to the TSM Server.

**TSM client:** A computer where a TSM client program has been installed. A TSM client can be a backup-archive client, a Tivoli Data Protection client (TDP), or a 3rd-party product that uses the TSM Application Program Interface (API).

**Data object:** A copy or backup of information sent to the TSM server by a TSM client. This can be a file, directory, database, database tablespace, database log, and so on.
**TSM backup-archive client:** Tivoli software program that allows data objects to be sent to a TSM server. This typically consists of the files and directories on a computer, but can include other data objects (that is, Windows 2000 System Object, UNIX Raw Logical Volume)

**TSM Application Program Interface (API):** Tivoli software that programmers use to interface with the TSM server. The TSM API is used by both Tivoli Data Protection products and other vendor products to send data objects to the TSM server.

### 4.3 Similarities between TSM and database software

In a computing environment, machines are typically referred to as either workstations or servers. The difference between these types of machines is that the servers are running software that provides services to other machines, the workstations.

An Oracle or DB2 server is a computer that is running the database server code. When the database server code is running, you can connect to the database using an administration client or an application client. The administration or application client can be running on the same machine as the database server or on a different machine.

Likewise, a TSM server is a computer that is running the TSM server code. When the TSM server code is running you can connect to the TSM server using an administration client, a backup-archive client, or an API client. The administration client, backup-archive client, or API client can be running on the same machine as the TSM server or on a different machine.

The purpose of the database server is to provide database services to an application client. For databases, the base functionality provided by the database server is viewing, adding, changing, or deleting rows or cells from tables. All other functionality hinges off this base functionality.

The purpose of the TSM server is to provide storage services to a backup-archive client or API client. The base functionality provided by the TSM server is storage of a data object and retrieval of that data object.

A database administrator (DBA) uses an administrative client to administer the database. Typical tasks include creating new databases, adding additional storage space to tablespaces, backup, recovery, daily monitoring, and managing authorization.
A TSM administrator also uses an administrative client to administer the TSM server. Typical tasks include backups of the TSM database, adding additional storage to storage pools, daily monitoring, and managing authorization.

### 4.4 TSM server base functionality explained

The TSM server base functionality is to provide storage and retrieval of data objects (that is, files, directories, databases tables, and so on).

In order to store and retrieve objects at an application level, the application needs to provide native support for TSM. Or a TDP product needs to interface between the application and the TSM API. (See Figure 8.)

![Figure 8. Native support or Tivoli Data Protection interfacing with the TSM API](image)

There are three basic steps that must take place before you can use the TSM server. These steps will be covered in greater detail in subsequent chapters:

1. Register a node with the TSM server.
2. Install the TSM client (backup-archive client, TDP or API client).
3. Configure the TSM client.

Following is a brief look into why these steps are necessary.

#### 4.4.1 Registering a node with the TSM server

Before a TSM client can use the storage services of a TSM server, it must first authenticate to the TSM server using a nodename and password. A node is created by the TSM Administrator using the `REGISTER NODE` command.

When the node is registered, it is given six key values that play an important role for performing backups and restores. `NODENAME`, `PASSWORD`, `DOMAIN`, `ARCHDELETE`, `BACKDELETE`, `MAXNUMMP`.

These values are explained in 4.8, “Node considerations” on page 52.
4.4.2 Installing the TSM client

This step includes installing the TSM backup-archive client, TSM API, and the TSM Tivoli Data Protection product. Each of these pieces can be updated independent of the others. However the TSM backup-archive client and the TDP product each are shipped with their own version of the TSM API. You may encounter situations where upgrading the TSM backup-archive client or upgrading the TSM Tivoli Data Protection product may change the TSM API level.

The backup-archive client is not required in order to do Tivoli Data Protection or DB2 backups. It is needed in order to use TSM scheduling or to backup files and directories that the TDP or DB2 programs are unable to.

Some products like DB2 provide native support for TSM, a TDP product is not necessary. The only thing required to do DB2 backups using TSM is to install and configure the TSM API.

4.4.3 Configuring the TSM client

In order to configure the TSM client, you must perform the following tasks: set the environment variables, create and specify options in the client options file, set up the scheduler, and perform configuration tasks specific to the TDP product being used.

TSM client programs cannot discover TSM servers on a network. The communication method and the address of the TSM server must be specified in a client options file. The client options file is a plain text file, whose default name is dsm.opt. On UNIX systems, the client options file is a combination of two plain text files, dsm.opt and dsm.sys.

The TSM backup-archive client looks for the client options file in a default location or in a location specified by environment variables. The environment variables that are used are DSM_CONFIG, DSM_DIR, and DSM_LOG.

All products that interface with the TSM server through the TSM API use a different set of environment variables. The environment variables that are used are DSMI_CONFIG, DSMI_DIR, and DSMI_LOG. If these environment values are not set, default values are used that are different than the default values that the TSM backup-archive client uses.
4.5 Managing data objects

The TSM server provides storage and retrieval of data objects for TSM clients. When the TSM server stores a data object, it requires the TSM client to provide three types of information:

- Whether to manage the data object as a backup or an archive object
- A description of the data object
- The management class to bind the object to

4.5.1 Archive or data object

The TSM server stores and manages data objects as either backup objects or archive objects. The TSM client determines whether the data object is sent to the TSM server as an archive object or a backup object. With the backup-archive client, you send a file or directory as a backup object by invoking the incremental or selective backup command. To send a data object as an archive object, you invoke the archive command. TDP and API products store data objects as backup or archive objects based on how the programmers wrote the code.

TDP for Oracle stores database and tablespace backups as backup objects on TSM storage.

4.5.2 Description of data object

When a data object is sent, the TSM server places the copy of the client data out on TSM pre-allocated storage (in a storage pool) and records information in its own internal database describing the object. This description includes such things as the node name, filespace name, high level qualifier, low level qualifier, management class. This information is stored in a TSM internal table that can be queried using an SQL `select` statement. For backup objects, you can run the command, `select * from backups`. For archive objects you use the command, `select * from archives`. Both commands are run using the TSM Administrative command line program (`dsmadmc`).

4.5.2.1 Viewing the description of a backup object

Information about each backup object is stored in a TSM internal database table named `backups`. You can use an SQL `select` statement to view the entries in this table. We will restrict the output of the select command by nodename. Before running this command, we did an Oracle RMAN backup of a database using Tivoli Storage Manager.
After taking the backup we ran the SQL query `select * from backups where node_name='JAMAICA_ORACLE'`. The entry in the quotes is case sensitive and must be upper case. The output of this command tells us that the backup object was sent by a client program (TDP Oracle) that authenticated with the TSM server as node JAMAICA_ORACLE. The client program did not specify which management class to use with this object so the TSM server bound the default management class for the node to this backup object. TDP for Oracle for Windows uses `\adsmorc` as the FILESPACE_NAME when sending the object to the TSM server. The HL_NAME or high level qualifier is `\orcnt`. For the LL_NAME or low level qualifier, TSM uses the piece name that is automatically generated by RMAN. The LL_NAME can be controlled by using the `format` option in the RMAN script.

```
RMAN> run{
  2> allocate channel t1 type 'sbt_tape';
  3> backup (database);
  4> }
RMAN-08503: piece handle=0ncm82dr_1_1 comment=API Version 1.1,MMS Version 2.1.10.0
RMAN-08525: backup set complete, elapsed time: 00:02:48
RMAN-08031: released channel: t1
```

4.5.2.2 Viewing the description of an archive object
TDP for Oracle does not store any database objects as archive objects on the TSM server. You could use the TSM backup-archive client archive feature to perform an operating system or file level backup of the Oracle database files. The archive feature does not perform versioning, but keeps all objects under an archive description.
Information about each archive object is stored in a TSM internal database table named **archives**. You can use an SQL `select` statement to view the entries in this table. Here is an example of such a query: `select * from archives where node_name='JAMAICA'`. The entry in the quotes is case sensitive and must be upper case. Prior to running this command, we used the backup-archive client to archive the tnsnames.ora file.

```sql
select * from archives where node_name='JAMAICA' and type='FILE'
AND2963W This SQL query may produce a very large result table, or may require a significant amount of time to compute.
```

Do you wish to proceed? (Yes (Y)/No (N)) y

<table>
<thead>
<tr>
<th>NODE_NAME: JAMAICA</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE: FILE</td>
</tr>
<tr>
<td>HL_NAME: \ORACLE\ORA8\NETWORK\ADMIN</td>
</tr>
<tr>
<td>LL_NAME: TNSNAMES.ORA</td>
</tr>
<tr>
<td>OBJECT_ID: 105875</td>
</tr>
<tr>
<td>ARCHIVE_DATE: 2001-04-05 16:36:04.000000</td>
</tr>
<tr>
<td>DESCRIPTION: Archive Date: 04/05/2001</td>
</tr>
<tr>
<td>CLASS_NAME: DEFAULT</td>
</tr>
</tbody>
</table>

### 4.5.2.3 Differences between backup and archive objects

Based on the output of the SQL commands for backup and archive objects, you can see that there are some similarities and some differences.

The backups table contains two fields that the archives table does not: **STATE** and **DEACTIVATE_DATE**. These two fields are used in implementing versioning which is specific to backup objects. The **STATE** field can be either **ACTIVE_VERSION** or **INACTIVE_VERSION**. When the backup object is inactivated, the **STATE** changes to **INACTIVE_VERSION** and a timestamp when this occurred is placed in the **DEACTIVATE_DATE** field.

The archives table contains one field that the backups table does not: **DESCRIPTION**. The **DESCRIPTION** field is used to group archive objects logically together. For example, the TSM backup-archive client GUI uses this field for the retrieve operation to group archive objects that were backed up with similar descriptions.

### 4.5.2.4 Owner field in backups and archives tables

Both the backups and archives tables have an **OWNER** field for each data object. This is used for client data sent from a UNIX operating system. For files and directories the **OWNER** corresponds to the UNIX security owner for the file or directory. For API data this field normally comes from the UNIX user that is used to perform the backup.
Only the owner of the data object or a root user can access data objects on a TSM server. In our example in the prior section, the OWNER field is blank. This is because the backup was done on a Windows operating system that does not use this field.

Oracle backups and restores source the OWNER value from the user performing the backup or restore. This plays an important role when doing redirected restores.

### 4.5.3 Management class to bind object to

Every data object is bound to a management class when it is first sent to the TSM server. The management class determines how the object will be managed on the TSM server. A management class is one of the structures that is used in TSM policy management. We will not cover all details of TSM policy management in this book. Just what is needed for backing up Oracle databases to TSM.

For a given node the following can be said. Every node belongs to one and only one TSM policy domain. The domain a node belongs to can be viewed with the TSM administrative command `query node <nodename>`. Replace `<nodename>` with the appropriate nodename. The output of that command for our node JAMAICA_ORACLE shows that it belongs to the API_DOMAIN.

```
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Platform Name</th>
<th>Policy Domain</th>
<th>Days Since Last Access</th>
<th>Days Since Locked?</th>
<th>Locked?</th>
</tr>
</thead>
<tbody>
<tr>
<td>JAMAICA_ORACLE</td>
<td>TDP</td>
<td>API_DOMAIN</td>
<td>&lt;1</td>
<td>18</td>
<td>No</td>
</tr>
</tbody>
</table>
```

Each domain contains one and only one ACTIVE policy set. The active policy set has one and only one default management class. The active policy set represents what policy set is currently in use. The default management class represents what management class will be used if the TSM client does not specify a different one or the TSM client specifies a management class that does not exist in the active policy set for the domain that the node belongs to.

The active policy set for a domain and the default management class can be viewed with the command `query policy <domain> active`. Replace `<domain>` with the appropriate domain. The output of that command for our API_DOMAIN shows that the default management class for nodes in the domain API_DOMAIN is API_MGMTCLASS.
The active policy set for a domain can contain more than one management class. Only one can be the default management class. All management classes for the active policy set can be viewed with the command `query mgmtclass <domain> active`. Replace `domain` with the appropriate domain name. The output of that command from our API_DOMAIN shows that there are two management classes in the active policy set for the domain API_DOMAIN. These classes are API_LONGER and API_MGMTCLASS with API_MGMTCLASS being the default management class.

<table>
<thead>
<tr>
<th>Policy Domain Name</th>
<th>Policy Set Name</th>
<th>Default Mgmt Class Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API_DOMAIN</td>
<td>ACTIVE</td>
<td>API_MGMTCLASS</td>
<td>PO for all DB-Backups</td>
</tr>
</tbody>
</table>

Each management class can contain one archive copygroup and one backup copygroup. The copygroup is the final structure in the TSM policy management scheme and contains the most important information. The copygroup is where you specify where the initial destination storage pool of the data object should be. You also specify how the life cycle of the data object is to be managed. The life cycle concept will be covered in the next section. The life cycle is the term used in this book to describe how long a data object (backup or archive) resides on TSM storage before being purged.

The retention settings for a backup copygroup can be viewed with the command `query copygroup <domain> active <mgmtclass>`. Replace `<domain>` and `<mgmtclass>` with the appropriate values for your configuration. The output of that command for our default management class API_MGMTCLASS in the domain API_DOMAIN shows the retention settings for the backup copygroup.

```
tsm: BRAZIL>q policyset api_domain active

<table>
<thead>
<tr>
<th>Policy Domain Name</th>
<th>Policy Set Name</th>
<th>Mgmt Class Name</th>
<th>Default Mgmt Class ?</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API_DOMAIN</td>
<td>ACTIVE</td>
<td>API_LONGER</td>
<td>No</td>
<td>MG for all DB-Backups</td>
</tr>
<tr>
<td>API_DOMAIN</td>
<td>ACTIVE</td>
<td>API_MGMTCLASS</td>
<td>Yes</td>
<td>MG for all DB-Backups</td>
</tr>
</tbody>
</table>
```
In order to see the initial destination storage pool for this copygroup you have to use the `format=detailed` option. Each of the query commands can be used with the additional parameter `format=detailed` to obtain additional information. We only use it in this last command because we did not need the additional information in the prior commands. The same command used previously to view the backup copygroup with the `format=detailed` option shows that the initial destination (COPY DESTINATION) is the API_DISK_B storage pool.

```
   tsm: BRAZIL> q copygroup api_domain active api_mgmtclass
   Policy   Domain      Name: API_DOMAIN
   Policy   Set Name: ACTIVE
   Mgmt Class Name: API_MGMTCLASS
   Copy Group Name: STANDARD
   Copy Group Type: Backup
   Versions Data Exists: 1
   Versions Data Deleted: 0
   Retain Extra Versions: 0
   Retain Only Version: 0
   Copy Mode: Modified
   Copy Serialization: Shared Dynamic
   Copy Frequency: 0
   Copy Destination: API_DISK_B
   Last Update by (administrator): ADMIN
   Last Update Date/Time: 03/21/2001 14:23:48
   Managing profile:
```

In order to see the retention settings for an archive copygroup you add the option `type=archive` to the `query copygroup` command `query copygroup <domain> active <mgmtclass> type=archive`. Replace `<domain>` and `<mgmtclass>` with the appropriate values for your configuration. The output of that command for our default management class API_MGMTCLASS in the domain API_DOMAIN shows the retention settings for the archive copygroup.

```
   tsm: BRAZIL> q copygroup api_domain active api_mgmtclass format=detailed
   Policy Domain Name: API_DOMAIN
   Policy Set Name: ACTIVE
   Mgmt Class Name: API_MGMTCLASS
   Copy Group Name: STANDARD
   Copy Group Type: Backup
   Versions Data Exists: 1
   Versions Data Deleted: 0
   Retain Extra Versions: 0
   Retain Only Version: 0
   Copy Mode: Modified
   Copy Serialization: Shared Dynamic
   Copy Frequency: 0
   Copy Destination: API_DISK_B
   Last Update by (administrator): ADMIN
   Last Update Date/Time: 03/21/2001 14:23:48
   Managing profile:
```
In order to see the initial destination storage pool for this copygroup you have to use the format=detailed option. The same command used previously to view the archive copygroup with the format=detailed option shows that the initial destination (COPY DESTINATION) is the API_DISK_A storage pool.

We now summarize the information presented in this section with regards to data objects:

Every archive data object sent to the TSM server is associated with an archive copygroup that determines the initial destination storage pool and how long the archive data object should reside on TSM storage.

Every backup data object sent by the TSM server is associated with a backup copygroup that determines the initial destination storage pool and how long the backup data object should reside on TSM storage.

For the archive or backup data object, the copygroup comes from the management class to which the object is bound. The management class that is used to bind comes from the active policy set for the domain of the node that sent the data object. The management class used will either be the default management class for the active policy set or one specified by the TSM client.
Once the object is sent to the TSM server, you can verify which management class was used by performing an SQL query of the archive or backup TSM internal database table.

4.5.4 Life cycle of TSM data objects

Backup data objects and archive data objects are managed differently. We will use the term life cycle to describe how these data objects exist on TSM storage from initial creation to when they are purged.

4.5.4.1 Life cycle of archive data objects

An archive object exists in two states, current and expired, before being purged from the TSM server. Figure 9 shows the three steps involved in the life cycle of an archive data object.

Step 1: A copy of the client data is sent to the TSM server as an archive object. Upon initial creation the archive object is in a current state.

Step 2: It remains in a current state until the TSM client program deletes the archive object manually, or the archive object exceeds its retention setting. At this point the archive object changes state from current to expired.

Step 3: The archive object remains in the expired state until expiration processing runs on the TSM server. This process is invoked by a TSM administrator with the expire inventory command. When expiration processing encounters an archive object in the expired state, it purges that object from the TSM database and frees up the storage space where the archive object resided.

![Figure 9. Archive object life cycle](image)

44 Backing Up Oracle using Tivoli Storage Management
When an archive object moves into the expired state, it is no longer accessible by the TSM client. Additionally, there is no way for the archive object to change back to a current state once it has become expired.

4.5.4.2 Life cycle of backup data objects

A backup object exists in three states, active, inactive, and expired, before being purged from the TSM server. Figure 10 shows the four steps involved in the life cycle of a backup data object.

**Step 1:** A copy of the client data is sent to the TSM server as a backup object. When a backup object is sent to the TSM server, it becomes the active version.

**Step 2:** It remains in an active state until the TSM client program deletes the backup object manually, or a newer version of the backup object is sent. At this point the backup object changes state from active to inactive.

**Step 3:** The backup object remains inactive until it exceeds its retention settings. A backup object can exceed retention settings by either time or number of versions. At this point the backup object changes state from inactive to expired.

**Step 4:** The backup object remains in the expired state until expiration processing runs on the TSM server. This process is invoked by a TSM administrator with the `expire inventory` command. When expiration processing encounters a backup object in the expired state, it purges that object from the TSM database and frees up the storage space where the backup object resided.

![TSM Server backup object states diagram](image.png)

*Figure 10. Backup object life cycle*
A backup object that is the active version or in the active state will never be purged from TSM storage that is, it never expires. It must first be inactivated by the TSM client program. The TSM client program can do this by manually deleting the backup object or sending a new version of the backup object.

When a backup object becomes inactive or moves into the inactive state, it is still accessible by the TSM client. A main difference between active and inactive is that an active object becomes inactive due to a client operation. An inactive object becomes expired automatically by the TSM server as soon as it exceeds its retention criteria. Changing from inactive to expired does not require a client operation. There is no way for a backup object to change back to the active state once it has become inactive.

When a backup archive object moves into the expired state, it is no longer accessible by the TSM client. Additionally, there is no way for the backup object to change back to the inactive state once it has become expired.

If the retention for the backup object is set to retain zero inactive objects(retextra=1,verdel=0) or to retain inactive copies for zero days(retextra=0, retonly=0), the active backup object will change to the expired state as soon as the active backup is inactivated.

**What a unique backup object is to TSM**

Both backup and archive objects can be manually deactivated by the client program that initially backed them up. A key difference between backup and archive objects is that a backup object changes states when a newer version of the backup object is sent to the TSM server.

This brings up the question: How does the TSM determine what a unique version is? A backup object is considered “unique” based on NODE_NAME, FILESPACE_NAME, HL_NAME, LL_NAME. These fields and how to view them were discussed in Section 4.5.2.1, “Viewing the description of a backup object” on page 37. When a backup data object is sent to the TSM server, if it has the same NODE_NAME, FILESPACE_NAME, HL_NAME, LL_NAME as an existing backup data object. The new data object becomes the ACTIVE_VERSION, and the older version changes state and becomes an INACTIVE_VERSION.

Many API products use a unique value for the LL_NAME based on a timestamp or a random non-recurring value. Because of this unique value for the LL_NAME, the backup object only changes states from active to inactive when the API product manually inactivates (deletes) the backup object. An active object is not subject to retention settings until it is inactivated. You must run the appropriate command from the API product to inactivate these backups objects, or else they will remain forever on the TSM server.
4.6 TSM server considerations for Oracle backups

Whenever you use the TSM server to backup and restore data objects, it is of the utmost importance to consider which management class the data objects will be bound to. This is true of both API and backup-archive clients. Failure to do so will result in storing the data objects in one of three situations: too long, too short, or just right.

It is highly unlikely that you will manage the objects “just right” if you do not take the time to define your storage requirements, configure the TSM server appropriately, and configure the TSM client to use the correct management classes. If you store the data objects for too long, then you waste space and storage resources on the TSM server. If you store the data objects for too short a time, then you do not have the required files when you need them.

Each of the TDP products and any product that uses the TSM API should have a section in the documentation that describes exactly what retention settings the product uses and how to bind the data objects to the appropriate management class. The TSM server cannot and does not know how long a client program needs to keep the data objects. This must be done by the client.

4.6.1 How TDP for Oracle stores data objects

Database objects stored on the TSM server by TDP for Oracle are stored as backup objects. Each Oracle backup is stored as a unique object by generating a random character string as the low level qualifier (LL_NAME). The RMAN script can control what the LL_NAME or backup piece name is by using the RMAN format command.

If using the format command, you should generate unique backup piece names by either random character string (%U option) or timestamp (%s and %t). This is documented in the Oracle RMAN manual. Failure to do so will cause inconsistency between the RMAN catalog and the TSM server.

Using unique names means that the Oracle backups must be manually inactivated. This is done by allocating a channel for deletion using the same nodename and filespace name that was used to perform the initial backup (see 9.6.6, “Deleting the backup piece from RMAN” on page 139). This also means that the management class to which the backup objects are bound should have retention settings that change the inactivated backup objects to be expired immediately. The retention settings for a backup copy group that would facilitate this is RETONLY=0 and VERDELETED=0.
4.7 Policy management considerations

When deciding how to set up your TSM server to store Oracle backup objects, you need to decide on how you are going to organize your nodes into domains. You must also correctly define and configure your management classes.

4.7.1 Domain considerations

You can use one domain for all your client data or you can specify multiple TSM domains to group nodes with the same backup characteristics together. Using multiple domains helps ensure that the data objects get bound to the appropriate management class. The following are some items to consider if you wish to use multiple domains:

- Backup-archive client, API client
- Platforms separated, such as AIX, Sun, Windows
- Critical data, non-critical data
- Group nodes with the same data characteristics together

Because the policy requirements for Oracle backups are different from the desired settings for regular TSM backup clients, a different management class must be defined within TSM for managing these Oracle backups. There are two ways to implement this different management class setup:

- Define a new management class within an existing policy domain.
- Define a separate policy domain where the default management class contains the required settings.

4.7.1.1 Define a new management class in an existing domain

If you choose to define a new management class within an existing policy domain (which is not the default management class for that domain), then you must add an include statement in the client options file (dsm.opt for Windows, dsm.sys for UNIX) that is used by the Oracle node. This include statement binds the Oracle backup objects to that management class that you have defined for managing these backups.

The include statement would look similar to this:

```
include * ManagementClassName
```

4.7.1.2 Define a separate policy domain

The recommended way is to define a separate policy domain where the default management class has the required settings. Then just register the node that will be used for Oracle backups to this new domain.
If the node is already defined, the Update Node command can be used to move the node to this new domain. This method allows the default management class to be utilized for the Oracle backups. There is no concern over an include statement not being recognized during the backup process.

4.7.2  TSM management class considerations

This section describes how the management class and its associated copygroups should be configured.

4.7.2.1 Backup copy group considerations.

Normally TSM backup copy groups are designed to hold multiple versions of files and directories in order to restore not only the latest (active) but also older versions that had been changed or deleted (inactive).

TDP for Oracle will send backups through the TSM client API directly to the backup copy group of the default management class to which the node is assigned. Oracle assigns unique names to every database backup. The settings that pertain to multiple versions do not apply.

The following retention settings should be used for the management class that will be bound to the Oracle backups.

- \texttt{VEREXISTS=1}
  
  Keeps only one version of the backup file as the name of the backup is unique. (There will not be a newer version of the backup image with the same name).

- \texttt{VERDELETED=0}
  
  If the backup file has been deleted then TSM should not keep an inactive version of this file.

- \texttt{RETEXTRA=0} (the same value as \texttt{RETONLY})
  
  This parameter will never be used as you will never have more than one version of the backup file. To prevent confusion set this parameter to the same value as \texttt{RETONLY}.

- \texttt{RETONLY=0}
  
  When a backup image file becomes inactive it will be purged from the TSM server at the next expiration.

4.7.2.2 Archive copy group considerations

Archive copy groups are designed to hold archive objects for a dedicated time and than to automatically delete the object.
TDP for Oracle does not send any archive objects to the TSM server, so you do not need to worry about this. You may wish to have an archive copy group to prevent warning messages on the TSM server.

The archive feature of the TSM backup-archive client can be used to perform operating system / file level backups of the Oracle files. With archives you can group files together to expire at the same time. If you choose to do this, pay attention to the archive copygroup that you are using. The archive copygroup will determine for how long the objects will exist before expiring.

### 4.7.3 TSM client include-exclude option

The include-exclude list is normally configured for the TSM backup-archive client. It provides the TSM backup-archive client with the default information which files to include in a backup and which to exclude. Another issue of this option is to define to which management class each file should be bound.

Each Oracle backup file will be bound, if not otherwise specified, to the default management class to which the TDP for Oracle node is assigned at the TSM server. To send a backup to another management class, the include-exclude option can be used.

Before we start, we need to know how the TSM server stores the Oracle backups. Following are the commands to get the required information.

Login via the administrative command line interface to the TSM server. Use `dsmadmc` with your userid and password. Then enter the following command on the `tsm:` prompt. Be careful to write the TSM node name in uppercase.

```
tsm: BRAZIL>select * from backups where node_name='JAMAICA_ORACLE'
ANR2963W This SQL query may produce a very large result table, or may require a significant amount of time to compute.
Do you wish to proceed? (Yes (Y)/No (N)) y

NODE_NAME: JAMAICA_ORACLE
FILESPACE_NAME: \adsmorc
STATE: ACTIVE_VERSION
TYPE: FILE
HL_NAME: \orcnt\LL_NAME: 0ncm82dr_1_1
OBJECT_ID: 94130
BACKUP_DATE: 2001-04-02 08:36:30.000000
DEACTIVATE_DATE:
OWNER:
CLASS_NAME: DEFAULT
```
The name of the file consists of three parts. The filespace name, the high level name and the low level name. To decide which file should go into which management class any combination can be specified in the include-exclude option.

For Windows-based systems, the INCLUDE statement must be specified in the dsm.opt file. For UNIX-based systems, the INCLUDE statement must be specified in the dsm.sys file. Additionally, there is an INCLEXCL option for UNIX-based systems to define a file that contains all include-exclude statements for a specific server.

The following is a sample include-exclude list file for a UNIX system. This file will be compared, from bottom to top, every time a file is being backed up to the TSM server. The file will be sent to the management class listed on the line where the backup object first matches during the comparison:

```
INCLUDED * API_LANFREE
INCLUDED *LOGS* API_MGMTCLASS
```

In our example, any backup object that contains the string LOGS will be bound to the API_MGMTCLASS. All other backups will go to the API_LANFREE management class.

If a backup object does not match an include or exclude statement, it will go to the DEFAULT management class of the TSM node.

For UNIX, the include-exclude list itself can have any name, and it can reside anywhere on the system. As there can be more clients on the system using include-exclude list files, it is a good idea to use the default directory /usr/tivoli/tsm/client/api/bin to contain all the inclexcl.list files. This file should be owned and controlled by the TSM server administrator.
4.8 Node considerations

A TSM client program must authenticate to the TSM server using a previously defined node. A TSM client can be a backup-archive client or an API client. There may be more than one TSM client on a single machine.

We strongly recommend registering a dedicated node for the Oracle backups that is separate from the node that is used with the backup-archive client. If you have a separate policy domain for Oracle backups, this is required, since a node can only belong to one domain.

When registering a node, be sure to specify the following parameters:

- NODENAME
- PASSWORD
- BACKDELETE
- MAXNUMMP
- DOMAIN

4.8.1 Choosing a nodename

The TSM backup-archive client defaults the nodename to be the hostname of the machine. We chose to register separate nodes for the Oracle backups, and we used the naming convention of hostname_oracle.

4.8.2 Choosing a password

This password will be used with the aobpswd command to encrypt the password for future use when PASSWORDACCESS is set to GENERATE with TDP for Oracle for Windows. With TDP for Oracle for UNIX, PASSWORDACCESS is set to PROMPT; however, the aobpswd command generates an encrypted password in the TDP directory, which accomplishes the same purpose.

4.8.3 Setting the BACKDELETE option

The node that is used to backup the database objects to TSM must be able to manually delete the backup objects. To do so specify BACKDELETE=YES when registering the node. You can update a node that has this value set to NO and set it to YES with the update node command.

If this value is set to NO, RMAN will not be able to manually inactivate the Oracle backup objects, and they will reside in TSM storage forever.
4.8.4 Specifying the domain

The node that is used for backing up Oracle to TSM must belong to a domain that contains the required management class. The domain that a node belongs to can be specified when the node is initially registered, or the node can be updated to belong to a new domain. To do this, specify DOMAIN=domain_name when registering or updating the node.

4.8.5 Setting the MAXNUMMP

A node is restricted to the number of tape mounts specified with this option. In order for the TSM client to use multiple sessions when sending backups to tape, the maximum number of mount points parameter MAXNUMMP must be equal to or greater than the number of sessions involved in the backup. The number of session used during a backup corresponds to the number of channels allocated in an RMAN script.

The MAXNUMMP must not be set to a value greater than the actual number of physical drives defined to the TSM server. If TDP for Oracle is unable to acquire enough tape mounts, either because this value is set lower than the number of sessions, or because the tape mounts are not available, the backups may fail, or have to wait for one session to finish. MAXNUMMP=2 (to allow the client to mount a maximum of 2 drives).

4.9 Storage pool considerations

Storage pools (STG) represent the actual physical devices that will hold your data. Each copy group (backup or archive) will send its data to the associated storage pool (configuration parameter: destination). It is possible to have only one storage pool (for example one tape storage pool) to hold all the data from all the TSM clients. We do not recommend to do this.

In the remainder of this section, we provide some suggestions to consider when designing your storage pools:

- Specifying MAXSIZE (disk STG parameter)

  For a storage pool, you can specify the maximum size object that it will accept. If a data object is larger than the maxsize, it will try to go to the next storage pool (typically tape). If the data object is larger than the maxsize of all storage pools in the hierarchy, then the TSM server will reject that object.

  You can utilize this option to send really large database objects to tape instead of to your disk pool. If you do not have enough disk space to hold an entire days worth of backups, and you use migration to free up space,
you can save time by sending the biggest objects straight to tape. This way all of your client sessions can be sending data to either disk or tape, and you do not have client sessions waiting on migration to finish. The other reason you save time is because the end result of migration is that the objects end up on tape. If you send it straight to tape, you do not spend the time sending it to disk, then migrating it to tape.

We updated our storage pool so that the maxfile size was 50% of the total size of the disk storage pool:

```
update stgpool api_disk maxsize=2g
```

- **Using dedicated storage pools**
  Dedicated in this context means that the storage pool is reserved for one specific client. No other client sends files to this storage pool. There is no configuration parameter to implement this. It is a matter of how the policy setup had been done. There are different reasons why you may wish to use dedicated storage pools. Certain TSM server processes operate based on the storage pool, and you may want have the server processes affect only certain clients. The workaround for this is to send these clients to dedicated storage pools, you can then run the TSM server processes on these dedicated storage pools separate from your standard storage pools. Among these processes, the use of copy storage pools, collocation, and reclamation may warrant the use of dedicated storage pools.

- **Copying storage pools**
  To safeguard against defective tapes, copy storage pools can be used. If you have dedicated storage pools you can be more flexible in which data should be copied to a copy storage pool and which not. As backups and logfiles are often very critical to a company these data are a good candidate to be copied. It may be useful to define a dedicated storage pool for the backup copy group and the archive copy group and only setup the archive copy group (containing the logfiles) to be copied to a copy storage pool.

- **Reclamation**
  It should be the responsibility of the Oracle administrator to delete old backups. If the Oracle database administrator decides to delete all database backups after they reached a well defined age, all these files will expire naturally. If these files reside in a dedicated tape pool reserved for the Oracle client, the reclaim value of this tape pool can be set to a very high value. There is no need of doing much reclamation if the files will be deleted anyway.
4.10 Our TSM server setup

In this section we show the specific setup that we used for our existing TSM server in order to be ready to receive the Oracle backup files.

Our existing environment already contained the following components:

- TSM server 4.1.2
- TSM client 4.1.1
- A tape driver 5.4.4.0
- Ultrium Tape Library 3583 with two 3580 tape drives.
- 10 formatted disk volumes named vol1 to vol10 each 2GB in size.

In the first step, we create the domain the policy and the management class:

```bash
  tsm: BRAZIL> def domain api_domain DESC='DO for all DB-Backups'  
  ANR1500I Policy domain API_DOMAIN defined.
  tsm: BRAZIL>
  tsm: BRAZIL> def policyset api_domain api_policy DESC='PO for all DB Backups'  
  ANR1510I Policy set API_POLICY defined in policy domain API_DOMAIN.
  tsm: BRAZIL> def mgmtclass api_domain api_policy api_mgmtclass DESC='MC for all DB-Backups'  
  ANR1520I Management class API_MGMTCLASS defined in policy domain API_DOMAIN,  
  set API_POLICY.
  tsm: BRAZIL>
```

Next, the tape and disk storage pools are being created, and some pre-formatted disk volumes are assigned to the disk pool:

```bash
  tsm: BRAZIL> def stgpool api_3580_a 3580class DESC='Archive tape pool' rec=95  
  maxscr=10 maxsize=10g  
  ANR2200I Storage pool API_3580_A defined (device class 3580CLASS).
  tsm: BRAZIL> def stgpool api_3580_b 3580class DESC='Backup tape pool' rec=95  
  maxscr=10 maxsize=10g  
  ANR2200I Storage pool API_3580_B defined (device class 3580CLASS).
  tsm: BRAZIL>
```
After the storage pools are created, the copygroups can be configured, and these storage pools can be assigned as their destination storage pools:

```
> def stg api_disk_a DISK DESC='Archive disk pool' NEXT=api_3580_a hl=70 lo=20
ANR2200I Storage pool API_DISK_A defined (device class DISK).
> def stg api_disk_b DISK DESC='Backup disk pool' NEXT=api_3580_a hl=70 lo=20
ANR2200I Storage pool API_DISK_B defined (device class DISK).
> def vol api_disk_a /tsm/dskpool/vol1
ANR2206I Volume /tsm/dskpool/vol1 defined in storage pool API_DISK_A (device class DISK).
> def vol api_disk_b /tsm/dskpool/vol2
ANR2206I Volume /tsm/dskpool/vol2 defined in storage pool API_DISK_B (device class DISK).
```

To actually use this configuration, the policy set needed to be validated and activated:

```
> validate policyset api_domain api_policy
ANR1515I Policy set API_POLICY validated in domain API_DOMAIN (ready for activation).
> activate policyset api_domain api_policy
Do you wish to proceed? (Yes/No) y
ANR1514I Policy set API_POLICY activated in policy domain API_DOMAIN.
```
Now, the configuration is usable, and we can define the client nodes to work with this setup.

```
> register node jamaica_oracle jamaica_oracle domain=api_domain
    backdelete=yes maxnummp=2
ANR2060I Node JAMAICA_ORACLE registered in policy domain API_DOMAIN.
ANR2099I Administrative userid JAMAICA_ORACLE2 defined for OWNER access to node JAMAICA_ORACLE.
```

This completes the server setup.
Part 2. TDP for Oracle on the UNIX platforms
Chapter 5. Configuration and setup of TDP for Oracle on UNIX

This chapter describes how to install and setup Tivoli Data Protection for Oracle on a UNIX platform like AIX or Sun Solaris.

The environment we used for the examples described in this book is shown in Figure 11.

Figure 11. AIX and Sun Solaris over ethernet

5.1 TDP for Oracle

The following sections cover the installation details of version 2.1.10 of the Tivoli Data Protection for Oracle including how to install, setup environment variables and edit the dsm.opt file.

5.1.1 Install TDP for Oracle

To install TDP for Oracle, use `smit` on the AIX system or `pkgadd` on the Sun Solaris system.

For further informations on requirements, check the *Tivoli Data Protection for Oracle for UNIX Installation and User’s Guide*, SH26-4112-01.

TDP for Oracle requires the TSM API version 3.7.2 or later. On the Solaris system, this package must be installed prior to TDP for Oracle. The TSM API code is shipped together with TDP for Oracle.
If you are planning to use the 64-bit TDP for Oracle, you must use TCP/IP as the communication protocol.

There are two packages for different UNIX platforms. The default installation directories are shown in Table 1.

<table>
<thead>
<tr>
<th>Package</th>
<th>Operatin system</th>
<th>Package name</th>
<th>Default installation directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDP for Oracle 32-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.32bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TDP for Oracle 64-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.64bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TSM API 32-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.api.32bit</td>
<td>/usr/tivoli/tsm/client/api/bin</td>
</tr>
<tr>
<td>TSM API 64-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.api.64bit</td>
<td>/usr/tivoli/tsm/client/api/bin64</td>
</tr>
<tr>
<td>TDP for Oracle 32-bit</td>
<td>Solaris</td>
<td>TivsmCapi</td>
<td>/opt/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TDP for Oracle 64-bit</td>
<td>Solaris</td>
<td>TivsmCapi</td>
<td>/opt/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TSM API 32-bit</td>
<td>Solaris</td>
<td>TDPoracle, TDPorclic</td>
<td>/opt/tivoli/tsm/client/api/bin</td>
</tr>
<tr>
<td>TSM API 64-bit</td>
<td>Solaris</td>
<td>TDPoracle, TDPorclic</td>
<td>/opt/tivoli/tsm/client/api/bin64</td>
</tr>
</tbody>
</table>

For further information on requirements, refer to these sources:

- The Web site for online manuals:
- The file /opt/tivoli/tsm/client/oracle/README.AOB

### 5.1.2 Define environment variables

The TSM API uses unique environment variables to locate files. This allows you to use different files for API applications to those that are used by the TSM backup client, if necessary.

For Oracle version 7 and the Enterprise Backup Utility, the user initiating the backups and restores had to define environment variables. Oracle Version 8 and RMAN work differently, because RMAN is started by a database process now. The environment variables for TDP for Oracle are passed by the RMAN allocate channel command.

For the TDP for Oracle password file generation, described in 5.1.5, “Initialize the password” on page 66, environment variables are still used.
Table 2 lists the most important environment variables.

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSMO_NODE</td>
<td>This is used to specify a unique node name for TDP for Oracle. The value can contain a string of 1 to 64 characters. The default value is the value returned by the UNIX hostname command. TDP for Oracle does not use the node name in the dsm.sys file.</td>
</tr>
<tr>
<td>DSMO_AVG_SIZE</td>
<td>This is the average size of an object in MB. This information is passed to the TSM server, where it is used to determine storage pool usage and device usage. The default is 50 MB. Important: It is recommended that you set the variable to a value larger than your estimated object size. This enables the TSM server to use the correct storage pools (disk or tape) from the beginning of the backup.</td>
</tr>
<tr>
<td>DSMI_ORC_CONFIG</td>
<td>This points to the user option file dsm.opt. If the variable is not set, TDP for Oracle tries to source dsm.opt from the default TSM API installation directory.</td>
</tr>
<tr>
<td>DSMI_LOG</td>
<td>This points to a directory that contains the TSM API error log file dsierror.log. Make sure that the user who is performing backups has the necessary authorization to add or modify the file here.</td>
</tr>
<tr>
<td>DSMO_FS</td>
<td>This contains the filespace name used on the TSM server. You can use different filespace names to make use of different management classes. The default filespace name is /adsmorc.</td>
</tr>
<tr>
<td>DSMO_OWNER</td>
<td>This option specifies a session owner name and object owner name. The value can contain a string of 1 to 64 characters. The default value is the value returned by the UNIX id command.</td>
</tr>
<tr>
<td>DSMO_DEBUG=49</td>
<td>When set, TDP for Oracle will create a file called orcagent.log in $ORACLE_HOME/dbs containing useful trace information.</td>
</tr>
</tbody>
</table>
5.1.3 Edit the client options files

The TDP for Oracle uses client option files, dsm.sys and dsm.opt, similar to those of the backup-archive client.

These files define how the TDP for Oracle client can communicate with a TSM server, and specify which options are in use.

Because we used a new nodename for TDP for Oracle, we had to create a user option file, dsm.opt, which points to the correct servername stanza in the system option file, dsm.sys.

The file dsm.sys has to include the statement PASSWORDACCESS PROMPT. Figure 12 explains that correlation between the environment variables defined in the RMAN allocate channel command, the user option file, and the system option file.
RMAN allocate channel command including TDP for Oracle environment variables

```sql
allocate channel ch type "sub_tape" parms
  "ENV=(DSM_NODE=sol_oracle,
   DSM_AVG_SIZE=256,
   DSM_ADDRESS=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
   DSM_LOG=/usr/local/oracle/log);$`
```

Figure 12. Relationship environment variable — file dsm.opt — file dsm.sys

```text
SERVERNAME     sol_oracle
traceflag      config
tracefile      /usr/local/oracle/log/tdmporacletrace
```

```text
SERVERNAME     sol_oracle
COMM_METHOD    tcpip
TCPSERVERADDR  193.1.1.11
NODENAME       sol-e
PASSWORDACCESS  generate
```
5.1.4 Registering with the TSM server

For our database server, we decided to use different TSM node names. Each node name is bound by its domain name and policy name to an appropriate management class.

The nodenames we are using in our dsm.sys file are listed in Table 3.

<table>
<thead>
<tr>
<th>Nodename</th>
<th>Server name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>sol_e</td>
<td>brazil_ba</td>
<td>Generic backup/archive clients</td>
</tr>
<tr>
<td>sol_oracle</td>
<td>sol_oracle</td>
<td>TDP for Oracle</td>
</tr>
<tr>
<td>sol_db2</td>
<td>brazil-db2</td>
<td>DB2 backups and log archives</td>
</tr>
</tbody>
</table>

See 4.8, “Node considerations” on page 52, for more details about the setup used.

5.1.5 Initialize the password

TDP for Oracle ships a program, aobpswd, to simulate the TSM function PASSWORDACCESS=GENERATE. This eliminates the need to enter a password manually, allowing RMAN to establish a session with the TSM Server without prompting each time for a password. The aobpswd program encodes the TSM password and stores it in a secure file located with the DSMO_PSWDPATH environment variable. The password is then passed to the TSM API to the TSM server without user intervention.

To initialize the TDP for Oracle password, perform the following steps as root user:

1. Export DSMO_NODE=sol_oracle.
2. Export DSMI_CONFIG=/opt/tivoli/tsm/api/bin/dsm.opt.
3. Export DSMI_LOG=$PWD.
4. Set PASSWORDACCESS PROMPT in the options file (dsm.sys).
5. Start the aobpswd program.
   a. Enter the current node password.
   b. You are prompted for a new password. If you just press the Enter key, the password will not be changed.

Successful execution of the aobpswd program generates a password file named TDPO.<your node name> in your default installation directory.
Chapter 5. Configuration and setup of TDP for Oracle on UNIX

5.1.6 Setting up RMAN

Because we are using Oracle Version 8.1.7, we had to remove a symbolic link:

- $ORACLE_HOME/lib/libbk.so (for Sun Solaris)
- $ORACLE_HOME/lib/libbk.a (for AIX)

Then we had to symbolically link it with:

- /usr/lib/libbk.so (for Sun Solaris)
- /usr/lib/libbk.a (for AIX):
Chapter 6. Oracle backup considerations on UNIX

RMAN is the backup and recovery tool provided by Oracle for Oracle8 Server databases and is packaged with the Oracle8 Server product. RMAN manages the process of backing up, restoring, and recovering Oracle8 target database instances. RMAN can automatically back up, restore, and recover the following database objects:

- Database
- Tablespaces
- Data files
- Control files
- Archived redo logs

In addition to backup and restore operations, RMAN can:

- Generate log records of all backup and recovery operations
- Run backup and restore operations in parallel to improve performance
- Find database objects that require a backup, on the basis of user-defined values

RMAN provides the interface to the Oracle8 database and the functions for backup, restore, and recovery. It does not provide any storage management capabilities and must be integrated with other storage management products such as TSM to provide a complete enterprise wide storage management solution.

6.1 RMAN: Oracle's Recovery Manager

This section presents an overview of the architecture and the components involved in backing up Oracle8 databases with RMAN and TDP for Oracle. It is an introduction to RMAN and in particular how it integrates with TSM. It is not an authoritative guide to RMAN or Oracle8 backup and recovery. This chapter should be read in conjunction with the Oracle8 Server Backup and Recovery Guide.

6.1.1 RMAN system components

RMAN consists of several components that interact during the backup and recovery process.

Note: The system components involved with RMAN, and the flow of an RMAN operation to TSM, are illustrated in Figure 13 on page 72.
6.1.1.1 RMAN command
The RMAN command is the database administrator’s interface to RMAN. It invokes a Command Line Interface that provides a scripting language (operating system independent) for performing backup and recovery operations. RMAN can be executed either interactively, where a command prompt is displayed and additional RMAN commands entered, or in batch mode, where an RMAN command file is executed.

6.1.1.2 Target database
The target database is the Oracle8 database instance on which RMAN executes specified backup, restore, and recovery actions. When the RMAN command is executed, it connects to the target database. The target database is specified by using RMAN parameters.

6.1.1.3 Communication channel
RMAN can perform backup and restore functions to either local disk or to external media management products such as an TSM server through the libobk.a library provided by TDP for Oracle for UNIX. These I/O operations are performed over a communication channel that defines the device to be used for the operation. The channel is used by RMAN to send or receive backup data to and from the I/O device.

For backup and restore operations, you must allocate a channel before the operation is performed. A channel corresponds to a single device. With the TDP for Oracle for UNIX, a channel is a single session to an TSM server using the SBT API. Multiple channels can be allocated. RMAN provides a multiplexing feature that enables parallel data streams to be sent over multiple allocated channels to maximize backup and recovery performance.

6.1.1.4 Recovery catalog
The recovery catalog is the repository for information about backup objects created by RMAN. It is a Oracle8 database instance, separate from the target databases, and can contain information for multiple target databases. The data stored in the recovery catalog comprises structural information about the target databases to back up and restore. The recovery catalog contains information about:

- Physical schema of a target database

You must register the target database at the recovery catalog to define the physical schema of the target database. RMAN needs to know about any structural change of the target database, and obtains this information from the target database control file.
• Database backup history

RMAN backs up databases, tablespaces, data files, control files, and archive logs to the TSM server. Details of these backup objects held on TSM is stored in the recovery catalog.

• Backup and recovery history

RMAN stores backup, restore, and recovery information to maintain a history of previously performed operations. When backup and restore operations are performed, this information enables RMAN to determine:

- Database files that require backing up
- Old backup files that can be deleted
- Files that are not recoverable

• Stored RMAN scripts

RMAN commands can be stored in the recovery catalog as stored scripts. Scripts can be created to automate the execution of a several RMAN operations.

---

**Note**

Oracle strongly recommends that a recovery catalog be used with RMAN. However, it is possible to execute RMAN commands without one. If you operate without a recovery catalog, RMAN uses the target database control file to store backup and structural information about the database. The following limitations apply when operating without recovery catalog:

- Point-in-time recovery is not possible.
- Stored RMAN scripts cannot be used.
- Recovery cannot be performed if the control files are lost or damaged.

Oracle recommends the use of multiplexed control files, with each file located on different disks to protect against media failure.

---

Refer to Figure 13 for the following discussion.
As shown in Figure 13, the operation is started by invoking the command `rman` and entering the appropriate commands directly or by submitting a command file containing the commands (1).

RMAN connects to the recovery catalog (2) and the target database (3).

Before any backup or restore operations can be performed, RMAN allocates a channel to TSM, using the SBT API (4).

RMAN then creates a server process on the target database instance that performs the operation (5).

For restore operations, RMAN queries the recovery catalog to determine which files to restore from TSM. For a backup operation, RMAN backs up the objects specified in the command to TSM. In both cases, the data is transferred on the previously defined channel.
6.2 Creating a database for the RMAN catalog

When RMAN is used to back up several databases using the same RMAN catalog, you may wish to create a separate recovery catalog database and create the RMAN user in that database.

In our test environment, we used an existing database on a Sun Solaris system as a recovery catalog for databases residing on an AIX system and on a Windows system.

The recovery database itself was backed up with RMAN using a recovery catalog in another database.

**Note**

Oracle recommends that the recovery catalog database is operated in ARCHIVELOG mode and located on another server than the production database(s).

Use the Export utility to make additional backups of your recovery catalog data. An export of the catalog allows the most flexibility when the recovery catalog must be restored, because it can be restored to any existing Oracle8 database.

6.2.1 Creating the RMAN catalog

The procedure we used to create the recovery catalog is described in the manual *Oracle8i Recovery Manager User's Guide and Reference*, A76990-01.
The following are sample print screens showing the creation of the user `rman` and the `create catalog` command:

```
oracle@sol-0 -> sqlplus sys/manager
SQL*Plus: Release 8.1.7.0.0 - Production on Wed Mar 7 11:24:12 2001
(C) Copyright 2000 Oracle Corporation. All rights reserved.

Connected to:
Oracle8i Enterprise Edition Release 8.1.7.0.0 - Production
JServer Release 8.1.7.0.0 - Production
SQL> create user rman identified by rman temporary tablespace TEMP default tablespace USERS quota unlimited on USERS;
User created.
SQL> grant connect, resource to rman;
Grant succeeded.
SQL> grant recovery_catalog_owner to rman;
Grant succeeded.
SQL> 
```

```
oracle@sol-0 -> rman catalog rman/rman@RIO
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-06008: connected to recovery catalog database
RMAN-06428: recovery catalog is not installed
RMAN> create catalog tablespace users;
RMAN-06431: recovery catalog created
```
6.2.2 Registering the target database

Having created a recovery catalog, you must register the target databases to it, using RMAN commands. Registration must be done before any other RMAN functions can be performed on the target databases.

You register each of the target database instances that will use the recovery catalog. As the Oracle user, run the `rman` command to connect to the target and recovery catalog database, and then run the `register database` command.

The next example shows the commands to connect to the PARIS target database as the internal user and the RIO recovery catalog database as the newly created `rman` user, followed by the target database being registered:

```
oracle@sql-e -> rman target 'internal/manager@PARIS' rcvcat 'rman/rmanRIO' log="register_parts.log" ondiskfile="register_database.rman"
RMAN> oracle@sql-e ->
oracle@sql-e -> more register_parts.log
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-08005: connected to target database: PARIS (DBID=3920710371)
RMAN-08006: connected to recovery catalog database
RMAN> register database:
RMAN-08002: compiling command: register
RMAN-08002: executing command: register
RMAN-08006: database registered in recovery catalog
RMAN-08002: starting full resync of recovery catalog
RMAN-08004: full resync complete
Recovery Manager complete.
oracle@sql-e ->
```

After the target database is registered in the recovery catalog, RMAN re-synchronizes the metadata between the current control file and the recovery catalog.
To list the result of the register command, use the RMAN `report schema` command:

```
oracle@sol-e -> rman target "internal/manager@PARIS" rcat "rman/rman@RIO" log="report_schema_paris.log" condfile="report_schema.rman"
RMAN> 2> oracle@sol-e -> oracle@sol-e -> more report_schema_paris.log
```

Recovery Manager: Release 8.1.7.0.0 - Production

RMAN-06015: connected to target database: PARIS (DBID=3820710371)
RMAN-06018: connected to recovery catalog database

RMAN> report schema;
2>
RMAN-03022: compiling command: report

Report of database scheme

<table>
<thead>
<tr>
<th>File</th>
<th>K-bytes</th>
<th>Tablespace</th>
<th>RB</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>153600</td>
<td>SYSTEM</td>
<td>YES</td>
<td>/opt/acp/oracle/oradata/paris/system01.dbf</td>
</tr>
<tr>
<td>2</td>
<td>8192</td>
<td>TOOLS</td>
<td>NO</td>
<td>/opt/acp/oracle/oradata/paris/tools01.dbf</td>
</tr>
<tr>
<td>3</td>
<td>262144</td>
<td>RBS</td>
<td>YES</td>
<td>/opt/acp/oracle/oradata/paris/rbs01.dbf</td>
</tr>
<tr>
<td>4</td>
<td>51200</td>
<td>TEMP</td>
<td>NO</td>
<td>/opt/acp/oracle/oradata/paris/temp01.dbf</td>
</tr>
<tr>
<td>5</td>
<td>204800</td>
<td>USERS</td>
<td>NO</td>
<td>/opt/acp/oracle/oradata/paris/users01.dbf</td>
</tr>
<tr>
<td>6</td>
<td>204800</td>
<td>INNR</td>
<td>NO</td>
<td>/opt/acp/oracle/oradata/paris/ind-01.dbf</td>
</tr>
</tbody>
</table>

Recovery Manager complete.
```

```
oracle@sol-e =>
```

### 6.3 Back up the database using TDP for Oracle and RMAN

When you execute the `backup` command, you create one or more backup sets. A `backup set`, which is a logical construction, contains one or more physical `backup pieces`. Backup pieces are operating system files that contain the backed up datafiles, control files, or archived redo logs. You cannot split a file across different backup sets or mix archived redo logs and datafiles into a single backup set.

A backup set is a complete set of backup pieces that constitute a full or incremental backup of the objects specified in the `backup` command. Backup sets are in an RMAN-specific format; image copies, in contrast, are available for use without additional processing.

Each backup piece contains control and checksum information that allows the Oracle server process to validate the backup piece during a restore. A backup set is created by the `backup` command. A `restore` command is required to extract files from a backup set.
See Figure 14 for an illustration for backup sets and backup pieces.

![Diagram of backup sets and pieces]

Figure 14. backup sets and backup pieces created by RMAN

### 6.3.1 Full backup

A full backup is a non-incremental backup of one or more data files. A full backup has no effect on incremental backups and is not considered to be part of the incremental strategy.

If the database is in ARCHIVELOG mode, you can choose to do full backup while the database is online or offline. If the database is in NOARCHIVELOG mode, the database must be closed by a clean shutdown. Full backups can be taken of:

- Data files
- Tablespaces
- Databases
- Control files
- Archive logs
6.3.2 Whole database backup

A whole database backup set contains the control files and all database files that belong to that database. Whole database backups do not require the database to be operated in a specific archiving mode. They can be taken whether a database is operating in ARCHIVELOG or NOARCHIVELOG mode. If the database is in ARCHIVELOG mode, you can choose to back up the database while it is open or closed. If running in NOARCHIVELOG mode, the database must be shut down first. There are two types of whole database backups:

- Consistent whole database backup
  
  A consistent whole database backup is a backup set where all files within it are consistent to the same point in time. A consistent whole database is the only valid backup for databases running in NOARCHIVELOG mode. The only way to take a consistent whole database backup is to shut down the database cleanly and take a backup while the database is offline.

- Inconsistent whole database backup
  
  An inconsistent whole database backup is a backup of an online database. It is inconsistent because portions of the databases may have been modified and written to disk during the backup process. The database must be in ARCHIVELOG mode in order to run an inconsistent backup.

Note

After an inconsistent backup is performed, the archived and online redo logs should also be backed up. Inconsistent whole database backups are restored and made consistent by applying any subsequent incremental backups and redo logs, online and archive, during the recovery process.

6.3.3 Incremental backup

RMAN provides the capability of incrementally backing up databases at the individual block level. An incremental backup is a backup of one or more data files and contains only those blocks that have been modified since a previous backup at the same or lower level.

The multilevel incremental backup feature allows you to create different levels of incremental backups. Each level is denoted by an integer, with 0 being the lowest backup level. An incremental backup performed at a given level backs up only those blocks that have been modified since the last backup at the same or lower level.
Chapter 6. Oracle backup considerations on UNIX

An incremental backup can be performed on:

- Individual data files
- Tablespaces
- The entire database

Incremental backup of control files or archived logs is not supported. There are two types of incremental backups: non-cumulative and cumulative.

### 6.3.3.1 Non-cumulative incremental backup

A non-cumulative incremental backup backs up only those blocks that have changed since the previous incremental backup at the same or lower level. This is the default mode of operation for incremental backups.

A level 0 backup backs up all blocks that contain data. It performs the same backup as a full backup. A level 0 backup is required for subsequent incremental backups at other levels. An incremental backup at a level greater than level 0 backs up only those blocks that have changed since a previous incremental backup at the same or lower level. The size of the backup depends on the number of blocks modified.

Figure 15 illustrates part of a monthly cycle of non-cumulative incremental backups. The cycle is based on backup levels 0, 1, and 2. A weekly backup is performed at level 0 on Sunday, incremental backups level 2 are performed on Monday to Wednesday and on Friday and Saturday, and weekly incremental backups at level 1 on each Thursday.

![Figure 15. Non-cumulative incremental backups](image)
6.3.3.2 Cumulative incremental backup
A cumulative incremental backup at level $n$ contains only blocks that have been changed since the most recent backup at level $n - 1$ or lower. Cumulative backups require more storage space than differential backups, but they are preferable during a restore operation because only one backup for a given level is needed. See Figure 16.

Note that the first incremental backup must be a level 0 backup that contains all used blocks. A cumulative backup at level 2 will contain all blocks changed since the most recent level 1 backup, copying all blocks changed since the base level 0 backup only if a previous level 1 is unavailable. In contrast to a cumulative backup, a differential backup at level 2 will determine which level 1 or level 2 backup occurred most recently and copies all blocks changed since that backup.

![Figure 16. Cumulative incremental backups](image)

6.3.4 Image copies
An image copy is a single file (data file, archive log, or control file) that can be used as-is to perform a recovery. It is similar to an operating system copy of a single file, except that it is produced by an Oracle server process which performs additional tasks such as validating the blocks in the file and registering the copy in the control files. An image copy can be done only to disk.

Image copies are not discussed further in this chapter, because RMAN does not send image copies to TSM — they are always stored locally on disk.
6.4 Backup examples

This section describes some typical backup examples using RMAN, complete with sample scripts. The examples are based on the following database instances:

- Target database: paris
- Recovery catalog database: rio

For the examples, we used RMAN command files and redirected the output to logfiles. RMAN command files are regular text files containing commands executed by RMAN.

The examples illustrate how to perform the following functions:

- Consistent whole database backup
- Archive redo log backup
- Incremental level 0 backup

6.4.1 Consistent whole database backup

In the following examples, we are using RMAN to shut down and restart the database, to guarantee that all read-write datafiles and control files have been check-pointed with respect to the same system change number (SCN).

These examples are suitable for databases operating in ARCHIVELOG and NOARCHIVELOG mode, and in our opinion, they represent the most common backup operation:

```
rman target internal/manager@PARIS rcvcat rman/rman@RIO
cmdfile=backup_offline_paris log=backup_offline_paris.log
```
To guarantee that a database's datafiles are consistent, shut down the database with the NORMAL, IMMEDIATE or TRANSACTIONAL options before making a whole database backup. To bring up the database in the desired state, we chose the sequence:

- shutdown immediate;
- startup force dba;
- shutdown immediate;
- startup mount;

For TDP for Oracle environment variables in the allocate channel command:

- DSMO_NODE is set because we use a different node name to the TSM backup client.
- DSMO_AVG_SIZE is a size estimate used by the TSM server to reserve space on TSM storage.
- DSMI_ORC_CONFIG points to our dsm.opt file.
- DSMI_LOG points to our backup log directory - the default is the $ORACLE_HOME/dbs directory of the Oracle instance PARIS.
- The allocation of multiple channels is only possible when TSM node’s MAXNUMMP (maximum mount points allowed) is greater than or equal to the number of channels.
The following is an explanation of various statements in our coding example:

- set maxcorrupt for datafile 1 to 0 causes Oracle to abort the backup process if RMAN detects corrupt blocks in the SYSTEM tablespace. In our database, we used AUTOEXTEND=ON for all datafiles. Datafile 1 is always used for the SYSTEM tablespace.
- backup full causes a whole database backup.
- filesperset 3 forces RMAN to include maximal 3 files in a backup set. A possible restore from a single datafile will be processed faster.
- format '%d%t%s%p' creates a TSM low level name including database name/timestamp/backup set number/backup piece number
- tag 'paris offline 20010316.1111' is a user defined symbolic name which is assigned to backup sets. It makes it easier to address these backups in RMAN restore or change commands.
- alter database open finally makes the database accessible to users.

### 6.4.2 Archived redo log backup

The next examples show how to back up archived redo logs.

In both examples, the current log is archived before the backup archivelog is started.

The first example uses only one log archive destination as input for the RMAN backup and two different channels to speed up the process.

With the filesperset 20 RMAN creates up to 20 backup pieces in one backup set.

The delete input option tells RMAN to deletes the archive logs on disk after a successful backup:
The second archivelog backup example is more complex — but also more safe. Assume that a very important database is configured in the following way:

- Oracle uses multiple log groups
- Each group contains multiple log members
- At least two log archive destinations are in use

Because the backup archivelog all command backs up exactly one copy of each distinct log sequence number, RMAN does not put two copies of the same log sequence number into the same backup set. Furthermore, if you specify the delete input option, RMAN only deletes the specific copy of the archived redo log that it backs up.

The easiest solution in this case is to back up both copies of each archived redo log and then delete both copies. Use the like pathname parameter to indicate which destination to use. The like parameter allows you to match filenames in both archive destinations.
Refer to Figure 18 for a diagram of this complex situation.

Figure 18. Multiple log groups, archive log destinations, different tape drives in TSM
The following example shows the coding we used to backup archivelogs from two log archive destinations:

```sql
alter system archive log current;
run {
  allocate channel t1 type 'sbt_tape' parms
  'ENV=(DSMO_NODE=sol_oracle,
   DSMO_AUG_SIZE=1,
   DSMI_ORC_CONFIG=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
   DSMI_LOG=/usr/local/oracle/log/mmn_saves)';
  allocate channel t2 type 'sbt_tape' parms
  'ENV=(DSMO_NODE=sol_oracle,
   DSMO_AUG_SIZE=1,
   DSMI_ORC_CONFIG=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
   DSMI_LOG=/usr/local/oracle/log/mmn_saves)';
  backup filesperset 20
  format '%d/sbt/%s/%%p'
  archivelog like '/opt/asp/oracle/admin/paris/ARCH/%%'
    channel t1 delete input
    archivelog like '/var/oracle/admin/paris/ARCH/%%'
    channel t2 delete input);
  release channel t1;
  release channel t2;
}
```

**Note**

Oracle recommends that you archive the current online redo log file by using the `alter system archive log current` command before the archivelogs are backed up.
6.4.3 Incremental level 0 backup

This example shows how to perform an incremental backup of the database with RMAN. The level 0 backup is the base for further incremental backups with a higher level:

```sql
run {
  allocate channel t1 type 'sbt_tape' parms
  'ENV=(OSMO_NODE=sol_oracle,
   DSMtoolbar=265,
   DSMNAME_CONFIG=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
   DSMLOG=/usr/local/oracle/log/rman_saves)';
  allocate channel t2 type 'sbt_tape' parms
  'ENV=(OSMO_NODE=sol_oracle,
   DSMtoolbar=265,
   DSMNAME_CONFIG=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
   DSMLOG=/usr/local/oracle/log/rman_saves)';
  backup incremental level 0
  setsize = 265000
  format '%d/%m/%y/%s/%p'
  tag 'paris inc1v10 20010320.1400'
  database;
  release channel t1;
  release channel t2;
}
```

In this example, `setsize = 265000` specifies the maximum size of a backup set. It must be greater or equal to the largest datafile in the backup. That is another option to control the composition of the datafiles in a backup set.
Chapter 7. Day-to-day monitoring of Oracle backups on UNIX

Once the TDP for Oracle implementation and environment has been set up, as described in the preceding chapters, it is recommended that the regular day-to-day tasks be monitored and automated for more efficient execution. This chapter details some of the processes and tasks which can be used.

7.1 Automating database tasks

There are some Oracle database tasks which can be automated as detailed in the following sections. These are best determined by the Oracle DBA.

7.1.1 Starting SQL*Net and RDBMS services at system reboot

During the boot of a UNIX system, `/etc/inittab` is processed. The file `/etc/inittab` controls process dispatching by `init`. The processes most typically dispatched by `init` are daemons — but there is also a section in the `inittab`, where user defined scripts are executed.

We chose to start Oracle services when the system is switching to run level 3, which is also called extended multi-user mode. In this mode, all important system resources are available.

On both systems, AIX and Sun Solaris, we placed a kill-script and a start-script into:

- `/etc/rc.d/rc3.d` for AIX
- `/etc/rc3.d` for Sun Solaris

The start scripts first start SQL*Net services, and then the RDBMS is started. Because it runs under root authority, a superuser command to the Oracle instance owner is issued. An example of the start script follows:

```
# pmc
/etc/rc3.d
# kis = 1a "oracle"
-1w------- 1 root other 156 Mar 12 15:34 K40oracle
1w------- 1 root other 145 Mar 12 15:36 S40oracle
# more 540oracle
#!/usr/bin/ksh
su - oracle -c 'lsnrctl start'
su - oracle -c 'dbstart'
su - oracle -c 'lsnrctl dbcmp_start'
su - oracle -c 'lsnrctl start_oms'
```

© Copyright IBM Corp. 2001
• The normal way to start SQL*Net services is lsnrctl start.
• The normal database start script shipped from Oracle is dbstart. It reads the /etc/oratab file and starts for every instance which is registered in the oratab file.

7.1.2 Automating RMAN backups

In order to automate Oracle RMAN backups, you use the RMAN cmdfile option. If you specify cmdfile when invoking RMAN, then RMAN executes the file specified line by line.

We used a kornshell script which generates and executes RMAN command files. It is easier to manage a big number of different Oracle instances on the UNIX platform when all backups are performed the same way. The initial creation of the script is laborious, but you will benefit because subsequent corrections or modifications to the backup script need be done only once.

As we started to develop our own script for the backups, we first researched useful RMAN examples.

In the directory $ORACLE_HOME/rdbms/demo are five different RMAN examples. None of them uses TDP — but the RMAN commands are useful.

We also found good hints and tips in the Oracle8i Recovery Manager User’s Guide and Reference, A76990-01.

7.1.2.1 How often should backups be done?
This depends on different factors:
• Service level agreement with the data owner
• Size of the database
• Importance of the availability of the database
• The type of application using the database

Because we tried to simulate a production environment with small databases, we scheduled inconsistent whole database backups between 6:00 and 7:00 in the morning and archivelog backups between 7:00 and 19:00 every hour.

7.1.2.2 Job scheduling by the UNIX crontab program
We used the UNIX crontab program from user root to automate database and archivelog backups. The reason for using the root's crontab is the centralization aspect — the more different crontabs that are in use, the more complicated it is to find a schedule.
The entries in the `crontab` are the UNIX crontab, database, and archlog backup schedule:

```plaintext
# Oracle instance 'oslo'
# Inconsistent whole db backups: mo-fr, at 06:00
# backup archives: mo-fr, between 20:00-23:00, hourly
# Oracle instance 'berlin'
# Inconsistent whole db backups: mo-fr, at 06:30
# backup archives: mo-fr, between 23:30-00:30, hourly

00 6  * * 1-5  su - oracle -c "export ORACLE_SID=oslo ; rman save -t online"
30 6  * * 1-5  su - oracle -c "export ORACLE_SID=berlin ; rman save -t online"
00 7-18 * * 1-5  su - oracle -c "export ORACLE_SID=oslo ; rman save -t archive"
30 7-18 * * 1-5  su - oracle -c "export ORACLE_SID=berlin ; rman save -t archive"
```

7.1.2.3 Job scheduling using the TSM scheduler

A better way to schedule backups is to use the TSM scheduler, which is included in the TSM client backup/archive code.

The benefits of TSM scheduler are that:

- The TSM administrator coordinates the different client schedules from a focal point so that you do not have to login into different servers.
- With a single select, you are informed of the status from all events of all client schedules.

7.2 RMAN report, list, and crosscheck utilities

RMAN utilities provide information of the backups or image copies you have made. You can generate reports about backups and images copies with the `report` and `list` commands; and can validate your backups or image copies with the `crosscheck` command.

The `report` command performs a detailed analysis of the backups you have made. You would use this command to verify if you are implementing your backup strategy the way you intended. It can produce the following types of information from the recovery catalog:

- The database schema
- Which data files require a backup
- Backup files that are obsolete and can be deleted
- Data files that are not recoverable
The `list` command queries the recovery catalog, and its primary purpose is to determine which backup copies are available for use when recovery is needed. You require this list to locate and publish information, such as a list of backup sets containing:

- A backup of a specific data file
- Backups for a specific tablespace
- Backups for the whole database

The `crosscheck` command queries the recovery catalog and verifies whether backup sets marked `available` or `expired` are available.

- For objects stored on disk (type `disk`): The command determines whether the header of the backup piece is valid.
- For TSM objects (type `sbt_tape`): The command simply checks that the backups exist.

You can use this utility regularly to verify if your backup is still available.

Following are examples of how you can use these utilities to perform day-to-day monitoring. For the complete syntax of these utilities, please refer to the `Oracle8i Recovery Manager User’s Guide and Reference, A76990-01`.

### 7.2.1 Report utility

The `report` utility provides analysis of your backup and recovery situation. Following are examples from the `report` utility.

#### 7.2.1.1 Report target database schema

You can report the structure of the database as follows:

- This example shows an overview of the target database structure using the `REPORT SCHEMA` command:

```sql
RMAN> report schema;
RMAN-03022: compiling command: report
Report of database schema
File K-bytes Tablespace RB segs Name
---- ---------- ------------------ ------- -------------------
1  153600 SYSTEM YES /opt/app/oracle/oradata/paris/system01.dbf
2   8192 TOOLS NO /opt/app/oracle/oradata/paris/tools01.dbf
3  262144 RBS YES /opt/app/oracle/oradata/paris/rbs01.dbf
4   51200 TEMP NO /opt/app/oracle/oradata/paris/temp01.dbf
5  20480 USERS NO /opt/app/oracle/oradata/paris/users01.dbf
6  20480 INDX NO /opt/app/oracle/oradata/paris/indx01.dbf
```
You can also use the `REPORT SCHEMA` command to show the structure of the database at a prior time. For example, if you want to know the structure of the database 2 weeks ago, you can issue the following commands:

```
RMAN> report schema at time 'sysdate-14';
```

```
RMAN-03022: compiling command: report
Report of database schema

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Tablespace</th>
<th>RB segs</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>153600</td>
<td>SYSTEM</td>
<td>YES</td>
<td>/opt/app/oracle/oradata/paris/system01.dbf</td>
</tr>
<tr>
<td>2</td>
<td>8192</td>
<td>TOOLS</td>
<td>NO</td>
<td>/opt/app/oracle/oradata/paris/tools01.dbf</td>
</tr>
<tr>
<td>3</td>
<td>262144</td>
<td>RBS</td>
<td>YES</td>
<td>/opt/app/oracle/oradata/paris/rbs01.dbf</td>
</tr>
<tr>
<td>4</td>
<td>51200</td>
<td>TEMP</td>
<td>NO</td>
<td>/opt/app/oracle/oradata/paris/temp01.dbf</td>
</tr>
<tr>
<td>5</td>
<td>20480</td>
<td>USERS</td>
<td>NO</td>
<td>/opt/app/oracle/oradata/paris/users01.dbf</td>
</tr>
<tr>
<td>6</td>
<td>20480</td>
<td>INDX</td>
<td>NO</td>
<td>/opt/app/oracle/oradata/paris/indx01.dbf</td>
</tr>
</tbody>
</table>
```

### 7.2.1.2 Report on data files needing backup

You can report on data files that require backup using three options:

- **You can report on data files needing backup based on the number of days of redo log activities.** For example, your normal backup strategy calls for having daily backups. Knowing that backups fail, and that they are normally run at night, you can tolerate backup failure as long the databases do not accumulate more than 3 days of activity from the last backup. You would run this command to report on data files whose recovery requires more than 3 days of archive logs:

```
RMAN> report need backup days 3;
```

```
RMAN-03022: compiling command: report
Report of files whose recovery needs more than 3 days of archived logs

<table>
<thead>
<tr>
<th>File</th>
<th>Days</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/system01.dbf</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/tools01.dbf</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/rbs01.dbf</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/temp01.dbf</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/users01.dbf</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>/opt/app/oracle/oradata/paris/indx01.dbf</td>
</tr>
</tbody>
</table>
```

- **You can report on data files needing backup based on the number of incremental backups.** The example shows how to use the command to report on data files whose recovery requires more than 3 incremental backups:
You can report on data files needing backup based on the number of redundant backups. For example, your backup policy calls for saving at least 3 backup generations (grandfather, father, son). The example reports on data files having less than 3 redundant backups:

```
RMAN> report need backup incremental 3;
```

```
RMAN> report need backup redundancy 3;
```

7.2.1.3 Report on obsolete backups
You can report on obsolete backups to determine which ones can be deleted.

- This example shows that any backup having more than one copy is considered obsolete:

```
RMAN> report obsolete redundancy 1 device type 'sbt_tape';
```

94 Backing Up Oracle using Tivoli Storage Management
• This example shows backups that are obsolete because there is at least one backup and they were made no more than 3 days ago:

```
RMAN> report obsolete redundancy 1 until time 'sysdate-3' device type 'sbt_tape';
```

```
Type | Key | Completion Time | Filename/Handle
-------------------------------
Backup Set | 1941 | 21-MAR-01 | PARIS/424966318/90/1
Backup Piece | 1944 | 21-MAR-01 | PARIS/424966318/91/1
Backup Set | 1942 | 21-MAR-01 | PARIS/424966374/92/1
Backup Piece | 1945 | 21-MAR-01 | PARIS/424966374/93/1
Backup Set | 1946 | 21-MAR-01 | PARIS/424966374/94/1
```

• The next example shows orphaned backups that are unusable because they belong to incarnations of the database that are not direct ancestors of the current incarnation:

```
RMAN> report obsolete orphan device type 'sbt_tape';
```

```
Type | Key | Completion Time | Filename/Handle
-------------------------------
Backup Set | 2194 | 26-MAR-01 | OSLO/425411996/16/1
Backup Piece | 2197 | 26-MAR-01 | OSLO/425411996/17/1
Backup Set | 2211 | 27-MAR-01 | OSLO/425461354/19/1
Backup Piece | 2214 | 27-MAR-01 | OSLO/425461354/20/1
Backup Set | 2212 | 27-MAR-01 | OSLO/425461354/21/1
Backup Piece | 2215 | 27-MAR-01 | OSLO/425461354/22/1
Backup Set | 2213 | 27-MAR-01 | OSLO/425461354/23/1
Backup Piece | 2216 | 27-MAR-01 | OSLO/425461354/24/1
```
7.2.1.4 Report on unrecoverable data files
To make sure that all data files are recoverable, use the \texttt{REPORT UNRECOVERABLE} command, which reports on data files that need backup because of unlogged changes made after the last backup, as follows:

```sql
RMAN> report unrecoverable device type 'sbt_tape';
RMAN-03022: compiling command: report
Report of files that need backup due to unrecoverable operations
File Type of Backup Required Name
---- ----------------------- -----------------------------------
RMAN>
```

7.2.2 List utility
The \texttt{list} utility queries the contents of the recovery catalog or the control file, primarily to determine which backup or copies are available. Examples of the \texttt{list} utility are given in the following sections.

7.2.2.1 List database incarnations
The next example lists incarnation of databases that are cataloged within the recovery catalog. Every time a \texttt{RESETLOGS} is performed, an incarnation is created for the database.

```sql
RMAN> list incarnation;
RMAN-03022: compiling command: list
List of Database Incarnations
DB Key Inc Key DB Name DB ID CUR Reset SCN Reset Time
------- ------- -------- ---------------- --- ---------- ----------
 640  641 BEACH  24844426 YES 262236 12-MAR-01
2060 2070 UNKNOWN  948509196 NO  1 09-MAR-01
2060 2061 BERLIN  948509196 NO 264394 23-MAR-01
2060 2117 BERLIN  948509196 NO 264583 26-MAR-01
2060 2128 BERLIN  948509196 YES 264583 26-MAR-01
2060 2103 BERLIN  948509196 NO 264633 26-MAR-01
1654 1655 OSLO 2939852137 NO  1 09-MAR-01
1654 2142 OSLO 2939852137 YES 264394 26-MAR-01
  1   2 PARIS  3929710371 YES  1 05-MAR-01
```
### 7.2.2.2 List old backups

You can list old backups to determine obsolete backups. The next example lists backups older than 5 days:

```sql
RMAN> list backup completed before 'sysdate-5';
RMAN-03022: compiling command: list

List of Backup Sets
--- Key Recid Stamp LV Set Stamp Set Count Completion Time
--- 1941 77 424966369 0 424966318 90 21-MAR-01

List of Backup Pieces
--- Key Pc# Cp# Status Completion Time Piece Name
--- 1944 1 1 AVAILABLE 21-MAR-01 PARIS/424966318/90/1

List of Datafiles Included
--- File Name LV Type Ckp SCN Ckp Time
--- 4 /opt/app/oracle/oradata/paris/temp01.dbf 0 Full 157535 21-MAR-01
--- 5 /opt/app/oracle/oradata/paris/users01.dbf 0 Full 157535 21-MAR-01
--- 6 /opt/app/oracle/oradata/paris/indx01.dbf 0 Full 157535 21-MAR-01

List of Backup Sets
--- Key Recid Stamp LV Set Stamp Set Count Completion Time
--- 1942 78 424966428 0 424966318 91 21-MAR-01

List of Backup Pieces
--- Key Pc# Cp# Status Completion Time Piece Name
--- 1945 1 1 AVAILABLE 21-MAR-01 PARIS/424966318/91/1

List of Datafiles Included
--- File Name LV Type Ckp SCN Ckp Time
--- 1 /opt/app/oracle/oradata/paris/system01.dbf 0 Full 157535 21-MAR-01
--- 2 /opt/app/oracle/oradata/paris/tools01.dbf 0 Full 157535 21-MAR-01

List of Backup Sets
--- Key Recid Stamp LV Set Stamp Set Count Completion Time
--- 1943 79 424966444 0 424966374 92 21-MAR-01

List of Backup Pieces
--- Key Pc# Cp# Status Completion Time Piece Name
--- 1946 1 1 AVAILABLE 21-MAR-01 PARIS/424966374/92/1

List of Datafiles Included
--- File Name LV Type Ckp SCN Ckp Time
--- 3 /opt/app/oracle/oradata/paris/rbs01.dbf 0 Full 157535 21-MAR-01
```
7.2.2.3 List backup of certain objects

By default, the list utility lists backups made for the target database. You can restrict the list to specific objects. The following are examples:

- To find out if there is a backup of the controlfile within the last 24 hours:

```
RMAN> list backup of controlfile completed after 'sysdate-1';
RMAN-03022: compiling command: list

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- ----- ------- -------------- --------- ---------- ----------------------
2402 90 425488744 0 425488651 103 27-MAR-01

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- -------------- ---------------------- ------------------------
2405 1 1 AVAILABLE 27-MAR-01 PARIS/425488651/103/1

Controlfile Included
Ckp SCN Ckp time
---------- ---------------
198333 27-MAR-01
```

- To find out if you can recover to a point in time for tablespace USERS before March 22, 2001:

```
RMAN> list backup of tablespace 'USERS' completed before '22-MAR-01';
RMAN-03022: compiling command: list

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- ----- ------- -------------- --------- ---------- ----------------------
1941 77 424966369 0 424966318 90 2001/03/21 14:12:49

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- -------------- ---------------------- ------------------------
1944 1 1 AVAILABLE 2001/03/21 14:12:49 PARIS/424966318/90/1

List of Datafiles Included
File Name LV Type Ckp SCN Ckp Time
---- ------------------------------------- -- ---- ---------- -------------
5 /opt/app/oracle/oradata/paris/users01.dbf Full 157535 2001/03/21 16:11:43
```
To list all backups of datafile `/opt/app/oracle/oradata/paris/system01.dbf` that have been backed up to TSM:

```
RMAN> list backup of datafile '/opt/app/oracle/oradata/paris/system01.dbf'
2> device type 'sbt_tape';
RMAN-03022: compiling command: list
List of Backup Sets
Key   Recid  Stamp   LV Set Stamp  Set Count  Completion Time
------- -------- ---------- -------------- ------------- ----------------------
1942   78      424966428 0 424966318  91  2001/03/21 14:13:48

List of Backup Pieces
Key   Pc#    Cp#   Status Completion Time   Piece Name
------- ---- ---- ----------- ---------------------- ------------------------
1945  1  1   AVAILABLE  2001/03/21 14:13:48 PARIS/424966318/91/1

List of Datafiles Included
File Name   LV Type   Ckp SCN  Ckp Time
---------- ---- ---------- ---- ----------
1 /opt/app/oracle/oradata/paris/system01.dbf 0   Full  157535  2001/03/21 14:11:43

List of Backup Sets
Key   Recid  Stamp   LV Set Stamp  Set Count  Completion Time
------- -------- ---------- -------------- ------------- ----------------------
2022  85      425130529 0 425130432  98  2001/03/23 11:48:49

List of Backup Pieces
Key   Pc#    Cp#   Status Completion Time   Piece Name
------- ---- ---- ----------- ---------------------- ------------------------
2025  1  1   AVAILABLE  2001/03/23 11:48:49 PARIS/425130432/98/1

List of Datafiles Included
File Name   LV Type   Ckp SCN  Ckp Time
---------- ---- ---------- ---- ----------
1 /opt/app/oracle/oradata/paris/system01.dbf 0   Full  178171  2001/03/23 11:46:58

List of Backup Sets
Key   Recid  Stamp   LV Set Stamp  Set Count  Completion Time
------- -------- ---------- -------------- ------------- ----------------------
2402  90      425488764 0 425488651 103  2001/03/27 15:19:04

List of Backup Pieces
Key   Pc#    Cp#   Status Completion Time   Piece Name
------- ---- ---- ----------- ---------------------- ------------------------
2405  1  1   AVAILABLE  2001/03/27 15:19:04 PARIS/425488651/103/1

List of Datafiles Included
File Name   LV Type   Ckp SCN  Ckp Time
---------- ---- ---------- ---- ----------
1 /opt/app/oracle/oradata/paris/system01.dbf 0   Full  198334  2001/03/27 15:17:33
```
To list backups of archive logs between SCN 178172 and 178175:

```
RMAN> list backup of archivelog from scn 178172 until scn 178175;

RMAN-03022: compiling command: list

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- --------- -------------- -------------- -------------- ----------------------
2039  87  425130567  0  425130566  100  2001/03/23 11:49:27

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- ----------- ---------------------- ------------------------
2041  1  1 AVAILABLE  2001/03/23 11:49:27 PARIS/425130566/100/1

List of Archived Logs Included
Thrd Seq Low SCN Next SCN Low Time Next Time
----- --------- --------- ---------- --- --- ----------------------

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- --------- -------------- -------------- -------------- ----------------------
2040  88  425130567  0  425130566  101  2001/03/23 11:49:27

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- ----------- ---------------------- ------------------------
2042  1  1 AVAILABLE  2001/03/23 11:49:27 PARIS/425130566/101/1

List of Archived Logs Included
Thrd Seq Low SCN Next SCN Low Time Next Time
----- --------- --------- ---------- --- --- ----------------------
```
7.2.3 Crosscheck utility

The crosscheck utility verifies whether backups still exist on disk or TSM. RMAN does not delete backup entries that it could not find, but instead marks them as expired. If the backup was erroneously marked expired, for example, because TSM was unavailable or misconfigured, the crosscheck utility will mark it available the next time it is run if the backup still exists.

Like the list utility discussed in 7.2.2.3, “List backup of certain objects” on page 98, you can run the crosscheck utility on certain objects. See the Oracle8i Recovery Manager User Guide and Reference, A76990-1, for the complete syntax of the crosscheck utility.

Here is a sample script to crosscheck all backup entries in the recovery catalog or control file:

```
allocate channel for maintenance type 'sbt_tape'
  parms 'ENV=(DSMO_NODE=sol_oracle,DSMO_AVG_SIZE=256,
  DSMI_ORC_CONFIG=/opt/tivoli/tsm/client/oracle/bin/dsm.opt,
  DSMI_LOG=/usr/local/oracle/log)';
crosscheck backup;
exit;
```

7.3 Automatic deletion of old backups

Oracle provides a sample UNIX shell script to delete obsolete backups. The example is in $ORACLE_HOME/rdbms/demo/rman1.sh. You can customize the script to your needs.

Next we show a very simplified sample script to delete backups made under TSM. It maintains copies that were created within the last three days, and at least three copies. (Please note that great care should be taken to ensure that any deletion of backups is planned and tested before implementation in a production environment.)
7.4 List objects in the TSM database

To display objects in the TSM catalog stored by TDP for Oracle, you can use standard SQL SELECT statements from the command line from an administrative client console.

To help you find what information is available in the database, TSM provides three system catalog tables:

- **SYSCAT.TABLES** contains information about all tables that can be queried.
- **SYSCAT.COLUMNS** describes the columns of each table.
- **SYSCAT.ENUMTYPES** defines the valid values for each enumerated type and the order of the values for each type.
The following examples show TSM SQL SELECT statements.

- This is an example of TSM select nodes — list node related information:

```
> select * from nodes where node_name = 'SQL_ORACLE'

<table>
<thead>
<tr>
<th>NODE_NAME: SQL_ORACLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLATFORM_NAME: TDP Oracle S \n</td>
</tr>
<tr>
<td>DOMIAN_NAME: API_DOMAIN</td>
</tr>
<tr>
<td>PASET_TIME: 2001-03-08 17:15:22.000000</td>
</tr>
<tr>
<td>INVALID_PW_COUNT: 0</td>
</tr>
<tr>
<td>CONTACT:</td>
</tr>
<tr>
<td>COMPRESSIION: CLIENT</td>
</tr>
<tr>
<td>ARCHDELETE: YES</td>
</tr>
<tr>
<td>BACKDELETE: YES</td>
</tr>
<tr>
<td>LOCKED: NO</td>
</tr>
<tr>
<td>LASTACC_TIME: 2001-04-02 09:26:20.000000</td>
</tr>
<tr>
<td>REG_TIME: 2001-03-08 17:15:22.000000</td>
</tr>
<tr>
<td>REG_ADMIN: ADMIN</td>
</tr>
<tr>
<td>LASTSESS_COMMAND: Tcp/Ip</td>
</tr>
<tr>
<td>LASTSESS_RECV: 0</td>
</tr>
<tr>
<td>LASTSESS_SENT: 157</td>
</tr>
<tr>
<td>LASTSESS_DUR: 158</td>
</tr>
<tr>
<td>LASTSESS_IDLE: 87</td>
</tr>
<tr>
<td>LASTSESS_COMM: 0</td>
</tr>
<tr>
<td>LASTSESS_MEDIWAIT: 0</td>
</tr>
<tr>
<td>CLIENT_VERSION: 4</td>
</tr>
<tr>
<td>CLIENT_RELEASE: 1</td>
</tr>
<tr>
<td>CLIENT_LEVEL: 2</td>
</tr>
<tr>
<td>CLIENT_SUBLEVEL: 12</td>
</tr>
<tr>
<td>CLIENT_OS_LEVEL: 5.7</td>
</tr>
<tr>
<td>OPTION_SET:</td>
</tr>
<tr>
<td>AGGREGATION: YES</td>
</tr>
<tr>
<td>URL:</td>
</tr>
<tr>
<td>NODETYPE: CLIENT</td>
</tr>
<tr>
<td>PASSSEC:</td>
</tr>
<tr>
<td>KEEP_MC: NO</td>
</tr>
<tr>
<td>MAX_NR_ALLOWED: 2</td>
</tr>
</tbody>
</table>
```
This is an example of TSM SQL SELECT backups — list backup object related information:

tsm: BRIASS> select * from backups where node_name = 'SOL_ORACLE'
order by backup_date desc

Answer? This SQL query may produce a very large result table, or may require a significant amount of time to compute.

Do you wish to proceed? (Yes(Y)/No(N)) y

<table>
<thead>
<tr>
<th>NODE_NAME: SOL_ORACLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILESPACE_NAME: /adsworc</td>
</tr>
<tr>
<td>STATE: ACTIVE_VERSION</td>
</tr>
<tr>
<td>TYPE: FILE</td>
</tr>
<tr>
<td>HL_NAME: //</td>
</tr>
<tr>
<td>LL_NAME: PARIS/425665087/116/1</td>
</tr>
<tr>
<td>OBJECT_ID: 92504</td>
</tr>
<tr>
<td>BACKUP_DATE: 2001-03-29 16:17:07.000000</td>
</tr>
<tr>
<td>DEACTIVATE_DATE:</td>
</tr>
<tr>
<td>OWNER: oracle</td>
</tr>
<tr>
<td>CLASS_NAME: DEFAULT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NODE_NAME: SOL_ORACLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILESPACE_NAME: /adsworc</td>
</tr>
<tr>
<td>STATE: ACTIVE_VERSION</td>
</tr>
<tr>
<td>TYPE: FILE</td>
</tr>
<tr>
<td>HL_NAME: //</td>
</tr>
<tr>
<td>LL_NAME: PARIS/425665087/117/1</td>
</tr>
<tr>
<td>OBJECT_ID: 92505</td>
</tr>
<tr>
<td>BACKUP_DATE: 2001-03-29 16:17:09.000000</td>
</tr>
<tr>
<td>DEACTIVATE_DATE:</td>
</tr>
<tr>
<td>OWNER: oracle</td>
</tr>
<tr>
<td>CLASS_NAME: DEFAULT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NODE_NAME: SOL_ORACLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILESPACE_NAME: /adsworc</td>
</tr>
<tr>
<td>STATE: ACTIVE_VERSION</td>
</tr>
<tr>
<td>TYPE: FILE</td>
</tr>
<tr>
<td>HL_NAME: //</td>
</tr>
<tr>
<td>LL_NAME: PARIS/425665087/116/1</td>
</tr>
<tr>
<td>OBJECT_ID: 92504</td>
</tr>
<tr>
<td>BACKUP_DATE: 2001-03-29 16:17:07.000000</td>
</tr>
<tr>
<td>DEACTIVATE_DATE:</td>
</tr>
<tr>
<td>OWNER: oracle</td>
</tr>
<tr>
<td>CLASS_NAME: DEFAULT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NODE_NAME: SOL_ORACLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILESPACE_NAME: /adsworc</td>
</tr>
<tr>
<td>more... (&lt;ENTER&gt; to continue, 'C' to cancel)</td>
</tr>
</tbody>
</table>
This is an example of TSM SQL SELECT CONTENTS — list volumes and file sizes:

```sql
TSM: BRAZIL> select * from contents where node_name = 'SQL_ORACLE' and file_name like '/PARIS/425665134/118%'

<table>
<thead>
<tr>
<th>VOLUME_NAME</th>
<th>NODE_NAME</th>
<th>FILESPACE_NAME</th>
<th>FILE_NAME</th>
<th>AGGREGATED</th>
<th>FILE_SIZE</th>
<th>SEGMENT</th>
<th>CACHED</th>
</tr>
</thead>
<tbody>
<tr>
<td>/tsm/dskpool/vol3</td>
<td>SQL_ORACLE</td>
<td>/admorc</td>
<td>/PARIS/425665134/118</td>
<td>No</td>
<td>28807705</td>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>
```
Chapter 8. Recovering Oracle Databases on UNIX

The earlier chapters in this UNIX section of the book have covered the various procedures and methods for backing up Oracle databases. Now we shall address the important topic of recovery of these databases.

8.1 Restore operations

Recovering a database, tablespace, or data file is a two-stage process. The object is first restored and then it is recovered.

The restore process restores the necessary full or incremental level 0 backups. Incremental backups at levels greater than 0 are not restored. These are restored during the subsequent recovery process.

By default, the objects are restored to their original location as specified in the recovery catalog. An alternative location can also be specified if required. RMAN uses the recovery catalog to select the most current backup sets for use in the restore. The mode in which the database is running determines whether a consistent or inconsistent restore operation can be performed.

8.1.1 Consistent database recovery

If the database is running in NOARCHIVELOG mode, it can be restored only from a whole database backup. The control files and all data files are restored from a consistent backup. The database must be mounted but not open during the restore operation.

After a consistent restore, the database can be opened without performing any recovery. Any updates to the database after the backup are lost.

8.1.2 Inconsistent database recovery

If the database is running in ARCHIVELOG mode, a subset of the database such as a data file, tablespace, or the entire database can be restored to the most current state or to a specific point in time. This type of restore is inconsistent because the database cannot be started after the restore. The restore operation must be followed by a recovery operation. After the recovery is finished, the database can then be opened. The following objects can be restored:

- Database
- Tablespace
- Data file
• Control file
• Archive log

If the control files are lost, they must be restored before other restore operations can be performed. Only after restoring the control files can the target database be mounted and the other restore operations started.

A restore operation is typically followed by a recovery operation.

### 8.2 Recovery Operation

Recovery is a process whereby a restored file is made available, either to the most current state or to a specific point in time.

Once the data files are restored, they have to be made consistent with each other. The `recover` command is used to perform media recovery and to apply incremental backups. During the recovery process, RMAN automatically restores the archived redo logs as required.

If RMAN has a choice between restoring incremental backup sets or applying redo logs, it always uses the incremental backups. If overlapping levels of incremental backup are available, the lowest level of incremental backup, the one covering the longest period of time, is chosen automatically.

The following objects can be recovered:

• **Individual data files**
  One or more data files can be recovered. The target database must be started and mounted. If the target database is open, the data file must be offline.

• **Tablespaces**
  All data files in a tablespace can be recovered or a tablespace can be recovered to a previous point in time. The target database must be started and mounted. If the target database is open, the tablespace must be offline.

• **The entire database**
  The entire database can be recovered under the following circumstances:
  - A media failure has damaged the entire database.
  - The entire database must be recovered to a previous point in time.
  - The control files have been lost.

For database recovery, the database must be started but not open.
Archive log backup sets are restored as needed to perform a recovery. They are restored to the current archive log destination as specified in the `init.ora` file.

8.3 Recovery examples

In the following sections we provide some examples of recovery using RMAN.

8.3.1 Datafile recovery

To test a datafile recovery with RMAN, we performed following steps:

- The Oracle instance `paris` was up and running.
- User `user1` in a SQL*Plus session was connected to `paris`. We created a new table in the `USERS` tablespace, inserted a row, and committed the changes.
- In a second telnet session, the `root` user destroyed the file with an `echo 'oops' > /opt/app/oracle/oradata/paris/users01.dbf`
- User `user1` inserted another row in the table and committed again.
- Oracle user `sys` connected to the database and issued an `alter system checkpoint`.
- After the checkpoint, Oracle recognized the destroyed file and took that datafile in status `RECOVER`. In the alertlog file of the instance, we saw following entries:

```
Completed: alter database open
Thu Mar 22 10:19:59 2001
Errors in file /opt/app/oracle/admin/paris/bdump/paris_ckpt_221021.trc:
ORA-01117: datafile 5 going offline due to error advancing checkpoint
ORA-01122: database file 5 failed verification check
ORA-01110: data file 5: '/opt/app/oracle/oradata/paris/users01.dbf'
ORA-01251: Unknown File Header Version read for file number 5
```

```
SQL> select status from v$datafile
  2  where NAME = '/opt/app/oracle/oradata/paris/users01.dbf' ;

STATUS
    RECOVER
```
• Next, we created the following RMAN command file and executed it:

```
run {
    allocate channel t1 type 'sbt_tape' parms
        'ENV=(DSM_NODE=sql_oracle, 
          DSM_RPC_PORT=/opt/tivoli/tivoli/client/oracle/bin/dsm.opt, 
          DSM_LOG=/var/log/oracle/log/rman_saves)';
    restore datafile '/opt/app/oracle/oradata/paris/users01.dbf';
    recover datafile '/opt/app/oracle/oradata/paris/users01.dbf';
    sql 'alter database datfile '/opt/app/oracle/oradata/paris/users01.dbf' online';
    release channel t1;
}
```

• Unfortunately, RMAN was not able to take to datafile online after the successful restore and recover. We had to execute exactly the same command in a SQL*Plus session, and it worked fine:

```
SQL> alter database datfile
    2 '/opt/app/oracle/oradata/paris/users01.dbf' online;
Database altered.
```

• Finally, user user1 was able to select all rows from its table.

### 8.3.2 Complete recovery

Our database oslo on an AIX system failed, because of a handling error with SAN zoning and the AIX lost access to that disk device. We had a file system /u02/oradata on that drive and some datafiles, an archivelog destination and a control file in this file system. The database was running in **ARCHIVELOG** mode and an incomplete backup, started by **cron**, which was running just before that happened. We had to recover the database to a most current state. To achieve that, we performed following steps:

• We edited the **initoslo.ora** file in the **$ORACLE_HOME/dbs** directory:
  
  - We changed the **log_archive_dest_2** to /u03/oradata/oslo/arch instead of /u02/oradata/oslo/arch.
  
  - We changed the **control_files** parameter. Originally, we had control files in /u01/oradata/oslo, /u02/oradata/oslo and /u03/oradata/oslo - we simply removed the string containing the /u02/oradata/oslo.
  
• We created a new directory for the second log archive destination by the command **mkdir -p /u03/oradata/oslo/arch.**
- We started the database in nomount mode with `startup nomount`.
- We invoked RMAN and performed a `report schema` to list the file id’s, the size and the corresponding file names.
- We increased the space of file system `/u03/oradata - the df -k`:
  `/u03/oradata` lists the amount of free KB in the file system, the output from the RMAN report schema lists the KB allocated for a single datafile.
- We created a RMAN command file including the options `set newname` for the datafiles originally located on `/u02/oradata/oslo`.

```sql
oracle@brazil => more recover.oslo
shutdown abort;
startup nomount;
run {
  allocate channel t1 type 'sbt_tape' parms
  'EN=(PSWO_NODE=brazil_oracle,
   DSM_MACHINE=us/client/oracle/bin/dsm.opt,
   DSM_LOCK=/usr/local/oracle/bin/dsm.opt,
   DSM_LOG=/usr/local/oracle/bin/dsm.opt)',
  allocate channel t2 type 'sbt_tape' parms
  'EN=(PSWO_NODE=brazil_oracle,
   DSM_MACHINE=us/client/oracle/bin/dsm.opt,
   DSM_LOCK=/usr/local/oracle/bin/dsm.opt)',
  set newname for datafile 4 to '/u03/oradata/oslo/rdb02.dbf';
  set newname for datafile 5 to '/u03/oradata/oslo/rdb02.dbf';
  set newname for datafile 7 to '/u03/oradata/oslo/data01.dbf';
  restore database;
  alter database mount;
  switch datafile all;
  recover database;
  alter database open;
  release channel t1;
  release channel t2;
}
```

- In this example:
  - `shutdown abort` and `startup nomount` are included in the script because we had to run it several times. This way, it is guaranteed that the status of the instance (STARTED) is correct.
  - `set newname` is used for the `restore` and for the `switch` command.
  - `restore` restores all datafiles; file 4, 5, and 7 are renamed during the `restore`.
  - `alter database mount` is required because of the following `recover` command.
  - `switch datafile all` renames datafiles 4, 5, and 7 in the control files — it is working like `alter database oslo rename file 'xxx' to 'yyy'`.
  - `recover database` applies redo log records.
  - `alter database open` finally makes the database available.
Some hours after the successful recover, the failed disk went online again. We found not only the suspected datafiles, control file, and log archive directory, but also redo log files in the /u02/oradata file system and a datafile belonging to a temporary tablespace.

We queried the database and recognized that only every second member of each group was INVALID, but after a log switch they were usable again:

```
SVRMGR> select * from v$logfile ;
GROUP# STATUS MEMBER
1 STALE /u02/oradata/oslo/redo01.log
1 INVALID /u02/oradata/oslo/redo01a.log
2 INVALID /u02/oradata/oslo/redo02.log
3 INVALID /u02/oradata/oslo/redo03.log
3 INVALID /u02/oradata/oslo/redo03a.log
6 rows selected.
SVRMGR> alter system switch logfile ;
Statement processed.
SVRMGR> select * from v$logfile ;
GROUP# STATUS MEMBER
1 STALE /u02/oradata/oslo/redo01.log
1 INVALID /u02/oradata/oslo/redo01a.log
2 INVALID /u02/oradata/oslo/redo02.log
2 INVALID /u02/oradata/oslo/redo02a.log
3 INVALID /u02/oradata/oslo/redo03.log
3 INVALID /u02/oradata/oslo/redo03a.log
6 rows selected.
```

### 8.3.3 Incomplete recovery

This example shows how to restore and recovery a database to a specific point in time.

Incomplete recovery is sometimes also database point in time recovery or DBPITR called.

To test an incomplete recovery, we did these steps:

- We started the Oracle instance berlin.
- We performed a consistent whole database backup with tag = ‘berlin offline 20010326.0925’.
- We created new tables and inserted some rows.
- We archived the redo log files.
- We stopped the Oracle instance berlin by a shutdown abort.
- We removed all control files.
• We started the instance *berlin* with the `mount` option:

$ export ORACLE_SID=berlin
$ sqlplus /NOCOPY

(c) Copyright 2000 Oracle Corporation. All rights reserved.

SQL> connect internal;
Connected to an idle instance.
SQL> startup nomount;
ORACLE instance started.
Total System Global Area 42712960 bytes
Fixed Size 73620 bytes
Variable Size 20083328 bytes
Database Buffers 1030400 bytes
Redo Buffers 172032 bytes
SQL> exit
Disconnected

• We listed the last backup using `tag = 'berlin offline 20010326.0925'`:

```
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-08005: connected to target database: BERLIN (DBID=948508195)
RMAN-08008: connected to recovery catalog database
RMAN> list backup of database tag = 'berlin offline 20010326.0925';
RMAN-03022: compiling command: list
List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
2072 1 425381234 0 425381233 2 25-MAR-01

List of Backup Pieces
Key P# Cmp Status Completion Time Piece Name
2075 1 1 AVAILABLE 25-MAR-01 BERLIN/425381233/2/1

List of Datafiles Included
File Name LV Type Ckp SCN Ckp Time
4 /u02/oradata/berlin/rbs02.dbf D Full 264592 25-MAR-01
5 /u02/oradata/berlin/users01.dbf D Full 264592 25-MAR-01
6 /u02/oradata/berlin/index01.dbf D Full 264592 25-MAR-01

List of Backup Sets
```
• We created an RMAN command file including the checkpoint SCN from this listing:

```bash
$ more recover.berlin
run {  
  allocate channel t1 type 'sbt_tape' parms 'SERVER=brazil_oracle,  
  DSM_MIG_CONFIG=/usr/tivoli/tsm/client/oracle/bin/dsm.opt,  
  DSM_TAP=/usr/local/oracle/bin/dsm.opt';
  allocate channel t2 type 'sbt_tape' parms 'SERVER=brazil_oracle,  
  DSM_MIG_CONFIG=/usr/tivoli/tsm/client/oracle/bin/dsm.opt,  
  DSM_TAP=/usr/local/oracle/bin/dsm.opt';
  set until scn = 264582;
  restore controlfile;
  restore database;
  alter database mount;
  recover database delete archivelog;
  alter database open resetlogs;
  release channel t1;
  release channel t2;
}
```

• In this example:
  - `set until scn = 264582` specifies a SCN for the subsequent `restore` and `recovery` commands.
  - `restore controlfile` causes RMAN to restore the controlfiles to all control file destinations. The control files are not restored by the `restore database` command.
  - `restore database` restores all datafiles belonging to the database. Existing datafiles are overwritten.
  - `alter database open resetlogs` is the only way to open the database again, because the data in the redo logs cannot be used any longer.
  - After a successful database point in time recovery, a backup of the database is required.

### 8.3.4 Duplicate database

The RMAN duplicate command allows you to use your target database backups to create a test database while still retaining your original database. The command takes backup sets or image copies of your target database files and generates a new database. A duplicate database is especially useful if your production database must be up and running 24 hours per day, 7 days a week.
As part of the duplication operation, RMAN manages the following:

- Restores the target datafiles into the duplicate database and performs incomplete recovery using all available archived redo logs and incremental backups.
- Opens the duplicate database with the \texttt{RESETLOGS} option after incomplete recovery to create the online redo logs.
- Generates a new, unique database identifier for the duplicate database.

When duplicating a database, you can do the following:

- Skip read-only tablespaces with the \texttt{skip readonly} clause. Read-only tablespaces are included by default. If you omit them, you can add them later.
- Create your duplicate database in a new host. If the directory structure is the same on the new host, you can use the \texttt{nofilenamecheck} option and reuse the target datafile filenames for the duplicate datafiles.
- Use the \texttt{set until} option when creating the duplicate database to recover it to a non-current time. By default, the \texttt{duplicate} command creates the database using the most recent backups of the target database and then performs recovery to the most recent consistent point contained in the incremental backups and archived redo logs.
- Use the \texttt{duplicate} command without a recovery catalog.
- Register the duplicate database in the same recovery catalog as the target database. This option is possible because the duplicate database receives a new database identifier during duplication. If you copy the target database using operating system utilities, the database identifier of the copied database remains the same so you cannot register it in the same recovery catalog.

In our test environment, we duplicated the database \texttt{oslo} to the database \texttt{berlin}. Both databases are located on the same AIX system.

We performed the following steps:

- We created an Oracle password file for the new instance \texttt{berlin}:

  \begin{verbatim}
  cd $ORACLE_HOME/dbs
  orapwd file=orapwberlin password=change_on_install entries=10
  \end{verbatim}
- We created an `init.ora` file for instance `berlin`. In our case, we copied the `initoslo.ora` file to `initberlin.ora` file and changed these entries:

```bash
$ more initberlin.ora
grdap

db_name = "berlin"
instance_name = berlin
service_name = berlin.ainoden.tum.com
control_files = ("/u01/oradata/berlin/control01.ctl",
	"/u02/oradata/berlin/control02.ctl")
background_dump_dest = /u01/oradata/berlin/bdump
core_dump_dest = /u01/oradata/berlin/cdump
user_dump_dest = /u01/oradata/berlin/udump
```

- We created the necessary directories for datafiles, archived redo logs and dump destinations.
- We started the new instance `berlin`:

```bash
$ export ORACLE_SID=berlin
$ srvrmgrl

Oracle Server Manager Release 3.1.7.0.0 - Production
Copyright (c) 1982, 1993, Oracle Corporation. All Rights Reserved.
Oracle Enterprise Edition Release 8.1.7.0.0 - Production
with the Partitioning option
Server Release 8.1.7.0.0 - Production

SURMGRL> connect internal
Connected.
SURMGRL> status nonmount :
Oracle instance started.
Total System Global Area 42712980 bytes
Fixed Size 73620 bytes
Variable Size 25063328 bytes
Database Buffers 16384000 bytes
Redo Buffers 172032 bytes
SURMGRL>
```

- We ensured connectivity to the new instance:
  - We edited `$ORACLE_HOME/network/admin/listener.ora` and add an entry for `berlin`.
  - We edited `$ORACLE_HOME/network/admin/tnsnames.ora` and add an entry for `berlin`.
  - We reloaded the SQL*Net listener by the command `lsnrctl reload`. 

116  Backing Up Oracle using Tivoli Storage Management
We created an RMAN command file for the duplicate process:

```
$ more duplicate.oslo.to.berlin
run {
  allocate auxiliary channel t1 type 'sbt_tape' parms
    'REN=DSMO_NODE=brazil_oracle,DSMO_DEBUG=49,
     DSMO_LOG_CONFIG=/usr/tivoli/tivcmdClient/oracle/bin/dsm.opt,
     DSMO_LOG=/usr/local/oracle/bin/';
  set newname for datafile 1 to '/u01/oradata/berlin/system01.dbf';
  set newname for datafile 2 to '/u01/oradata/berlin/tools01.dbf';
  set newname for datafile 3 to '/u02/oradata/berlin/bs01.dbf';
  set newname for datafile 4 to '/u02/oradata/berlin/bs02.dbf';
  set newname for datafile 5 to '/u01/oradata/berlin/users01.dbf';
  set newname for datafile 6 to '/u02/oradata/berlin/index01.dbf';
  set newname for datafile 7 to '/u02/oradata/berlin/data01.dbf';
  duplicate target database to berlin
  logfile
    group 1 ('/u01/oradata/berlin/rad01.log',
             '/u02/oradata/berlin/rad02.log') size 500K reuse,
    group 2 ('/u01/oradata/berlin/rad01.log',
             '/u02/oradata/berlin/rad02.log') size 500K reuse;
}
```

In this example:
- The keyword `auxiliary` in the `allocate channel` command specifies a connection between RMAN and the new instance `berlin`.
- `set newname` is used to rename datafiles during restore. To produce a list of file ids and filenames, you may use the output of RMAN `report schema`.

```
$ rman target sys/change_on_install@oslo rcvcast rman/rman@BID
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-03005: connected to target database: OSLO (DBID=2930852137)
RMAN-03008: connected to recovery catalog database
RMAN> report schema;
RMAN-03022: compiling command: report
Report of database schema
Tablespace Name
file  k-bytes  YES  /u01/oradata/oslo/system01.dbf
1  265240  SYSTEM
2  528384  RS01
3  528384  RS01
4  528384  RS02
5  104356  USERS
6  55250  INDEX
7  108576  DATA01
```

Chapter 8. Recovering Oracle Databases on UNIX  117
• In this example:
  - The syntax for the \texttt{logfile} options is the same used in the \texttt{create database} command.

• We invoked RMAN to duplicate the database:

  \texttt{rman target sys/change_on_installsOSLO auxiliary sys/change_on_installsBERLIN rcvcat rman/rman@RIO cmdfile=duplicate.oslo.to.berlin log=duplicate.oslo.to.berlin.log}

• In this example:
  - RMAN must be connected to the target database:
    • This corresponds to the source database, status open.
  - And to the auxiliary database:
    • This corresponds to the target database, status mount.
  - And optionally to the recovery catalog database.
Chapter 9. Installation of TSM software on Windows 2000

In order to back up Oracle databases using TDP for Oracle for Windows, you must perform the following steps:

1. Register a node on the TSM server.
2. Stop all Oracle services.
3. Install TDP for Oracle for Windows.
4. Configure the client options file (dsm.opt).
5. Generate the encrypted password using aobpswd.exe.
6. Test a TDP for Oracle for Windows RMAN backup.
7. Install the TSM backup-archive client (optional).

The TSM backup-archive client is not required for performing TDP for Oracle for Windows RMAN backups. However, you do need the TSM backup-archive client in order to use TSM scheduling and to perform file level (operating system) backups of the Oracle database files using TSM.

9.1 Registering a node on the TSM Server for TDP for Oracle backups

The first step in configuring Oracle to back up to a TSM server is to register a node with the TSM server. Before registering the node, you should create a new storage policy on the TSM server for the nodes that will be performing TDP for Oracle backups. The storage policy consists of a policy domain, policy set, management classes, backup copygroups, and archive copygroups. This is discussed in Chapter 4, “TSM server considerations” on page 33.

On our TSM server we have already defined a new policy domain called ‘API_DOMAIN’, a policy set ‘API_POLICY’, a management class ‘API_MGMTCLASS’. The management class contains both a backup copygroup and an archive copygroup.

The backup copygroup has retention settings of VERExists=1, VERDelete=0, RETEExtra=0, RETOnly=0, and a destination to a disk storage pool. The rest of the values were left as defaults when defining the backup copy group. The backup copy group will be used for all the data objects that are sent by TDP for Oracle to the TSM server.
With this storage policy defined, the nodename of JAMAICA_ORACLE was registered with the TSM server. We made sure to specify that the node could delete backup objects with the BACKDELETE=YES option. If this option was not set, the RMAN delete command would fail. Also, we set the maximum number of mount points to 2, corresponding to the number of drives:

```
tsck: BRAZIL>register node jamaica_oracle jamaica_oracle domain=api_domain backdelete=yes maxnummp=2
ANR2060I Node JAMAICA_ORACLE registered in policy domain API_DOMAIN.
ANR2099I Administrative userid JAMAICA_ORACLE defined for OWNER access to node JAMAICA_ORACLE.
```

9.2 Stopping Oracle services

You should stop all Oracle services before installing TDP for Oracle for Windows to ensure that all the files are correctly installed. The Oracle Administration Guide for NT contains different methods for stopping these services. We stopped the Oracle services using the Windows 2000 Services tool. You access this tool through the Control Panel or by selecting Start -> Programs -> Administrative Tools -> Services (Figure 19).

![Figure 19. Windows Services tool](image-url)
From the Services tool, we selected each Oracle service and stopped it. In a production environment, you would need to coordinate this with the DBA and users, as this will prevent any access to the databases (Figure 20).

![Figure 20. Stopping the Oracle services](image)

9.3 Installing TDP for Oracle for Windows

The latest level of TDP for Oracle at the time of this publication was 2.1.10. If you have an older level of TDP for Oracle for Windows, you must first install the older level from the CD-ROM. This installs the license file (agent.lic) that is needed. Then you should download the latest level of TDP for Oracle for Windows from an IBM download site.

The version on the FTP site contains all files except for the agent.lic license file. If you have paid for the license (license is per client), you could copy this file from a machine that already has TDP for Oracle installed, install the 2.1.10 version from the FTP site, and finally copy the agent.lic file into the ...\AgentOBA directory. This would bypass installing the older version from the CD-ROM (Figure 21).
After stopping all the Oracle Services, we inserted the TDP for Oracle for Windows 2.1.10 CD-ROM and ran the `setup.exe` executable (Figure 22).

![Figure 22. First screen of TDP for Oracle for Windows 2.1.10 installation](image)
Select **Next** and you will be presented with the following screen. We chose a typical install and selected **Next** (Figure 23).

![Setup options](image)

**Figure 23. Type of install**
We wanted to change the installation directory to be the same as the default directory for the TSM backup-archive client, so we selected Browse (Figure 24).

Figure 24. Select destination directory
We chose the installation directory to be ‘c:\program files\tivoli\tsm’ so that it would be the same as the default path of the backup-archive client (Figure 25).

Figure 25. Choosing the installation directory

The destination directory was changed; we then selected **Next** (Figure 27).

Figure 26. Changed destination directory
We left the program folder as the default value (Figure 27).

Figure 27. Select program folder
The next screen shows the settings that were selected (Figure 28).

![Figure 28. Confirmation screen](image-url)
The next screen was the registration screen. If we had been an actual customer, we would have made the extra effort to fill in the registration screen. After the registration screen, the final screen confirmed that the install was complete (Figure 29).

Figure 29. Installation completed

9.4 Configuring the TDP for Oracle client options file

After installing TDP for Oracle for Windows, you need to configure the client options file (dsm.opt). There is an icon in the Tivoli Storage Manager program folder that opens the dsm.opt file using notepad.exe. You can use whatever text editor you wish.

This file is located in the directory <TDP-installpath>\AgentOBA. In our case, this directory is ‘c:\program files\Tivoli\TSM\AgentOBA’ (Figure 30).
We then edited the dsm.opt client options file using Notepad. We specified values for:

- COMMMETHOD
- TCPSERVERADDRESS
- TCPPORT
- NODENAME
- PASSWORDACCESS

Our TSM server is on an RS/6000 AIX machine with a TCP/IP address of 193.1.1.11 and is listening for client connections on port 1500. Therefore, we chose TCPIP as our communication method, 193.1.1.11 as our TCP server address, and 1500 as our TCP port. We used the node that we registered at the beginning of the chapter for the nodename. We set PASSWORDACCESS to GENERATE (Figure 31).
9.5 Generating the encrypted password

Once the client options file is configured, we can now run the `aobpswd.exe` to generate an encrypted password in the registry. To do this, we opened a command prompt and changed to the `AgentOBA` directory. This executable asks for the current password, a new password, and confirmation of the new password (Figure 32).

Note

TDP for Oracle for Windows requires PASSWORDACCESS to be set to GENERATE. This is unlike TDP for Oracle for UNIX, which requires a setting of PROMPT.
We rebooted the system for the changes to take effect.

9.6 Verify a TDP for Oracle for Windows RMAN backup

The best method for verifying that you have correctly configured TDP for Oracle for Windows is to take an RMAN backup of type sbt_tape and specify DSMO_DEBUG=49 when you allocate the channel. We recommend using a non-production database for this test purpose.

9.6.1 Starting RMAN from a command prompt

Open a command prompt window and type `set ORACLE_SID=<SID>`. Replace `<SID>` with the SID of your database. Our database’s SID was BEACH so we entered `SET ORACLE_SID=BEACH`.

Then type `rman nocatalog` from the command prompt. This will bring you to the RMAN prompt. You now need to connect to the desired database. The syntax is `connect target user/password@<SID>`. Where `<SID>` once again is the SID of the database. The user/password must be a user and password that has the appropriate Oracle privileges to shutdown the database, mount the database, and perform RMAN backups. The results of entering the three commands are shown in Figure 33.
If you do not know what your SID is, go to the SERVICES administrative tool and look for a service name like OracleService<NAME>, where the <NAME> is your SID. You may have multiple services with the prefix OracleService corresponding to multiple databases. Make sure that this service is running, or else RMAN will fail when connecting to that database.

The user INTERNAL is an Oracle default user that every database has. If you have not manually changed the password, the password will be set to the default of change_on_install.

Once you have opened a command prompt, set the ORACLE_SID environment variable, start RMAN with the nocatalog option, and connect to the database you wish to back up. You are now ready to enter the commands to perform the backup. The following steps are for taking a full backup of a database that is in NOARCHIVELOG mode.

1. **Command:** shutdown immediate

   ```sql
   RMAN> shutdown immediate
   RMAN-06405: database closed
   RMAN-06404: database dismounted
   RMAN-06402: Oracle instance shut down
   ```
2. **Command:** `startup force dba`

```
RMAN> startup force dba
RMAN-06196: Oracle instance started
RMAN-06199: database mounted
RMAN-06400: database opened
Total System Global Area 237856796 bytes
Fixed Size 75804 bytes
Variable Size 80416768 bytes
Database Buffers 157286400 bytes
Redo Buffers 77824 bytes
```

3. **Command:** `shutdown immediate`

```
RMAN> shutdown immediate
RMAN-06405: database closed
RMAN-06404: database dismounted
RMAN-06402: Oracle instance shut down
```

4. **Command:** `startup mount`

```
RMAN> startup mount
RMAN-06193: connected to target database (not started)
RMAN-06196: Oracle instance started
RMAN-06199: database mounted
Total System Global Area 237856796 bytes
Fixed Size 75804 bytes
Variable Size 80416768 bytes
Database Buffers 157286400 bytes
Redo Buffers 77824 bytes
```

The previous four steps ensure that the database was brought down cleanly, is in a consistent state, and is ready to be backed up. The next four steps run the actual backup:

5. **Command:** `run {
6. **Command:** `allocate channel t1 type 'sbt_tape' parms
7. **Command:** `ENV=(DSMO_DEBUG=49)';
8. **Command:** `backup (database);}

Chapter 9. Installation of TSM software on Windows 2000
After entering step 8, RMAN executed the backup. Since we did not specify a specific format to use, RMAN generated a unique name for the backup piece. In our case, the name of this piece is `0ecl33t7_1_1`. This name can be seen from the console output. The console output of this backup is as follows. Both RMAN and TSM use this backup piece name for identifying the different backups:

```
RMAN> run {
  2> allocate channel t1 type 'sbt_tape' params
  3> "ENV=(DSMO_DEBUG=49);"
  4> backup (database);};
```

```
RMAN-03022: compiling command: allocate
RMAN-03023: executing command: allocate
RMAN-08030: allocated channel: t1
RMAN-08500: channel t1: sid=13 devtype=SBT_TAPE
RMAN-08526: channel t1: MMS Version 2.1.10.0
RMAN-03022: compiling command: backup
RMAN-03023: executing command: backup
RMAN-08030: channel t1: starting full datafile backupset
RMAN-08010: channel t1: specifying datafile(s) in backupset
RMAN-08522: input datafile fno=00001 name=C:\ORACLE\ORADATA\BEACH\SYSTEM01.DBF
RMAN-08522: input datafile fno=00002 name=C:\ORACLE\ORADATA\BEACH\RBS01.DBF
RMAN-08522: input datafile fno=00008 name=C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA
RMAN-08522: input datafile fno=00003 name=C:\ORACLE\ORADATA\BEACH\USERS01.DBF
RMAN-08522: input datafile fno=00004 name=C:\ORACLE\ORADATA\BEACH\TMP01.DBF
RMAN-08522: input datafile fno=00006 name=C:\ORACLE\ORADATA\BEACH\INDX01.DBF
RMAN-08522: input datafile fno=00007 name=C:\ORACLE\ORADATA\BEACH\DR01.DBF
RMAN-08522: input datafile fno=00005 name=C:\ORACLE\ORADATA\BEACH\TOOLS01.DBF
RMAN-08013: channel t1: piece 1 created
RMAN-08503: piece handle=0ecl33t7_1_1 comment=API Version 1.1, MMS Version 2.1.10.0
RMAN-08525: backup set complete, elapsed time: 00:02:48
RMAN-08031: released channel: t1
```

9.6.2 Viewing the orcagent.log

Because we specified the DSMO_DEBUG=49 in the RMAN backup, the file `orcagent.log` was created. On our system, this file was located in the `C:\oracle\ora81\database\` directory.
The orcagent.log shows the following useful information for verifying that the TDP for Oracle installation is correct:

- The location of the license file and verification that it is valid
- The Agent Level as being 2.1.10
- That version 2.1.10 of TDP for Oracle was compiled with the 3.7.2 API
- That the API level installed on the system is 3.7.2 (if you upgrade the TSM API by installing the TSM backup-archive client, this value will change)
- The value for the DSMI_DIR environment variable
- The value for the DSMI_CONFIG environment variable
- TCPServeraddress of the TSM server
- Whether the compression option is on (0) or off (1)
- Whether the compressalways option is false (0) or true (1)
- Whether passwordAccess is set to prompt (0) or generate (1)

9.6.3 Listing the backup piece from RMAN

After taking the backup, you can run the RMAN command `list backup;` to view the backup piece that was taken. You can see that the backup piece name is `0ec13c7_1_1`. This corresponds to the name that was seen during the initial RMAN backup.
9.6.4 Where the backup is stored on the TSM server

A query occupancy jamaica_oracle command run from a TSM server administrative client shows where the backup data object was stored on the TSM server for this nodename. This command shows that there is one backup object currently on the TSM server for this node. This is represented by the value for Number of Files. If we were to take another backup, this value would increase. When we delete backups, this value will decrease.
9.6.5 Viewing the backup object in greater detail

Each RMAN backup piece has a corresponding backup data object on the TSM server. This backup object is created by TDP for Oracle based on a NODE_NAME, FILESPACE_NAME, HL_NAME, and LL_NAME:

- The NODE_NAME comes from the client options file.
- The FILESPACE_NAME is either the default value of \adsmorc or the value specified by the variable DSMO_FS.
- The HL_NAME or High Level Name is set by TDP for Oracle, and the value is \orcnt\ This value cannot be changed and is specific to TDP for Oracle for Windows.
- The LL_NAME or Low Level Name comes from the name of the RMAN backup piece.

From a TSM server administrative command line client, you can run a command to view the TDP for Oracle backup data objects in greater detail. The command we typed in was:

```
select * from backups where node_name='JAMAICA_ORACLE'
```

The nodename specified in the quotes is case sensitive and must be in upper-case.

From the output of this command you can see the values for this backup data object.

```
NODE_NAME: JAMAICA_ORACLE
FILESPACE_NAME: \adsmorc
STATE: ACTIVE_VERSION
TYPE: FILE
HL_NAME: \orcnt\nLL_NAME: 0ecl33t7_1_1
OBJECT_ID: 44937
BACKUP_DATE: 2001-03-19 09:15:00.000000
DEACTIVATE_DATE:
OWNER:
CLASS_NAME: DEFAULT
```

9.6.6 Deleting the backup piece from RMAN

To finish the test, we will now delete the backup piece that we created.
From the RMAN prompt, we need to allocate a channel before deleting the backup piece (this is described in the Oracle product documentation in greater detail).

```
RMAN> allocate channel for delete type 'sbt_tape';
RMAN-03022: compiling command: allocate
RMAN-03023: executing command: allocate
RMAN-08030: allocated channel: delete_02
RMAN-08500: channel delete_02: sid=15 devtype=SBT_TAPE
RMAN-08526: channel delete_02: MMS Version 2.1.10.0
```

Next, we change the backup piece to be deleted. The name of the backup piece is obtained from the `list backup;` command:

```
RMAN> change backuppiece '0ecl33t7_1_1' delete;
RMAN-03022: compiling command: change
RMAN-03023: executing command: DELETE
RMAN-08073: deleted backup piece
RMAN-08517: backup piece handle=0ecl33t7_1_1 recid=9 stamp=424775594
```

At this point, the backup piece 0ecl33t7_1_1 has been deleted from RMAN. Since we were running in NOCATALOG mode, the information about this backup in the control file has been deleted. On the TSM server, the backup data object corresponding to this RMAN backup piece has been marked inactive. It will be deleted during the next expiration processing on the TSM server. The select command that was used to view this backup data object can be used to see that the STATE has change from ACTIVE_VERSION to INACTIVE_VERSION.

```
NODE_NAME: JAMAICA_ORACLE
FILESPACE_NAME: \adsmorc
    STATE: INACTIVE_VERSION
    TYPE: FILE
    HL_NAME: \orcnt\0ecl33t7_1_1
    LL_NAME: 0ecl33t7_1_1
    OBJECT_ID: 44937
    BACKUP_DATE: 2001-03-19 09:15:00.000000
    DEACTIVATE_DATE: 2001-03-19 09:15:00.000000
    OWNER:
    CLASS_NAME: DEFAULT
```
Chapter 9. Installation of TSM software on Windows 2000

9.7 Installing the TSM backup-archive client

The TSM backup-archive client is required to perform file level (operating system) backups to a TSM server. It is also required if you wish to use the TSM scheduling service.

Three basic steps are needed before you can perform file level backups to a TSM server.
1. Register a node with the TSM server
2. Install the TSM backup-archive client
3. Configure the client options file (dsm.opt)

9.7.1 Registering a node with the TSM server

You should use a nodename for the backup-archive client that is different from the nodename that is used for TDP for Oracle. The node that will be used for the TSM backup-archive client requires a different storage policy than TDP for Oracle, therefore it should not belong to the same domain as the TDP for Oracle node. We chose the STANDARD domain on our TSM server that is default for all backup-archive clients. We used the following command to register the node.

```
.brazil>register node jamaica jamaica_password domain=standard
ANR2060I Node JAMAICA registered in policy domain STANDARD.
ANR2099I Administrative userid JAMAICA defined for OWNER access to node JAMAICA.
```

We do not plan on manually de-activating backup objects, so we did not specify BACKDELETE=YES. We also do not plan on using multiples sessions to send backups to tape drives, so we did not specify MAXNUMMP.

9.7.2 Installing the TSM backup-archive client

Here we provide instructions for Installing the TSM backup-archive client.

9.7.2.1 Downloading latest TSM backup-archive client and API

The current version of the TSM backup-archive client which contains the TSM API, can be downloaded from the following FTP site.

```
```
From this site, you can select which TSM client version and release. The latest is version 4 release 1. In Figure 34 this is the folder v4r1:

![Figure 34. TSM backup-archive client version and release](image)

After selecting the version and release, you are presented with a list of operating systems to choose from. Select the Windows folder, then select the i386 directory, and you can now select which level of code to download. Figure 35 shows the levels of the Version 4 Release 1 backup-archive client that were available in March of 2001. The folder LATEST always links to the latest level of client code available. Currently the latest level is 'v412', so the v412 folder or the LATEST folder will take you to the same place:

![Figure 35. Levels of TSM backup-archive client code](image)

Figure 36 shows the files available for the latest level of the TSM backup-archive client:

![Figure 36. Files available for latest level of TSM backup-archive client](image)
The file ending readme.ftp (IP22151_12_readme.ftp) contains instructions on how to download and install the client code. The file ending read1stc.txt (IP22151_12_read1stc.txt) is the readme for this level of code. The readme contains information that should be read before installation. Among other things the readme contains: system requirements, warnings, APARS fixed in the PTF, limitations, etc.

We downloaded the file IP22151_12.exe and saved it to the local hard drive of the machine that the DB2 server is running on. The full level information of this package is Version 4 Release 1 Level 2.12.

9.7.2.2 Installation

After downloading the TSM backup-archive client, double-click the package to start the installation process. You will be presented with a screen requesting a temporary location to extract the contents of the package (Figure 37).

![Figure 37. Location to Save Files screen](image)

Chapter 9. Installation of TSM software on Windows 2000
The next screen will prompt you for which Language you would like to use (Figure 38).

![Choose Setup Language](image)

*Figure 38. Language selection*

After selecting the desired language, you will be presented with a welcome screen to InstallShield informing you that this will install the Tivoli Storage Manager Client (Figure 39).

![Tivoli Storage Manager Client - InstallShield Wizard](image)

*Figure 39. Welcome screen of InstallShield*
Select the desired installation directory. We selected the default location of c:\Program Files\Tivoli\TSM\ (Figure 40).

![Figure 40. Installation directory selection](image-url)
Select either Complete or Custom. The option Complete includes the TSM backup-archive client, TSM API Runtime files, and documentation. Complete does not include the TSM API Software Developer Kit nor the TSM Server Administrative Client. If you select Custom, you can specify just the TSM API Run-time files. We chose the Custom option (Figure 41).

![Tivoli Storage Manager Client - InstallShield Wizard](image)

**Set up Type**

Choose the setup type that best suits your needs.

- **Complete**
  - All program features will be installed. (Requires the most disk space.)

- **Custom**
  - Choose which program features you want installed and where they will be installed. Recommended for advanced users.

*Figure 41. Complete or Custom*
When you select Custom, you can select which items to install. We wanted the administrative client for troubleshooting purposes and the API SDK files in order to be able to manually delete Oracle backup objects on the TSM server. Manually deleting backups objects is covered in B.5.2.1, "Using the API sample program to delete objects" on page 207. Our approach is shown in Figure 42.

Figure 42. Selecting the API SDK and Administrative client files
After selecting a Complete or Custom installation, you are presented with a screen where you can confirm your choice, as shown in Figure 43.

![Installation confirmation screen](image)

**Figure 43. Installation confirmation screen**
After the files have been installed, you are presented with a screen informing you of successful completion of the installation (Figure 44).

![Successful installation confirmation](image)

**Figure 44. Successful installation confirmation**

### 9.7.3 Configuring the backup-archive client options file (dsm.opt)

Once the TSM backup-archive client has been installed, you need to configure the client options file. The client options file is a plain text file named dsm.opt that resides in the `<install_dir>\baclient` directory. In our installation, this directory was `c:\program files\Tivoli\TSM\baclient`.

To configure the client options file, you can manually create the dsm.opt file using a text editor. Or you can start the backup-archive client graphical user interface, and when it detects that the dsm.opt does not exist, it will automatically start up a wizard to guide you through the process. We chose to create the file manually.
If you are using TCP/IP as your communication method, the only line you need in your options files is `TCPSERVERADDRESS <ipaddress>`, where `<ipaddress>` is replaced with the dotted decimal address or DNS name of your TSM server. The TSM backup-archive client will use defaults for the rest of the values (for example, COMMMETHOD, TCPPORT, COMMMETHOD, PASSWORDACCESS). For clarity, we specified these options.

We created the client options file by Right-Clicking the folder `c:\program files\Tivoli\TSM\baclient` and selecting `New->Text Document`. We named this file `dsm.opt` (Figure 45).

Figure 45. Creating a new text document
We then edited the dsm.opt client options file using Notepad (Figure 46).

We specified values for COMMETHOD, TCPSERVERADDRESS, TCPPORT, NODENAME, and PASSWORDACCESS.

Our TSM server is on an RS/6000 AIX machine with a TCP/IP address of 193.1.1.11 and is listening for client connections on port 1500. Therefore, we chose TCPIP as our communication method, 193.1.1.11 as our TCP server address, and 1500 as our TCP port.

We used the nodename of JAMAICA that we registered in Section 9.7.1.

We had a choice to set PASSWORDACCESS to PROMPT or to GENERATE. For ease of management, we chose GENERATE.

![Editing the client options file using notepad.exe](Figure 46. Editing the client options file using notepad.exe)

Once the dsm.opt client options file has been saved, we can now use the TSM backup-archive client command line or GUI to perform backups and archives.

In order to use the TSM scheduling, additional steps are still required. For scheduling RMAN backups, see Section 11.2.1, “Scheduling TDP for Oracle backups” on page 172.
9.7.3.1 Coexistence of backup-archive client and TDP for Oracle

When using the incremental command of the backup-archive client, you should exclude the database objects that are backed up using TDP for Oracle. While the database is running, you will not get a consistent backup using the backup-archive client.

In our environment, all the files that were backed up using TDP for Oracle were in the c:\oracle\oradata directory. To prevent them from being backed during a full incremental backup, we added an exclude.dir statement to the dsm.opt file:

```
exclude.dir c:\oracle\oradata
```

To prevent those files from being backed up with a selective command, we also added this statement:

```
xclude c:\oracle\oradata\...\*
```

You may wish to use additional exclude statements to prevent other files from being backed up during your incremental backups. However, you should consider using an include statement to back up Oracle.ORA files (tnsnames.ora, init.ora, etc..) These files would make a disaster recovery restore much easier:

```
include c:\oracle\admin\...\*.ora
include c:\oracle\ora81\...\*.ora
```

After making these changes, our resulting dsm.opt file looked like this:

```
COMMETHOD TCPIP
TCPSERVERADDRESS 193.1.1.11
TCPPORT 1500
NODENAME JAMAICA
PASSWORDACCESS GENERATE
exclude.dir c:\oracle\oradata

include c:\oracle\admin\...\*.ora
include c:\oracle\ora81\...\*.ora
exclude c:\oracle\oradata\...\*
```

With these include/exclude statements, the Oracle database files will not be backed up in the TSM backup-archive client incremental.
Chapter 10. Oracle backup considerations on Windows 2000

RMAN is the backup and recovery tool provided by Oracle for Oracle8 Server databases and is packaged with the Oracle8 Server product. RMAN manages the process of backing up, restoring, and recovering Oracle8 target database instances. RMAN can automatically back up, restore, and recover the following database objects:

- Database
- Tablespaces
- Data files
- Control files
- Archived redo logs

In addition to backup and restore operations, RMAN can:

- Generate log records of all backup and recovery operations
- Run backup and restore operations in parallel to improve performance
- Find database objects that require a backup, on the basis of user-defined values

RMAN provides the interface to the Oracle8 database and the functions for backup, restore, and recovery. It does not provide any storage management capabilities and must be integrated with other storage management products such as TSM to provide a complete enterprise wide storage management solution.

10.1 RMAN: Oracle Recovery Manager

This section presents an overview of the architecture and the components involved in backing up Oracle8 databases with RMAN and TDP for Oracle. It is an introduction to RMAN and in particular how it integrates with TSM. It is not an authoritative guide to RMAN or Oracle8 backup and recovery. This chapter should be read in conjunction with the Oracle8 Server Backup and Recovery Guide.

10.1.1 RMAN system components

RMAN consists of several components that interact during the backup and recovery process. See Figure 47, “RMAN and TSM system components” on page 156.
10.1.1.1 RMAN command
The RMAN command is the administrator’s interface to RMAN. It invokes a
command line interface that provides an operating system independent
scripting language for performing backup and recovery operations. RMAN
can be executed either interactively, where a command prompt is displayed
and additional RMAN commands entered, or in batch mode, where an RMAN
script containing commands is executed.

10.1.1.2 Target database
The target database is the Oracle8 database instance on which RMAN
executes specified backup, restore, and recovery actions. When the RMAN
command is executed, it connects to the target database. The target
database is specified by using RMAN parameters.

10.1.1.3 Communication channel
RMAN can perform backup and restore functions to either local disk or to
external media management products such as a TSM server through the
orabsnt.dll library provided by TDP for Oracle for Windows. These I/O
operations are performed over a communication channel that defines the
device to be used for the operation. The channel is used by RMAN to send or
receive backup data to and from the I/O device.

For backup and restore operations, you must allocate a channel before the
operation is performed. A channel corresponds to a single device. With the
TDP for Oracle for Windows, a channel is a single session to a TSM server.
Multiple channels can be allocated. RMAN provides a multiplexing feature
that enables parallel data streams to be sent over multiple allocated channels
to maximize backup and recovery performance.

10.1.1.4 Recovery catalog
The recovery catalog is the repository for information about backup objects
created by RMAN. It is a Oracle8 database instance, separate from the target
databases, and can contain information for multiple target databases. The
data stored in the recovery catalog comprises structural information about the
target databases to back up and restore. The recovery catalog contains
information about:

- Physical schema of a target database

You have to register the target database at the recovery catalog to define
the physical schema of the target database. RMAN needs to know about
any structural change of the target database and obtains this information
from the target database control file.
• Database backup history

RMAN backs up databases, tablespaces, data files, control files, and archive logs to the TSM server. Details of these backup objects held on TSM is stored in the recovery catalog.

• Backup and recovery history

RMAN stores backup, restore, and recovery information to maintain a history of previously performed operations. When backup and restore operations are performed, this information enables RMAN to determine:

- Database files that require backing up
- Old backup files that can be deleted
- Files that are not recoverable

• Stored RMAN scripts

RMAN commands can be stored in the recovery catalog as stored scripts. Scripts can be created to automate the execution of several RMAN operations.

Note

Oracle strongly recommends that a recovery catalog be used with RMAN. However, it is possible to execute RMAN commands without one. If you operate without a recovery catalog, RMAN uses the target database control file to store backup and structural information about the database. The following limitations apply when operating without recovery catalog:

• Point-in-time recovery is not possible.
• Stored RMAN scripts cannot be used.
• Recovery cannot be performed if the control files are lost or damaged.

Oracle recommends the use of multiplexed control files, with each file located on different disks to protect against media failure.

Refer to Figure 47 for the following discussion.
As shown in Figure 47, the operation is started by invoking the command `rman` and entering the appropriate commands directly or by submitting a command file containing the commands (1).

RMAN connects to the recovery catalog (2) and the target database (3).

Before any backup or restore operations can be performed, RMAN allocates a channel to TSM, using the SBT API (4).

RMAN then creates a server process on the target database instance that performs the operation (5).

For restore operations, RMAN queries the recovery catalog to determine which files to restore from TSM. For a backup operation, RMAN backs up the objects specified in the command to TSM. In both cases, the data is transferred on the previously defined channel.
10.1.2 Create the RMAN catalog

The procedure we used to create the recovery catalog is described in the manual Oracle8i Recovery Manager User's Guide and Reference, A76990-01.

The following are sample print screens showing the creation of the user *rman* and the *create catalog* command:

```
oracle@sol-0 -> sqlplus sys/manager
SQL*Plus: Release 8.1.7.0.0 - Production on Wed Mar 7 11:24:12 2001
(c) Copyright 2000 Oracle Corporation. All rights reserved.

Connected to:
Oracle8i Enterprise Edition Release 8.1.7.0.0 - Production
Jserver Release 8.1.7.0.0 - Production
SQL> create user rman identified by rman temporary tablespace TEMP default tablespace USERS quota unlimited on USERS;
User created.
SQL> grant connect, resource to rman;
Grant succeeded.
SQL> grant recovery_catalog_owner to rman;
Grant succeeded.
SQL> 
```

```
oracle@sol-0 => rman catalog rman/rmanORIO
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-06008: connected to recovery catalog database
RMAN-06428: recovery catalog is not installed
RMAN> create catalog tablespace users;
RMAN-06431: recovery catalog created
```
10.1.3 Registering the target database

Having created a recovery catalog, you must register the target databases to it. Use RMAN commands. Registration must be done before any other RMAN functions can be performed on the target databases.

You register each of the target database instances that will use the recovery catalog. As the Oracle user run the `rman` command to connect to the target and recovery catalog databases and then run the `register database` command.

The following example shows the commands to connect to the BEACH target database as the internal user and the RIO recovery catalog database as the newly created `rman` user, followed by the target database being registered

```
C:\oracle\ora81\network\ADMIN>rman target internal/manager@beach rcvcat "rman/rman@rio"
RMAN-06005: connected to target database: BEACH (DBID=24844426)
RMAN-06008: connected to recovery catalog database
RMAN> register database;
RMAN-03022: compiling command: register
RMAN-03023: executing command: register
RMAN-08006: database registered in recovery catalog
RMAN-03023: executing command: full resync
RMAN-08002: starting full resync of recovery catalog
RMAN-08004: full resync complete
RMAN>
```

To list the result of the register command, use `report schema`. 

158 Backing Up Oracle using Tivoli Storage Management
10.2 Back up the database using TDP for Oracle and RMAN

Performing a backup with RMAN creates a backup set for that operation. A backup set contains backups of one or more data files or archive logs. A data file backup sets can also include a control file backup. Archive log backup sets can only include archive logs. Data file backup sets can be incremental or full and do not include empty blocks. A backup set is a logical entity composed of one or more physical output files called backup pieces. Each backup piece contains control and checksum information that allows the Oracle server process to validate the backup piece during a restore. A backup set is created by the `backup` command. A `restore` command is required to extract files from a backup set.

10.2.1 Full backup

A full backup is a non-incremental backup of one or more data files. A full backup has no effect on incremental backups and is not considered to be part of the incremental strategy.

If the database is in ARCHIVELOG mode, you can choose to do full backup while the database is online or offline. If the database is in NOARCHIVELOG mode, the database must be closed by a clean shutdown. Full backups can be taken of:

- Data files
- Tablespaces
- Databases
- Control files
- Archive logs
10.2.2 Whole database backup

A whole database backup set contains the control files and all database files that belong to that database. Whole database backups do not require the database to be operated in a specific archiving mode. They can be taken whether a database is operating in ARCHIVELOG or NOARCHIVELOG mode. If the database is in ARCHIVELOG mode, you can choose to back up the database while it is open or closed. If running in NOARCHIVELOG mode, the database must be shut down first. There are two types of whole database backups:

- Consistent whole database backup

  A consistent whole database backup is a backup set where all files within it are consistent to the same point in time. A consistent whole database is the only valid backup for databases running in NOARCHIVELOG mode. The only way to take a consistent whole database backup is to shut down the database cleanly and take a backup while the database is offline.

- Inconsistent whole database backup

  An inconsistent whole database backup is a backup of an online database. It is inconsistent because portions of the databases may have been modified and written to disk during the backup process. The database must be in ARCHIVELOG mode in order to run an inconsistent backup.

  After an inconsistent backup is performed, the archived and online redo logs should also be backed up. Inconsistent whole database backups are restored and made consistent by applying any subsequent incremental backups and redo logs, online and archive, during the recovery process.

10.2.3 Incremental backup

RMAN provides the capability of incrementally backing up databases at the individual block level. An incremental backup is a backup of one or more data files and contains only those blocks that have been modified since a previous backup at the same or lower level.

The *multilevel incremental backup* feature allows you to create different levels of incremental backups. Each level is denoted by an integer, with 0 being the lowest backup level. An incremental backup performed at a given level backs up only those blocks that have been modified since the last backup at the same or lower level. An incremental backup can be performed on:

- Individual data files
- Tablespaces
- The entire database
Incremental backup of control files or archived logs is not supported. There are two types of incremental backups: non-cumulative and cumulative.

10.2.3.1 Non-cumulative incremental backup
A non-cumulative incremental backup backs up only those blocks that have changed since the previous incremental backup at the same or lower level. This is the default mode of operation for incremental backups.

A level 0 backup backs up all blocks that contain data. It performs the same backup as a full backup. A level 0 backup is required for subsequent incremental backups at other levels. An incremental backup at a level greater than level 0 backs up only those blocks that have changed since a previous incremental backup at the same or lower level. The size of the backup depends on the number of blocks modified.

Figure 48 illustrates part of a monthly cycle of non-cumulative incremental backups. The cycle is based on backup levels 0, 1, and 2. A weekly backup is performed at level 0 on Sunday, incremental backups level 2 are performed on Monday to Wednesday and on Friday and Saturday, and weekly incremental backups at level 1 on each Thursday.

10.2.3.2 Cumulative incremental backup
A cumulative incremental backup at level \( n \) contains only blocks that have been changed since the most recent backup at level \( n - 1 \) or lower.
Cumulative backups require more storage space than differential backups, but they are preferable during a restore operation because only one backup for a given level is needed. See Figure 49.

Note that the first incremental backup must be a level 0 backup that contains all used blocks. A cumulative backup at level 2 will contain all blocks changed since the most recent level 1 backup, copying all blocks changed since the base level 0 backup only if a previous level 1 is unavailable. In contrast to a cumulative backup, a differential backup at level 2 will determine which level 1 or level 2 backup occurred most recently and copies all blocks changed since that backup.

A cumulative incremental backup is a backup of one or more data files that contains only those blocks that have changed since a previous backup at a lower level. A cumulative backup is run by specifying the **cumulative** option when defining the backup job.

### 10.2.4 Image copies

An image copy is a single file (data file, archive log, or control file) that can be used as-is to perform a recovery. It is similar to an operating system copy of a single file except that it is produced by an Oracle server process which performs additional tasks such as validating the blocks in the file and registering the copy in the control files. An image copy can be done only to disk.
Image copies are not discussed further in this chapter, because RMAN does not send image copies to TSM — they are always stored locally on disk.

10.3 Backup examples

This section describes some typical backup examples using RMAN, complete with sample scripts. The examples are based on the following database instances:

- **Target database**: paris
- **Recovery catalog database**: rio

For the examples, we used RMAN command files and redirected the output to logfiles. RMAN command files are regular text files containing commands executed by RMAN.

The examples illustrate how to perform the following functions:

- Whole consistent database backup
- Archive redo log backup
- Incremental level 0 backup

10.3.1 Consistent whole database backup

In our example, we are using RMAN to shut down and restart the database to guarantee that all read-write datafiles and control files have been checkpointed with respect to the same SCN.

This example is suitable for databases operating in ARCHIVELOG and NOARCHIVELOG mode and is in our opinion the most common backup operation.

On our environment we created a folder called scripts in the c:\oracle\admin directory. In this folder we store our RMAN scripts. Before running the RMAN scripts we first change to this directory and also set ORACLE_SID=BEACH.

```
C:\oracle\admin\scripts>rman target internal/manager@beach rcvcat rman/rman@rio
  cmdfile=backup_offline_beach log=backup_offline_beach.log
```

*Figure 50. Whole consistent backup, RMAN command*
To guarantee that a database’s datafiles are consistent, shut down the database with the NORMAL, IMMEDIATE or TRANSACTIONAL options before making a whole database backup. To bring up the database in the desired state, we chose the sequence:

- shutdown immediate;
- startup force dba;
- shutdown immediate;
- startup mount;

```bash
run {
    allocate channel t1 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=256)';
    allocate channel t2 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=256)';
    set maxcorrupt for datafile 1 to 0;
    backup full
        filesperset 3
        format '%d/%t/%s/%p'
        tag 'paris offline 20010316.1111' (database);
    release channel t1;
    release channel t2;
    alter database open;
}
```

**Note:** Unlike TDP for Oracle for UNIX, we do not need to specify as many channel definitions. The only one that is required is DSMO_AVG_SIZE, the rest are left as defaults.

Channel definitions:

- DSMO_AVG_SIZE is a size estimate used by the TSM server to reserve space on TSM storage
- The allocation of multiple channels is only possible when TSM node’s MAXNUMMP (maximum mount points allowed) is greater than or equal to the number of channels.
The following is an explanation of various statements in our coding example:

- `set maxcorrupt for datafile 1 to 0` causes Oracle to abort the backup process if RMAN detects corrupt blocks in the SYSTEM tablespace. In our database, we used AUTOEXTEND=ON for all datafiles. Datafile 1 is always used for the SYSTEM tablespace.

- `backup full` causes a whole database backup.

- `filesperset 3` forces RMAN to include maximal 3 files in a backup set. A possible restore from a single datafile will be faster processed.

- `format '%d/t/s%p'` creates a TSM low level name including database name/timestamp/backup set number/backup piece number.

- `tag 'beach offline Monday Full'` is a user defined symbolic name which is assigned to backup sets. It makes it easier to address these backups in RMAN restore or change commands.

- `alter database open` finally makes the database accessible to users.

### 10.3.2 Archived redo log backup

This example shows how to back up the archived redo logs. Before you back up the archived redo logs, Oracle recommends that you archive the current online redo log file by using the alter system archive current log command.

```
C:\oracle\admin\scripts>rman target internal/manager@beach rcvcat rman/rman@ri
cmdfile=backup_archlog_beach log=backup_archlog_beach.log
sql 'alter system archive log current' ;
run {
allocate channel t1 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=1)';
allocate channel t2 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=1)';
backup filesperset 20
  format '%d/LOGS/\t%s/%p'
  archivelog all
  delete input ;
release channel t1 ;
release channel t2 ;
}
```

The next example shows how to back up archived redo logs from one archivelog destination.
The first statement `alter system archive log current` forces Oracle to perform a log switch and to include the present log in the backup set.

If the database is running in ARCHIVELOG MODE and it is not possible to perform consistent backups, then the backup of the archived redo logs should be processed immediately after the database backup. But there are other reasons too to perform the backup of the archived redo logs frequently:

- If the file system containing the LOG_ARCHIVE_DESTn has no space left, Oracle is no more able to archive the logs - and waits until the archiver process is able to archive again
- When a disk fails, it is still possible to recover without (or only a minimum) of data loss; even when the database backup is already some hours or days old

### 10.3.3 Incremental level 0 backup

This example shows how to perform an incremental backup of the database with RMAN. The level 0 backup is the base for further incremental backups with a higher level:

```sql
run {
    allocate channel t1 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=265)';
    allocate channel t2 type 'sbt_tape' parms 'ENV=(DSMO_AVG_SIZE=265)';
    backup incremental level 0
    setsize = 265000
    format '%d/%t/%s/%p'
    tag 'beach inc lvl0 weekly'
    database ;
    release channel t1 ;
    release channel t2 ;
}
```

In this example, `setsize = 265000` specifies the maximum size of a backup set. It must be greater or equal to the largest datafile in the backup. That is another option to control the composition of the datafiles in a backup set.
10.4 File level backup using TSM

You can use the TSM backup-archive client to archive Oracle database files. Oracle calls this an operating system or file level backup. You should use the archive feature of the backup-archive client so that all the files are expired at the same time (you do not want versioning).

In order to do this, you must first shut down the database and make sure it is in a consistent state. To do this, we use the Oracle RMAN commands in this order:

1. shutdown immediate;
2. startup force dba;
3. shutdown immediate;

With the database shut down, we can now perform an archive of the database files (Figure 52).

![Figure 52. Backup-archive client GUI](image)
After selecting the **Archive** button, we browsed to c:\oracle\oradata and selected it for archive (Figure 53).

![Figure 53. Selecting the database files and starting the archive](image)

After selecting the database files and clicking the **Archive** button, the TSM backup-archive client archived all of these files. If you did not properly shut down the Oracle database, some of the files would fail the Archive with File In Use errors.
To verify that the archive was successful, we can use the **Retrieve** button to view that something was backed up (Figure 54).
Chapter 11. Day-to-day monitoring on Windows 2000

Once the TDP for Oracle implementation and environment has been set up, as described in the preceding chapters, it is recommended that the regular day-to-day tasks be monitored and automated for more efficient execution. This chapter details some of the processes and tasks which can be used.

11.1 Automating database tasks

There are some Oracle database tasks which can be automated as detailed below. These are best determined by the Oracle DBA.

11.1.1 Starting SQL*Net and RDBMS services at system reboot

During the Oracle installation process, the Oracle services are set to start automatically. If they are not starting automatically, check the services to see what the startup mode is and modify it appropriately.

11.2 Automating RMAN backups

In order to automate Oracle RMAN backups, you use the RMAN cmdfile option. When you specify cmdfile when invoking RMAN, then RMAN executes the file specified line by line. The file should be plain text. In Section 10.3.1, “Consistent whole database backup” on page 163 we discussed the creation of the scripts directory to hold our RMAN scripts. Also in that section, we showed how to run the script backup_offline_beach using the cmdfile option.

Instead of typing in the SET ORACLE_SID and RMAN commands each time, we created a simple command file called AUTORMAN.CMD, that uses simple variable substitution to call the RMAN script and create the log file. This file is based on a database SID of BEACH, where the internal password of the BEACH database is manager, and the user/password and database of the recovery catalog is rman/rman@rio, a directory c:\oracle\admin\scripts that contains the RMAN scripts. You will need to modify all of these values to correspond with your environment.

```
set ORACLE_SID=BEACH
rman target internal/manager@beach rcvcat rman/rman@rio
cmdfile=c:\oracle\admin\scripts\%1 log=c:\oracle\admin\scripts\%1.log
```
To use this command file with the script named `backup_offline_beach`, we just add the name of the script when calling the AUTORMAN.CMD file.

```
autorman backup_offline_beach
```

Windows will then replace each occurrence of `%1` in the AUTORMAN.CMD file with `backup_offline_beach`. Here is an example of this:

```
C:\oracle\admin\scripts>autorman backup_offline_beach
C:\oracle\admin\scripts>set ORACLE_SID=BEACH
C:\oracle\admin\scripts>rman target internal/manager@beach
rcvcat rman/rman@rio cmdfile=c:\oracle\admin\scripts\backup_offline_beach
log=c:\oracle\admin\scripts\backup_offline_beach.log
```

Using this AUTORMAN.CMD file, we can now easily schedule RMAN scripts using TSM scheduling without creating a separate command file for each backup.

### 11.2.1 Scheduling TDP for Oracle backups

In order to use the TSM scheduling services, you must perform configuration on both the TSM server and the TSM client.

#### 11.2.1.1 TSM server steps for scheduling

On the TSM server you need to perform two commands:

1. **DEFINE SCHEDULE**

   With this command, you must specify the domain where the TDP for Oracle backup node belongs, `ACTION=COMMAND`, and `OBJECT=<path to backup command>`. The other options regarding time and frequency can be tailored to your environment. See the TSM server product documentation for details, or type `HELP DEFINE SCHEDULE` at the TSM administrative prompt.

   **Note**

   Do NOT define the schedule with `ACTION=INCREMENTAL`. This would cause a full incremental backup of the entire machine using the backup-archive client, but would do it using the TDP for Oracle nodename.

   This will show how to schedule the `c:\oracle\admin\scripts\autorman.cmd` command that we discussed in the preceding section.
2. DEFINE ASSOCIATION

Now we associate this schedule with the node that will execute the command. Our Windows DB2 backups use JAMAICA_ORACLE as the nodename, so that is what we used for this command:

```bash
tsm: BRAZIL>define association api_domain backup_offline jamaica_oracle
ANR2510I Node JAMAICA_ORACLE associated with schedule BACKUP_OFFLINE in policy domain API_DOMAIN.
```

### 11.2.1.2 TSM client steps for scheduling

Once the schedule has been defined on the TSM server, and the schedule has been associated with the nodename, the only thing left to do is set up a scheduler service.

We want to use a different scheduler service for the TDP for Oracle backups to make it easier to check whether the commands executed. In our Windows environment, we chose scheduling mode client polling.

If you have more than one scheduler service on a machine and you are using server prompted scheduling mode, you MUST specify the TCPCLIENTPORT option in one of the scheduler service’s to be different from the default. Otherwise, both services will try to listen on the same port.

The TSM backup archive client provides a wizard to assist with installing a scheduler service. We chose to use the command line utility `dsmcutil.exe` located in the `...aclient` directory. In order to keep this service separate from the backup client scheduler service, you must specify options for NAME, NODE, PASSWORD, OPTFILE, SCHEDLOG, and ERROR.LOG.

If you do not use different values for SCHEDLOG and ERROR.LOG, both scheduler services will write to the same log files:

```bash
C:\Program Files\Tivoli\TSM\baclient>dsmcutil install /name:"TSM Scheduler Oracle" /node:"jamaica_oracle" /password:"jamaica_oracle" /optfile:"c:\progra~1\tivoli\tsm\agentoba\dsm.opt" /schedlog:"c:\progra~1\tivoli\tsm\agentoba\dsmsched.log" /errorlog:"c:\progra~1\tivoli\tsm\agentoba\dsmerror.log"
```
After running that command, a number of messages are sent to the console. This will let you know if there are any errors. The most important ones to see are the following:

The service was successfully installed.
Starting the 'TSM Scheduler Oracle' service ..... 
The service was successfully started.

If you make a mistake, you can easily remove the scheduler service by specifying `dsmcutil remove /name:"Name of Scheduler Service"`.

You can check which scheduler services are installed with `dsmcutil list`:

```
C:\Program Files\Tivoli\TSM\baclient>dsmcutil list
Installed TSM Client Services:
1. TSM BAClient Scheduler
2. TSM Scheduler Oracle
```

To verify the settings and configuration of our scheduler, we used the command, `dsmcutil query /name:"Name of Scheduler Service"`.

```
C:\Program Files\Tivoli\TSM\baclient>dsmcutil query /name:"TSM Scheduler Oracle"

Service Name : TSM Scheduler Oracle
Logon Account : LocalSystem
Start Type : Demand
Current Status : Started
TSM Client Service Registry Settings:

Client Service Type = Client Scheduler Service
Client Directory = "C:\Program Files\Tivoli\TSM\baclient\dsmcsvc.exe"
Options file = c:\progra~1\tivoli\tsm\agentoba\dsm.opt
Event Logging = YES
TSM Client Node = JAMAICA_ORACLE
Comm Protocol = (value not currently set)
Server Port = (value not currently set)
Schedule Log = c:\progra~1\tivoli\tsm\agentoba\dsmsched.log
Error Log = c:\progra~1\tivoli\tsm\agentoba\dsmerror.log
MSCS Enabled Node = (value not currently set)
Cluster name = (value not currently set)
```
Upon successful installation of the scheduler service, the scheduler service will automatically start and connect to the TSM server to retrieve any associated schedules. The dsmsched.log for the scheduler service will show whether a schedule was retrieved from the TSM server. Our dsmsched.log shows that the c:\oracle\admin\scripts\autorman backup_offline_beach will be executed in 2 minutes.

We checked the dsmsched.log and it said that the command failed. We checked the backup_offline_beach.log file and it gave us this message:

```plaintext
RMAN-00554: initialization of internal recovery manager package failed
RMAN-04005: error from target database: ORA-01017:
   invalid username/password; logon denied
```

That message would lead you to believe that the password in the AUTORMAN.CMD file was incorrect. However, we had previously run it manually and it was successful. The most common cause of a scheduled action failing when the manual action works in Windows is that the scheduler service is running as an account that does not have sufficient rights. We stopped the scheduler service:

```plaintext
net stop "TSM Scheduler Oracle"
```

Then, using Administrative Tools -> Services, we changed the logon type for the scheduler service from system account to the Administrator account. You should be able to use any account that is successful in running the command manually.
We then updated the TSM schedule backup_offline_beach so that it would run again:

    update schedule api_domain backup_offline_beach starttime=now

Then we started the scheduler service:

    net start "TSM Scheduler Oracle"

Once again, the dsmsched.log said that the command would execute in 2 minutes.

So, 2 minutes later, we checked the dsmsched.log again and saw that this time it successfully executed the command.
11.3 RMAN report, list, and crosscheck utilities

RMAN can generate reports about backups and images copies with the report, list and crosscheck commands.

The report command can produce the following types of information from the recovery catalog:

- The database schema
- Which data files require a backup
- Backup files that are obsolete and can be deleted
- Data files that are not recoverable

The list command queries the recovery catalog and produces information such as:

- List of backup sets containing a backup of a specific data file
- List of backup sets containing backups for a specific tablespace
- List of backup sets containing backups for the whole database

The crosscheck command queries the recovery catalog and verifies, whether backup sets marked available or expired are available.

- For objects stored on disk (type disk): the command determines whether the header of the backup piece is valid
- For TSM objects (type sbt_tape): the command simply check that the backups exist

You would want to use this utility regularly to verify if your backup is still available.

Below are examples of how you can use these utilities to perform day-to-day monitoring. For the complete syntax of these utilities, please refer to the Oracle 8i Recovery Manager User's Guide and Reference, A76990-01.
11.3.1 Report utility

The report utility provides analysis of your backup and recovery situation. Below are examples of the report utility.

11.3.1.1 Report target database schema

You can report the structure of the database using the REPORT SCHEMA command.

- This example shows an overview of the target database structure using the REPORT SCHEMA command:

```
RMAN> report schema;
RMAN-03022: compiling command: report
Report of database schema
File K-bytes Tablespace RB segs Name
----- ---------- ------------------ ------- -------------------
1 280576 SYSTEM YES C:\ORACLE\ORADATA\BEACH\SYSTEM01.DBF
2 51200 RBS YES C:\ORACLE\ORADATA\BEACH\RBS01.DBF
3 20480 USERS NO C:\ORACLE\ORADATA\BEACH\USERS01.DBF
4 20480 TEMP NO C:\ORACLE\ORADATA\BEACH\TEMP01.DBF
5 10240 TOOLS NO C:\ORACLE\ORADATA\BEACH\TOOLS01.DBF
6 20480 INDX NO C:\ORACLE\ORADATA\BEACH\INDX01.DBF
7 20480 DRSYS NO C:\ORACLE\ORADATA\BEACH\DR01.DBF
8 30728 OEM_REPOSITORY NO C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA
```

- You can also use the REPORT SCHEMA to show the structure of the database at a prior time. For example, if you want to know the structure of the database 2 weeks ago, you can issue the following:

```
RMAN> report schema at time 'sysdate-14';
RMAN-03022: compiling command: report
Report of database schema
File K-bytes Tablespace RB segs Name
----- ---------- ------------------ ------- -------------------
1 280576 SYSTEM YES C:\ORACLE\ORADATA\BEACH\SYSTEM01.DBF
2 51200 RBS YES C:\ORACLE\ORADATA\BEACH\RBS01.DBF
3 20480 USERS NO C:\ORACLE\ORADATA\BEACH\USERS01.DBF
4 20480 TEMP NO C:\ORACLE\ORADATA\BEACH\TEMP01.DBF
5 10240 TOOLS NO C:\ORACLE\ORADATA\BEACH\TOOLS01.DBF
6 20480 INDX NO C:\ORACLE\ORADATA\BEACH\INDX01.DBF
7 20480 DRSYS NO C:\ORACLE\ORADATA\BEACH\DR01.DBF
8 30728 OEM_REPOSITORY NO C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA
```

11.3.1.2 Report on data files needing backup

You can report on data files that require backup based on three options:
You can report on data files needing backup based on the number of days of redo log activities. This example reports on data files whose recovery requires more than 3 days of archive logs:

```
RMAN> report need backup days 3;
RMAN-03022: compiling command: report
Report of files whose recovery needs more than 3 days of archived logs
File Days Name
---- ----- -----------------------------------------------------
 1  2  C:\ORACLE\ORADATA\BEACH\SYSTEM01.DBF
 2  2  C:\ORACLE\ORADATA\BEACH\RBS01.DBF
 3  2  C:\ORACLE\ORADATA\BEACH\USERS01.DBF
 4  2  C:\ORACLE\ORADATA\BEACH\TEMP01.DBF
 5  2  C:\ORACLE\ORADATA\BEACH\TOOLS01.DBF
 6  2  C:\ORACLE\ORADATA\BEACH\INDX01.DBF
 7  2  C:\ORACLE\ORADATA\BEACH\DR01.DBF
 8  2  C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA
```

You can report on data files needing backup based on the number of incremental backups. The example shows how to use the command to report on data files whose recovery requires more than 3 days of incremental:

```
RMAN> report need backup incremental 1;
RMAN-03022: compiling command: report
Report of files that need more than 1 incrementals during recovery
File Incrementals Name
---- ------------ ----------------------------------------------
 1  2  C:\ORACLE\ORADATA\BEACH\SYSTEM01.DBF
 2  2  C:\ORACLE\ORADATA\BEACH\RBS01.DBF
 3  2  C:\ORACLE\ORADATA\BEACH\USERS01.DBF
 4  2  C:\ORACLE\ORADATA\BEACH\TEMP01.DBF
 5  2  C:\ORACLE\ORADATA\BEACH\TOOLS01.DBF
 6  2  C:\ORACLE\ORADATA\BEACH\INDX01.DBF
 7  2  C:\ORACLE\ORADATA\BEACH\DR01.DBF
 8  2  C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA
```

You can report on data files needing backup based on the number of redundant backups. The following example reports on data files having less than 3 redundant backups:
11.3.1.3 Report on obsolete backups

You can report on obsolete backups in order to determine which backups can be deleted.

- This example shows that any backup having more than four copies is considered obsolete.

- This example shows backups that are obsolete because there is at least one backup and they were made no more than 3 days ago:

- This example shows orphaned backups that are unusable because they belong to incarnations of the database that are not direct ancestors of the current incarnation:
11.3.1.4 Report on unrecoverable data files
To make sure that all data files are recoverable, use the REPORT UNRECOVERABLE command to report on data files that need backup because of unlogged changes that were made after the last backup:

```
RMAN> report unrecoverable device type 'sbt_tape';
```

11.3.2 List utility
The list utility queries the contents of the recovery catalog or the control file primarily to determine which backup or copies are available.

11.3.2.1 List database incarnations
The example below lists incarnations of databases that are cataloged within the recovery catalog. Every time RESETLOGS is performed, an incarnation is created for the database:

```
RMAN> list incarnation;
```

<table>
<thead>
<tr>
<th>DB Key</th>
<th>Inc Key</th>
<th>DB Name</th>
<th>DB ID</th>
<th>CUR</th>
<th>SCN</th>
<th>Reset Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>640</td>
<td>641</td>
<td>BEACH</td>
<td>24844426</td>
<td>YES</td>
<td>282236</td>
<td>12-MAR-01</td>
</tr>
<tr>
<td>2060</td>
<td>2070</td>
<td>UNKNOWN</td>
<td>948509196</td>
<td>NO</td>
<td>1</td>
<td>09-MAR-01</td>
</tr>
<tr>
<td>2060</td>
<td>2061</td>
<td>BERLIN</td>
<td>948509196</td>
<td>NO</td>
<td>264394</td>
<td>23-MAR-01</td>
</tr>
<tr>
<td>2060</td>
<td>2117</td>
<td>BERLIN</td>
<td>948509196</td>
<td>NO</td>
<td>264583</td>
<td>26-MAR-01</td>
</tr>
<tr>
<td>2060</td>
<td>2128</td>
<td>BERLIN</td>
<td>948509196</td>
<td>YES</td>
<td>264583</td>
<td>26-MAR-01</td>
</tr>
<tr>
<td>2060</td>
<td>2103</td>
<td>BERLIN</td>
<td>948509196</td>
<td>NO</td>
<td>264633</td>
<td>26-MAR-01</td>
</tr>
<tr>
<td>1654</td>
<td>1655</td>
<td>OSLO</td>
<td>2939852137</td>
<td>NO</td>
<td>1</td>
<td>09-MAR-01</td>
</tr>
<tr>
<td>1654</td>
<td>2142</td>
<td>OSLO</td>
<td>2939852137</td>
<td>YES</td>
<td>264394</td>
<td>26-MAR-01</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>PARIS</td>
<td>3920710371</td>
<td>YES</td>
<td>1</td>
<td>05-MAR-01</td>
</tr>
</tbody>
</table>

11.3.2.2 List old backups
You can list old backups to determine obsolete backups. The example below lists backups that are older than 5 days:

```
RMAN> list backup completed before 'sysdate-5';
RMAN-03022: compiling command: list
List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- -------- ---------- ---------- -- ---------- ---------- ----------------------
3159 23 425998026 0 425997944 28 02-APR-01

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- ----------- ---------------------- ------------------------
3162 1 1 AVAILABLE 02-APR-01 BEACH/425997944/28/1

List of Datafiles Included
File Name LV Type Ckp SCN Ckp Time
---- ------------------------------------- -- ---- ---------- -------------
3 C:\ORACLE\ORADATA\BEACH\USERS01.DBF 0 Full 612794 02-APR-01
6 C:\ORACLE\ORADATA\BEACH\INDEX01.DBF 0 Full 612794 02-APR-01
8 C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA 0 Full 612794 02-APR-01

RMAN> list backup completed before 'sysdate-5';
RMAN-03022: compiling command: list
List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- -------- ---------- ---------- -- ---------- ---------- ----------------------
3160 24 425998034 0 425997944 29 02-APR-01

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- ----------- ---------------------- ------------------------
3163 1 1 AVAILABLE 02-APR-01 BEACH/425997944/29/1

List of Datafiles Included
File Name LV Type Ckp SCN Ckp Time
---- ------------------------------------- -- ---- ---------- -------------
2 C:\ORACLE\ORADATA\BEACH\RBS01.DBF 0 Full 612794 02-APR-01
4 C:\ORACLE\ORADATA\BEACH\TEMP01.DBF 0 Full 612794 02-APR-01
7 C:\ORACLE\ORADATA\BEACH\DR01.DBF 0 Full 612794 02-APR-01
```

11.3.2.3 List backup of certain objects
By default, the list utility lists backups made for the target database. You can restrict the list to specific objects. The following are examples:

- To find out if there was a backup of the controlfile within the last 24 hours:
To find out if you can recover to a point in time for tablespace USERS before March 22, 2001:

```
RMAN> list backup of controlfile completed after 'sysdate-1';
RMAN-03022: compiling command: list
List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- ------- -------- -------------- ---------- --------- ----------------------
3161 25 425998150 0 425998030 30 02-APR-01

List of Backup Pieces
Key Pc# Cpl# Status Completion Time Piece Name
------- --- --- -------------- ---------------------- ------------------------
3164 1 1 AVAILABLE 02-APR-01 BEACH/425998030/30/1

Controlfile Included
Ckp SCN Ckp time
612794 02-APR-01
```

```
RMAN> list backup of tablespace 'USERS' completed before '22-MAR-01';
RMAN-03022: compiling command: list
RMAN>
RMAN> list backup of datafile 'C:\ORACLE\ORADATA\BEACH\DR01.DBF';
RMAN-03022: compiling command: list
List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- ------- -------- -------------- ---------- --------- ----------------------
3160 24 425998034 0 425997944 29 02-APR-01

List of Backup Pieces
Key Pc# Cpl# Status Completion Time Piece Name
------- --- --- -------------- ---------------------- ------------------------
3163 1 1 AVAILABLE 02-APR-01 BEACH/425997944/29/1

List of Datafiles Included
File Name LV Type Cpl SCN Cpl Time
---- --------------------- --- ----------- ----------
7 C:\ORACLE\ORADATA\BEACH\DR01.DBF 0 Full 612794 02-APR-01
```

To list all backup of datafile '/opt/app/oracle/oradata/paris/system01.dbf' that has been backed up to TSM:
To list backups of archive logs between SCN 633200 until SCN 633300:

```
RMAN> list backup of archivelog from scn 633200 until scn 633300;
RMAN-03022: compiling command: list

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- -------- --------- --------- ----------- ----------------------
3206 29 426005887 0 426005876 35 02-APR-01

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- -------- ---------------------- ------------------------
3208 1 1 AVAILABLE 02-APR-01 BEACH/LOGS/426005876/35/1

List of Archived Logs Included
Thrd Seq Low SCN Next SCN Low Time Next Time
---- ------ --------- --------- -------------- --------------
1 54 633190 633209 02-APR-01 02-APR-01
```

11.3.3 Crosscheck utility

The crosscheck utility verifies whether backups still exist on disk or TSM. RMAN does not delete backup entries that it could not find, but instead marks them as expired. If the backup was erroneously marked expired because for example, TSM was unavailable or misconfigured, crosscheck will mark it available the next time it is run if the backup still exists.

Like the list utility discussed in Section 11.3.2.3, “List backup of certain objects” on page 182, you can crosscheck certain objects. See the Oracle 8i Recovery Manager User Guide and Reference (A76990-1), for the complete syntax of the crosscheck utility.

Here is a sample script to crosscheck all backup entries in the recovery catalog or control file:

```
allocate channel for maintenance type 'sbt_tape';
crosscheck backup;
exit;
```

If you find that the crosscheck expires any backup pieces, you can then run the command delete expired backup; to remove the entries from RMAN.
11.4 Automatic deletion of old backups

Automating deletion of TDP for Oracle for Windows should be planned and tested carefully before implementation in a production environment. There is no simple command that can be run to easily deactivate backups based on redundancy or time. One way to get around this is to use command files for automation. Another way is to use tags as part of your backup process and use the same tags for deletion.

11.4.1 Command files to automate backup deletion

For this example to work, you must create a directory c:\oracle\admin\scripts. In this directory you need to create two files: autodelete.cmd and delete.sql, using a text editor. If you modify this directory path, you must modify the contents of both files. We needed to modify every occurrence of BEACH to match our database SID. You will need to modify manager in the occurrences of internal/manager to be the correct password for your database. And you must modify rman/rman@rio to match the username, password, and database of your recovery catalog.

The command file autodelete.cmd does three things, sets ORACLE_SID, calls sqlplus to execute the delete.sql file, and calls RMAN using the delete_backup.rman file created by sqlplus.

```
set ORACLE_SID=BEACH
sqlplus internal/manager@beach c:\oracle\admin\scripts\delete
rman target internal/manager@beach rcvcat rman/rman@rio cmdfile=c:\oracle\admin\scripts\delete_backup.rman
```

Figure 56. c:\oracle\admin\scripts\autodelete.cmd

The delete.sql creates a file c:\oracle\admin\scripts\delete_backup.rman which logs on to the Oracle database and queries the v$backup_piece for all backup pieces that are not deleted(deleted=’NO’) and were backed up 3 days ago (completion_time < SYSDATE - 3). You can modify the where statement to match any backup pieces that you want.
11.4.2 Using tags to help with deletion

Tags can be used to help automate the deletion of backups. In this example we are doing weekly backups on Sunday, and daily backups during the week. For the daily backups, we set the tag to be ‘Beach Daily’.

```sql
col handle for a30
set echo off
set verify off
set feedback off
set pagesize 0
spool c:\oracle\admin\scripts\delete.sql
select 'allocate channel for delete type ''SBT_TAPE' ;' from dual;
select 'change backuppiece ''\|\|handle\|\|'' delete,' from v$backup_piece where completion_time < SYSDATE - 3 and deleted = 'NO' ;
select 'release channel;' from dual;
spool off;
exit;
```

```sql
run {
allocate channel t1 type 'sbt_tape' parms
'\ENV=( DSMO_AVG_SIZE=256)';
allocation channel t2 type 'sbt_tape' parms
'\ENV=( DSMO_AVG_SIZE=256)';
set maxcorrupt for datafile 1 to 0 ;
backup
  full
  fileserset 3
  format '%d/%t/%s/%p'
tag 'Beach Daily'
  (database);
release channel t1 ;
release channel t2 ;
}
```
We did a couple of backups using that tag; now we can list backups using that tag qualifier:

```
RMAN> list backup tag = 'Beach Daily';
RMAN-03022: compiling command: list

List of Backup Sets
Key Recid Stamp LV Set Stamp Set Count Completion Time
------- ---------- ---------- -- ---------- ---------- ----------------------
3375 46 426020121 0 426020065 51 02-APR-01

List of Backup Pieces
Key Pc# Cp# Status Completion Time Piece Name
------- --- --- ----------- ---------------------- ----------------------
3378 1 1 AVAILABLE 02-APR-01 BEACH/426020065/51/1

List of Datafiles Included
File Name LV Type Ckp SCN Ckp Time
---- ------------------------------------- -- ---- ---------- ------------
3 C:\ORACLE\ORADATA\BEACH\USERS01.DBF 0 Full 653580 02-APR-01
6 C:\ORACLE\ORADATA\BEACH\INDX01.DBF 0 Full 653580 02-APR-01
8 C:\ORACLE\ORADATA\BEACH\OEM_REPOSITORY.ORA 0 Full 653580 02-APR-
```

Now we can use the tag qualifier to delete all backup pieces with that tag. We created a text file called delete_tag.rman that contained these entries.

```
allocate channel for delete type 'sbt_tape';
change backuppiece tag = 'Beach Daily' delete;
release channel;
```

Then we can call this RMAN script using a Windows command file that we created, delete_tag.cmd, which sets the ORACLE_SID and then executes the RMAN command. That way, you can schedule the delete_tag.cmd using Windows or TSM scheduling.

```
set ORACLE_SID=BEACH
rman target internal/manager@beach rcvcat rman/rman@rio
cmdfile=c:\oracle\admin\scripts\delete_tag.rman
```
Chapter 12. Recovering Oracle databases on Windows 2000

The earlier chapters in this Windows 2000 section have covered the various procedures and methods for backing up Oracle databases. Now we shall address the important topic of recovery of these databases.

12.1 Restore operations

Recovering a database, tablespace, or data file is a two-stage process. The object is first restored and then it is recovered.

The restore process restores the necessary full or incremental level 0 backups. Incremental backups at levels greater than 0 are not restored. These are restored during the subsequent recovery process.

By default, the objects are restored to their original location as specified in the recovery catalog. An alternative location can also be specified if required. RMAN uses the recovery catalog to select the most current backup sets for use in the restore. The mode in which the database is running determines whether a consistent or inconsistent restore operation can be performed.

12.1.1 Consistent database recovery

If the database is running in NOARCHIVELOG mode, it can be restored only from a whole database backup. The control files and all data files are restored from a consistent backup. The database must be mounted but not open during the restore operation.

After a consistent restore, the database can be opened without performing any recovery. Any updates to the database after the backup are lost.

12.1.2 Inconsistent database recovery

If the database is running in ARCHIVELOG mode, a subset of the database such as a data file, tablespace, or the entire database can be restored to the most current state or to a specific point in time. This type of restore is inconsistent because the database cannot be started after the restore. The restore operation must be followed by a recovery operation. After the recovery is finished, the database can then be opened. The following objects can be restored:

- Database
- Tablespace
- Data file
- Control file
- Archive log

If the control files are lost, they must be restored before other restore operations can be performed. Only after restoring the control files can the target database be mounted and the other restore operations started.

A restore operation is typically followed by a recovery operation.

### 12.2 Recovery Operation

Recovery is a process whereby a restored file is made available, either to the most current state or to a specific point in time.

Once the data files are restored, they have to be made consistent with each other. The `recover` command is used to perform media recovery and to apply incremental backups. During the recovery process, RMAN automatically restores the archived redo logs as required.

If RMAN has a choice between restoring incremental backup sets or applying redo logs, it always uses the incremental backups. If overlapping levels of incremental backup are available, the lowest level of incremental backup, the one covering the longest period of time, is chosen automatically.

The following objects can be recovered:

- **Individual data files**
  One or more data files can be recovered. The target database must be started and mounted. If the target database is open, the data file must be offline.

- **Tablespaces**
  All data files in a tablespace can be recovered or a tablespace can be recovered to a previous point in time. The target database must be started and mounted. If the target database is open, the tablespace must be offline.

- **The entire database**
  The entire database can be recovered under the following circumstances:
  - A media failure has damaged the entire database.
  - The entire database must be recovered to a previous point in time.
  - The control files have been lost.

For database recovery, the database must be started but not open.
Archive log backup sets are restored as needed to perform a recovery. They are restored to the current archive log destination as specified in the `init.ora` file.

### 12.3 Recovery examples

In the following sections we provide some examples of recovery using RMAN.

#### 12.3.1 Data file recovery

To test a datafile recovery with RMAN, we performed following steps:

- The Oracle instance `beach` was up and running, and a backup of the database was taken.
- User `scott` in a SQL*Plus session connected to `beach`, created a new table `oops` in the `USERS` tablespace, inserted a row, and committed the changes.
- The database was shut down with the `shutdown immediate` command.
- The `USERS` data file was destroyed by the command
  ```
  C:\oracle\oradata\beach>echo hello > users01.dbf
  ```
- The database was started with the `startup` command, an error was returned.
- The alertlog file of the instance contained the following entries.

```
Errors in file c:\oracle\admin\beach\bdump\beachDBW0.TRC:
ORA-01157: cannot identify/lock data file 3 - see DBWR trace file
ORA-01110: data file 3: 'C:\ORACLE\ORADATA\BEACH\USERS01.DBF'
ORA-27048: skgfifi: file header information is invalid
OSD-04004: invalid file header
```

- Next, we created the following RMAN command file and executed it:

```
RMAN> run {
    allocate channel t1 type 'sbt_tape';
    restore datafile 'C:\ORACLE\ORADATA\BEACH\USERS01.DBF';
    recover datafile 'C:\ORACLE\ORADATA\BEACH\USERS01.DBF';
    release channel t1;
    alter database open;
}
```
• RMAN successfully restored the database and applied the redo logs, bringing the datafile up to the point where the database was shut down. We started the database and was able to see the **OOPS** table that was created after the database backup was taken:

```sql
SQL> select table_name, tablespace_name from user_tables;
```

<table>
<thead>
<tr>
<th>TABLE_NAME</th>
<th>TABLESPACE_NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>MYTABLE</td>
<td>SYSTEM</td>
</tr>
<tr>
<td>OOPS</td>
<td>USERS</td>
</tr>
</tbody>
</table>

### 12.3.2 Complete recovery

For this complete recovery example, we did the following:

• Made sure we had a prior full RMAN database backup
• Archived the redo logs
• Created a new table `recoverme`
• Inserted some rows into this table
• Shut down the database `beach`
• Deleted the datafiles

```bash
C:\oracle\oradata\beach>dir
Volume in drive C has no label.
Volume Serial Number is 3AAC-EBBF
Directory of C:\oracle\oradata\beach
04/03/2001 12:06p <DIR> .
04/03/2001 12:06p <DIR> ..
04/03/2001 11:11a <DIR> archive
04/03/2001 03:02p 3,006,464 CONTROL01.CTL
04/03/2001 03:02p 3,006,464 CONTROL02.CTL
04/03/2001 03:02p 3,006,464 CONTROL03.CTL
04/03/2001 03:02p 20,979,712 DR01.DBF
04/03/2001 03:02p 20,979,712 INDX01.DBF
04/03/2001 02:51p 31,473,664 OEM_REPOSITORY.ORA
04/03/2001 03:02p 52,436,992 RBS01.DBF
04/03/2001 03:02p 1,049,088 REDO01.LOG
04/03/2001 03:02p 1,049,088 REDO02.LOG
04/03/2001 03:02p 1,049,088 REDO03.LOG
04/03/2001 03:02p 297,803,776 SYSTEM01.DBF
04/03/2001 03:02p 20,979,712 TEMP01.DBF
04/03/2001 03:02p 10,493,952 TOOLS01.DBF
04/03/2001 03:02p 20,979,712 USERS01.DBF
14 File(s) 488,293,888 bytes
3 Dir(s) 520,306,688 bytes free
C:\oracle\oradata\beach>del *.dbf
```
• Next we started RMAN, connected to the recovery catalog, and started the database in mount mode:

```sql
C:\oracle\admin\scripts>rman target internal/manager@beach rcvcat rman/rmanario
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-06193: connected to target database (not started)
RMAN-06008: connected to recovery catalog database
RMAN> startup mount;
RMAN-06196: Oracle instance started
RMAN-06199: database mounted
```

• Once the database is started in mount mode, we are ready to run the RMAN script to recover the database:

```sql
RMAN> run {
  2> allocate channel t1 type 'sbt_tape';
  3> restore database;
  4> recover database;
  5> alter database open;
  6> release channel t1;
  7> }
```

When the RMAN script completed, the database was restored, archived logs were applied, and the database was completely recovered.

### 12.3.3 Incomplete / partial disaster recovery

For this partial disaster recovery example, we did the following.

• Made sure we had a prior full RMAN database backup
• Archived the redo logs
• Created a new table `lost`
• Inserted some rows into this table
• Shut down the database `beach`
• Deleted the datafiles, controlfiles, redologs
Next we connected to the recovery catalog using RMAN, and listed the all the logs that had been previously backed up:

```
C:\oracle\oradata\beach>dir
Volume in drive C has no label.
Volume Serial Number is 3AAC-EBBF
Directory of C:\oracle\oradata\beach
04/03/2001 10:08a <DIR> .
04/03/2001 10:08a <DIR> ..
04/03/2001 10:36a <DIR> archive
04/03/2001 10:36a 3,006,464 CONTROL01.CTL
04/03/2001 10:36a 3,006,464 CONTROL02.CTL
04/03/2001 10:36a 3,006,464 CONTROL03.CTL
04/03/2001 10:36a 20,979,712 DR01.DBF
04/03/2001 10:36a 20,979,712 INDEX01.DBF
04/03/2001 10:36a 31,473,664 OEM_REPOSITORY.ORA
04/03/2001 10:36a 52,436,992 RBS01.DBF
04/03/2001 10:36a 1,049,088 REDO01.LOG
04/03/2001 10:36a 1,049,088 REDO02.LOG
04/03/2001 10:36a 1,049,088 REDO03.LOG
04/03/2001 10:36a 297,803,776 SYSTEM01.DBF
04/03/2001 10:36a 20,979,712 TEMP01.DBF
04/03/2001 10:36a 10,493,952 TOOLS01.DBF
04/03/2001 10:36a 20,979,712 USERS01.DBF
14 File(s) 488,293,888 bytes
C:\oracle\oradata\beach>del .
C:\oracle\oradata\beach\*, Are you sure (Y/N)? y
```

```
RMAN> list backup of archivelog all;
List of Archived Logs Included
Thrd Seq Low SCN Next SCN Low Time Next Time
---- ------- ---------- ---------- --------------- ---------------
1 69 653786 653828 03-APR-01 03-APR-01

```

• Next we connected to the recovery catalog using RMAN, and listed the all the logs that had been previously backed up:
Information from that command was used in the set until portion of the RMAN restore script:

```
RMAN> run {
  2> set until logseq = 69 thread = 1;
  3> allocate channel t1 type 'sbt_tape';
  4> restore controlfile;
  5> alter database mount;
  6> restore database;
  7> recover database delete archivelog;
  8> alter database open resetlogs;
  9> release channel t1;
 10> }
```

- In this example:
  - set until logseq = 69 thread = 1; specifies a logseq of 69 and a thread of 1 for the subsequent restore and recovery commands. This information was collected from the list backup of archivelog all; command.
  - restore controlfile causes RMAN to restore the controlfiles to all control file destinations. The control files are not restored by the restore database command.
  - restore database restores all datafiles belonging to the database. Existing datafiles are overwritten.
  - recover database delete archivelog deletes restored archived logs that are no longer needed.
  - alter database open resetlogs is the only way to open the database again, because the data in the redo logs cannot be used any longer.

- After a successful database point-in-time recovery, a consistent backup of the database is required.

12.3.4 Duplicate database

For an example of duplicating a database, see Section 8.3.4, “Duplicate database” on page 114.
## Appendix A. Quick-start checklist for configuration

<table>
<thead>
<tr>
<th>Action</th>
<th>UNIX chapters</th>
<th>Windows chapters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define TSM and TDP environment, edit user option and system option files</td>
<td>Chapter 5.1.2, “Define environment variables” on page 62 and Chapter 5.1.3, “Edit the client options files” on page 64</td>
<td>Chapter 9.4, “Configuring the TDP for Oracle client options file” on page 130</td>
</tr>
<tr>
<td>Registering the database server with the TSM server</td>
<td>Chapter 5.1.4, “Registering with the TSM server” on page 66</td>
<td>Chapter 9.1, “Registering a node on the TSM Server for TDP for Oracle backups” on page 121</td>
</tr>
<tr>
<td>Initializing the TDP for Oracle password file</td>
<td>Chapter 5.1.5, “Initialize the password” on page 66</td>
<td>Chapter 9.5, “Generating the encrypted password” on page 132</td>
</tr>
<tr>
<td>Set up RMAN with TDP for Oracle</td>
<td>Chapter 5.1.6, “Setting up RMAN” on page 67</td>
<td>Done automatically during installation</td>
</tr>
<tr>
<td>Create a RMAN recovery catalog</td>
<td>Chapter 6.2, “Creating a database for the RMAN catalog” on page 73 and Chapter 6.2.1, “Creating the RMAN catalog” on page 73</td>
<td>Chapter 10.1.2, “Create the RMAN catalog” on page 157</td>
</tr>
<tr>
<td>Register database with the recovery catalog</td>
<td>Chapter 6.2.2, “Registering the target database” on page 75</td>
<td>Chapter 10.1.3, “Registering the target database” on page 158</td>
</tr>
<tr>
<td>Backup the database using RMAN and TDP for Oracle</td>
<td>Chapter 6.3.2, “Whole database backup” on page 78</td>
<td>Chapter 10.3.1, “Consistent whole database backup” on page 163</td>
</tr>
<tr>
<td>Backup archived redo logs using RMAN and TDP for Oracle</td>
<td>Chapter 6.4.2, “Archived redo log backup” on page 83</td>
<td>Chapter 10.3.2, “Archived redo log backup” on page 165</td>
</tr>
<tr>
<td>Action</td>
<td>UNIX chapters</td>
<td>Windows chapters</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>List backups in RMAN catalog and TSM catalog</td>
<td>Chapter 7.2.2, “List utility” on page 96 and Chapter 7.4, “List objects in the TSM database” on page 102</td>
<td></td>
</tr>
<tr>
<td>Test a RMAN restore and recovery</td>
<td>Chapter 8.3.1, “Datafile recovery” on page 109</td>
<td></td>
</tr>
<tr>
<td>Automate backups and deletion of old backups</td>
<td>Chapter 7.1.2, “Automating RMAN backups” on page 90</td>
<td></td>
</tr>
</tbody>
</table>
Appendix B. Troubleshooting

In this appendix, we will have a look at common problems and how they can be resolved. We give a list of the most interesting logfiles and techniques to determine the source of the problem.

A useful thing to do when troubleshooting a problem is to start a TSM administrative client in console mode. This will allow you to see the activity that is occurring on the TSM server. Among other things you will see node sessions start and stop and any TSM server error messages.

The administrative client console can be started with the `dsmadmc -cons` command. See the following example where we monitor how the TSM client BRAZIL_ORACLE connects and disconnects from the TSM server.

```
$ dsmadmc -cons
Tivoli Storage Manager
Command Line Administrative Interface - Version 4, Release 1, Level 2.0
(C) Copyright IBM Corporation, 1990, 1999, All Rights Reserved.

Enter your user id: admin
Enter your password:

Session established with server BRAZIL: AIX-RS/6000
   Server Version 4, Release 1, Level 2.0
   Server date/time: 04/02/01 17:02:02 Last access: 04/02/01 17:01:01

ANR0406I Session 9114 started for node BRAZIL_ORACLE (Tcp/Ip 9.1.150.57(39918)).
ANR0403I Session 9114 ended for node BRAZIL_ORACLE .
```

B.1 "Gotchas"

One of the most common problems is that the TSM API client is not able to connect to the TSM server.

B.1.1 RC 406 options file not found

If you receive a return code of 406, it means that the client options file `dsm.opt` cannot be found. Here is an example of the error:

```
  db2 => backup db sample user db2admin using itsconj use tsm
SQL2062N An error occurred while accessing media
"C:\PROGRA~1\SQLLIB\bin\db2adms.dll". Reason code: "406"
```
B.1.1.1 Windows
Make sure that the environment variable DSMI_CONFIG is NOT specified. You should use a client options file named dsm.opt and have it located in the installation directory of TDP for Oracle for Windows. You should not specify a different options file in your RMAN scripts. Make sure that the options file exists at that directory.

B.1.1.2 UNIX
Make sure that your RMAN script environment variable points to an options file that exists and that the Oracle user has rights to read this file.

B.1.2 Checking the library files
One of the first things to do is to check if the TDP for Oracle library file is the same one that Oracle is trying to load.

B.1.2.1 Windows
On Windows the library file is named ORASBT.DLL. Upon installation this file is copied to the TDP for Oracle installation directory as well as the Windows ...\system32 path. You should check that these files have the same characteristics (size, date, and so on.). See Figure 58.

<table>
<thead>
<tr>
<th>Search Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>orasbt.dll</td>
</tr>
<tr>
<td>orasbt.dll</td>
</tr>
</tbody>
</table>

Figure 58. Using Windows search to check orasbt.dll files

B.1.2.2 UNIX
On UNIX, the libobk.<suffix> file exists in the TDP for Oracle installation directory. There are symbolic links to that file in /usr/lib and in the Oracle directory. The following UNIX command will provide a long listing of every occurrence of libobk.<suffix>:

```
find / -name 'libobk.*' -ls 2>/dev/null
```
B.1.3 Running a TSM CONFIG trace

You can waste a lot of time troubleshooting before finding out that you were using the wrong options file. There is a simple trace that can be run that provides a lot of useful information. To run this trace, add two trace lines to your dsm.opt file (for both UNIX and Windows):

```
traceflag config
tracefile c:\temp\apitrace.out
```

Modify the path and filename of the TRACEFILE line to your operating system. The path must exist.

On Windows, our dsm.opt file looks like this after adding these trace lines.

```
commmethod tcpip
tcpport 1500
tcpserveraddress 193.1.1.11

node jamaica_db2
passwordaccess generate

traceflag config
tracefile c:\temp\apitrace.out
```

With the trace lines in the client options file, the next time a backup using TSM is run, a file will be created as specified by the tracefile command.

If this file is created, you have already verified something important. That the client options file that you modified is the one that the TSM client is using.

Open this file with a text editor. This trace will show you information regarding the level of the TSM API, tcpserveraddress, commmethod, errorlogname, ds_dir, passwordaccess, and so on. Pay special attention to the errorlogname and the ds_dir.
The value for errorlogname will either be dsierror.log or a path and dsierror.log. For UNIX the Oracle user performing the backup must have write access to the directory and path of the errorlogname.

The ds_dir value corresponds to the DSM_DIR environment variable. For UNIX this value also determines from which directory the dsm.sys file will be read. Most TSM installations have a dsm.sys in the .../tivoli/tsm/ba/bin and the .../tivoli/tsm/api/bin directories. When matching the servername value in the dsm.opt file with the corresponding servername in the dsm.sys, use the ds_dir value to determine which dsm.sys is being used.

B.1.4 RC 137 authentication failure, incorrect password

The following are necessary considerations for Windows and UNIX systems.

B.1.4.1 Windows

The most common cause is that `passwordaccess generate` is not specified. Another cause is that you are not specifying the correct password, or your nodename is wrong. To update the password on the TSM server, run the command `update node nodename password`. Replace nodename and password with the appropriate values. Use the administrative client in console mode to verify which node is trying to access the TSM server.

B.1.4.2 UNIX

The most common cause is that you are not using the correct dsm.opt and dsm.sys. Use the config trace or the orcagent.log trace to try and isolate the problem. These files and how to create them are discussed elsewhere in this appendix. You must have `passwordaccess prompt` specified. Another cause could be that you are not specifying the correct password, or your nodename is wrong. To update the password on the TSM server, run the command `update node nodename password`. Replace nodename and password with the appropriate values. Use the administrative client in console mode to verify which node is trying to access the TSM server.

B.2 Things to check on the TSM server

This section covers various things you need to check on the TSM server.

B.2.1 Specifying commtimeout in the dsmserv.opt

If an Oracle database has a lot of empty blocks (database has very little data), a commtimeout error may be witnessed during the backup. This commtimeout error will cause the entire Oracle backup to fail.
The problem is that during the Oracle backup, a session is opened with the TSM server and the backup transaction is started. After data that exists in the Oracle file is sent to the TSM server, the backup then goes into a loop skipping the empty space in the database file (Null values). If there is a lot of empty space, the TSM server can time-out and close the session before RMAN can give the TDP client any additional data to send or finish the transaction.

This issue is normally witnessed if the value for comtimeout on the TSM server is set to a low value (such as the default of 60 seconds). It is suggested to set the comtimeout value to 600-1200 seconds when working with database backups. This value is set in the dsmserv.opt file for the TSM server and requires the TSM server to be restarted to take effect.

### B.2.2 Administrative client queries

If you are encountering errors, there are some things that you should check on the TSM server using the administrative client and the query command.

Run the command `query node nodename f=d` replacing nodename with the nodename that is used to back up Oracle. The value for Backup Deleted Allowed should be set to YES. The value for Maximum Mount Points Allowed should be equal to or less than the number of drives on the TSM server. The value for Locked? should be set to NO. The value for Policy Domain Name should be checked to make sure it belongs to the correct domain. This value determines what management classes are available to the node for managing the Oracle backups.

Using the value of Policy Domain Name, you can run some queries to see if the correct management classes for the Oracle backups are available. The command `query mgmtclass <domain> active` will show which is the DEFAULT management class for the node. The command `query copygroup <domain> active` will tell you what the retention settings are for the management classes.

### B.3 Isolating the problem

When troubleshooting, you must try to isolate the cause of the problem. The four main areas in which to try to isolate the problem are: Oracle RMAN, TDP for Oracle, TSM API, and TSM Server.
A key thing to remember is that TSM is just a storage repository for RMAN backups. RMAN is responsible for creating all the backup objects and the archivelogs. TDP for Oracle is responsible for taking what RMAN gives and passing that on to the TSM server. TSM server is responsible for storing the backup objects and the archivelogs, and then retrieving them when asked. If RMAN creates a corrupt backup object or corrupt archivelog, TSM will store it and retrieve it as corrupt. As TSM is not necessary to do an RMAN backup, you can eliminate TSM as a problem by creating an RMAN backup on a local file system. If RMAN fails creating a backup to disk, then it will probably also fail using TDP for Oracle. In this situation, you would then check the Oracle setup.

Another useful way to isolate a problem is to compare a failing machine with a working one. Then you can see what is the same and what is different. Are they using the same API level, TDP for Oracle level, Oracle level and fixpack, TSM server? Do the nodes belong to the same domain and have the same settings? What are the client options file differences between the failing and the working machine?

Try to use as many of the defaults as possible. Using environment variables when not required increases the likelihood of configuration errors.

### B.4 List of logfiles

The following is a collection of logfiles from which to get valuable information for problem determination:

- **dsierror.log**
  
  The file reside in the directory specified by the DSMI_LOG environment variable. It contains the most valuable information for problem determination for our purpose. Each entry may refer to a TSM API error number. A description of those errors can be found in Appendix D, “API Return Codes”, in the book: *Tivoli Storage Manager: Using the Application Program Interface*, SH26-4123.

- **dsmsched.log**
  
  By default, TSM stores the schedule log information in a file named dsmsched.log. The default name can be overwritten by specifying the SCHEDLOGNMAME option in the client option files.

- **sbtio.log**
  
  This file is created by Oracle and normally resides in the udump directory of the database instance.
• oragent.log
This file is created by specifying DSMO_DEBUG=49 when allocating channels for RMAN commands. This is one of the most useful files to generate for troubleshooting purposes. An example of creating and reading this file was given in Section 9.6, “Verify a TDP for Oracle for Windows RMAN backup” on page 133.

• readme file
The readme files contain information that has not made it into the product manual. There is often information in the readme that will help troubleshoot known problems. The readme files can be found in the installation package or in the install directory.

• dsoerror.log
This file is only on Windows and is created if Oracle encounters a problem loading the TDP for Oracle orasbt.dll file. TDP for Oracle backups will not work if this file is created when starting the Oracle database service. The readme for TDP for Oracle describes this in greater detail.

B.5 Inconsistencies between Oracle RMAN and TSM

RMAN is utilized to remove backup objects that were saved on the TSM server with the Tivoli Data Protection for Oracle. When performing these backup deletions, it is very important to ensure that the same environment variables are used for the deletion as were specified for the backups. If the same TDP environment variables are not specified, the backup item will be removed from the RMAN catalog, but will most likely remain on the TSM server in an active state. If they backup objects are not marked for deletion on the TSM server, they will remain in an active state and will never be expired and continue to use TSM storage space.

It is very important to verify that the deletion script being used list the environment variables in the ENV PARAMs correctly (including the case sensitivity of the node name). Also check that the variable for the DSI_LOG should only be a directory (the file name will be dsierror.log) and this directory must exist and have writable rights by the Oracle user doing the backup/deletion. This dsierror.log will possibly contain some error messages if there was a problem encountered during the deletion process.

Whenever Oracle RMAN performs a backup of a database or archivelog, RMAN writes an entry into the controlfile or recovery catalog. This happens whether you use TDP for Oracle or not. RMAN requires this entry for restores. When TDP for Oracle is used, there is an entry for the backup piece in the TSM server and in RMAN.
When you use the RMAN command to delete a backup piece, the corresponding object on TSM storage needs to be inactivated. If this does not happen, then an inconsistency has been introduced between Oracle RMAN and TSM. Common to all computing, whenever you keep information about an object in two locations, situations can arise that cause inconsistency between the two locations.

There are two types of inconsistent states that you find. Firstly, if Oracle RMAN has entries for backup pieces in the control file or recovery catalog, and the corresponding backup piece does not exist on TSM storage. The second is the reverse situation, where the TSM server has active backup pieces in its storage, but there is no corresponding entry in RMAN.

### B.5.1 RMAN references a non-existent TSM object

For this situation, you can use the RMAN `crosscheck` command. To simulate this type of a problem, we took an RMAN backup, then used the TSM server command `delete filespace` to delete the TSM objects without removing the RMAN references to these objects.

With RMAN and TSM in this inconsistent state, we did the following to synchronize them. All of these steps are done from RMAN.

1. allocate channel for maintenance type 'sbt_tape';

```
RMAN> allocate channel for maintenance type 'sbt_tape';
RMAN-03022: compiling command: allocate
RMAN-03023: executing command: allocate
RMAN-08030: allocated channel: delete_02
RMAN-08500: channel delete_02: sid=14 devtype=SBT_TAPE
RMAN-08526: channel delete_02: MMS Version 2.1.10.0
```

2. crosscheck backup;

```
RMAN> crosscheck backup;
RMAN-03022: compiling command: XCHECK
RMAN-03023: executing command: XCHECK
RMAN-08074: crosschecked backup piece: found to be 'EXPIRED'
RMAN-08517: backup piece handle=STARTER/426340296/12/1 recid=12 stamp=426340296
RMAN-03023: executing command: partial resync
RMAN-08003: starting partial resync of recovery catalog
RMAN-08005: partial resync complete
```
3. delete expired backup;

```
RMAN> delete expired backup;
RMAN-03022: compiling command: delete expired
RMAN-03023: executing command: delete expired
RMAN-08517: backup piece handle=STARTER/426340296/12/1 recid=12 stamp=426340296
RMAN-08073: deleted backup piece
RMAN-03023: executing command: partial resync
RMAN-08003: starting partial resync of recovery catalog
RMAN-08005: partial resync complete
```

The RMAN catalog now matches what is on the TSM server.

### B.5.2 TSM object exists without a corresponding RMAN reference

This type of situation is much harder to overcome. You should only undertake the following procedures if you are completely knowledgeable, since this will remove valid backups as well as inconsistent backups if not done correctly.

If a situation is encountered where there are RMAN backups that are not marked inactive on the TSM server, there are two ways to remove these unwanted active backups.

- Use the unsupported API sample program to delete the individual backup objects.
- On the TSM server, rename/delete the filespace (containing all the backups).

#### B.5.2.1 Using the API sample program to delete objects

There is a program called Sample that can be used to access the individual API objects on TSM. The Sample program is not supported, but development has included it in an uncompiled state as part of the API installation. Appendix E of the manual *Tivoli Storage Manager: Using the Application Program Interface*, SH26-4123, has some references to setting up and running this application. This manual can be downloaded from the Web site:


This Sample application has the capability to work with individual API backup objects and can delete a specific backup object.
B.5.3 Renaming / deleting the TSM filespace

Support normally recommends that you rename the existing filespace and then wait a few weeks until backups have accumulated again. After some period of time, you can delete the old (renamed) filespace in order to delete the backups that you no longer want to keep.

The filespace for a specific node can be renamed from the TSM Server using a TSM administrative client session (dsmadmc).

To see the existing filespaces for a specific node:

```
Query filespace <NODENAME>
```

To rename the filespace:

```
Rename filespace <nodename> <ExistingFilespaceName> <NewFilespaceName>
```

To delete the filespace:

```
Delete filespace <nodename> <ExistingFilespaceName>
```

Both the rename and delete of a filespace is on a per-Node basis. Therefore, if your Oracle backups are using different TSM nodenames for each instance, then you can work with these on an individual basis.

**Note:** The Oracle 8i RMAN has a SYNCH capability to look at the backups in the RMAN catalog and compare them with the Tape library vendor software (in this case, the TSM server). In version 2.2 of the TDP for Oracle, this functionality is being provided to remove the unwanted backup objects. See Appendix C, “TDP for Oracle Version 2.2” on page 209.
Appendix C. TDP for Oracle Version 2.2

The TDP for Oracle V2.2 code became available during the course of writing the redbook, so we were able to do some provisional setup and installation of this code. We have included the details in this appendix to highlight the new features.

C.1 New features

TDP for Oracle Version 2.2 is designed to take advantage of the following enhancements:

- The new tdpo.opt file for all options
- The duplex backup copy feature
- The new TDP for Oracle utilities, TDPOCONF and TDPOSYNC (command line utilities)
- The multi-buffering of RMAN data transfers

C.1.1 TDPO.OPT file

This feature provides a centralized place to define all the options needed by RMAN for TDP for Oracle backup and restore operations. This eliminates the need to specify environment variables for each session, thereby reducing the potential for human error. This also simplifies the establishment of multiple sessions.

C.1.2 Duplex copy (Oracle 8.1.x only)

This feature allows you to back up multiple copies of the same file to different media by assigning different management classes in the TDP for Oracle options file for each backup copy.

C.1.3 Configuration utility (TDPOCONF)

This external executable helps run certain tasks to facilitate setup and aid in troubleshooting setup scenarios. The tdpoconf utility uses the tdpo.opt file for its options.
C.1.4 Synchronization utility (TDPOSYNC)

This external executable allows the DBA to resolve discrepancies between an RMAN catalog and the TSM server due to errors that can occur during RMAN file deletion. Identifying and removing backups from the TSM server that no longer exist in the Oracle catalog frees up space on the server. The tdposync utility uses the tdpo.opt file for its options.

C.1.5 Multi-buffering of data transfers

TDP for Oracle introduces the option to turn on multi-buffering during backups, which can significantly increase poor performance due to network constraints. If you use this option, data buffers sent from RMAN are saved in internal TDP for Oracle buffers instead of being sent directly to the TSM server across the network. TDP for Oracle then starts a new thread, which picks up these internal buffers and sends them to the server. In this way, RMAN does not need to wait for each actual data transfer to be completed via the network before reading more data.

C.1.6 LAN-free data movement

TDP for Oracle supports LAN-free backup operations, which shifts the movement of data from the communications network to a storage area network (SAN). Data moves over the SAN to a SAN-attached storage device via the TSM Storage Agent. Running TDP for Oracle in a LAN-free environment decreases the load on the TSM server, allowing the server to support a greater number of simultaneous connections, and avoids constraints of the network. To enable LAN-free support, you must install the following applications on the same system as TDP for Oracle:

- TSM Storage Agent
- TSM Managed System for SAN Storage Agent

C.1.7 Migration and coexistence

- Existing backups created using older versions of TDP for Oracle are restorable using TDP for Oracle Version 2.2.
- TDP for Oracle relies heavily on the tdpo.opt file for accuracy in setup and maintenance. The contents of the tdpo.opt file are different to the environment variables for TDP for Oracle Version 2.1;
- Version 2.1 RMAN scripts for environment variables are ignored.
- TSM Version 3.7 and above servers are supported.
C.2 Configuration and setup on UNIX

This section contains client environment information and software requirements for installing TDP for Oracle on a AIX or Sun Solaris platform.

C.2.1 Installing TDP for Oracle

These are the main differences from the previous version of TDP:

- Oracle7 inclusive Enterprise Backup Utility (EBU) is no longer supported.
- TSM API version 3.2 is not supported, but TSM API version 4.1.2 is included on the CD.
- AIX 4.3.1 is not supported.
- SNA LU6.2 is not supported.
- A new utility is included as a separate package.
- A new installation path is provided for TDP 64-bit architecture.

C.2.1.1 Operating system requirements

- AIX 4.3.2
- AIX 4.3.3
- Solaris 2.6
- Solaris 7
- Solaris 8

C.2.1.2 Software requirements

- TSM Server Version 3 (or higher)
- TSM API Version 4.1.2
- Oracle8 Version 8.0.x or 8.1.x

C.2.1.3 Communication methods

- TCP/IP
- Shared memory

The installation itself is performed with `smitty install_latest` on AIX and with `pkgadd` on Solaris, respectively.
Table 4 contains an overview of the different packages and installation paths.

Table 4. Default installation directories.

<table>
<thead>
<tr>
<th>Package</th>
<th>O/S</th>
<th>Package name</th>
<th>Default installation directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDP for Oracle 32-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.32bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TDP for Oracle utility 32-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.tools.32bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TSM API 32-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.api.32bit</td>
<td>/usr/tivoli/tsm/client/api/bin</td>
</tr>
<tr>
<td>TDP for Oracle 64-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.64bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin64</td>
</tr>
<tr>
<td>TDP for Oracle utility 64-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.oracle.tools.64bit</td>
<td>/usr/tivoli/tsm/client/oracle/bin64</td>
</tr>
<tr>
<td>TSM API 64-bit</td>
<td>AIX</td>
<td>tivoli.tsm.client.api.64bit</td>
<td>/usr/tivoli/tsm/client/api/bin64</td>
</tr>
<tr>
<td>TDP for Oracle 32-bit</td>
<td>Solaris</td>
<td>TDPoracle32.pkg</td>
<td>/opt/tivoli/tsm/client/oracle/bin</td>
</tr>
<tr>
<td>TSM API 32-bit</td>
<td>Solaris</td>
<td>TIVsmCapi32.pkg</td>
<td>/opt/tivoli/tsm/client/api/bin</td>
</tr>
<tr>
<td>TDP for Oracle 64-bit</td>
<td>Solaris</td>
<td>TDPoracle64.pkg</td>
<td>/opt/tivoli/tsm/client/oracle/bin64</td>
</tr>
<tr>
<td>TSM API 64-bit</td>
<td>Solaris</td>
<td>TIVsmCapi64.pkg</td>
<td>/opt/tivoli/tsm/client/api/bin64</td>
</tr>
</tbody>
</table>

C.2.2 Edit the TDP for Oracle options file

As mentioned in C.1.1, “TDPO.OPT file” on page 209, there is a new file which contains all TDP related configuration information.

There are several new options, and some of the TDP version 2.1.10 options have been renamed. See the TDP Installation and Users Guide for a detailed description of the new options.

For our tests; we placed the file `tdpo.opt` in the default installation path. This way, we were able to invoke RMAN without a PARMs ‘ENV...’ string in the allocate channel command:

```
DSM_LOG   /usr/local/oracle/log
DSM.ORC_CONFIG /usr/tivoli/tsm/client/oracle/bin/dsm.opt
TDCO_SECRET  TDCO_SECRET
TDCO_NODE   brazil_oracle
```

If you are using a TDP for Oracle options file that is not the default name, you must specify the options file name in the PARMs ‘ENV=()’; section of the channel allocation of RMAN:
allocate channel t1 type 'sbt_tape' parms
'ENV=(TDPO_OPTFILE=/mydir/myoptfile)';

C.2.3 Registering with TSM server

Before you can begin requesting services from a TSM server, the server must recognize your nodename. TSM uses a node name and a password (if one is required). TSM maintains a password for each node name. The process of setting up a node name and password is called registration.

Your TSM administrator must register your node name as a client with the server. You can then back up and restore Oracle data with the server after TDP for Oracle is properly configured. Because we only migrated the software and kept the same node name and file space name, no action by a TSM administrator was required.

C.2.4 Initializing the password

TDP for Oracle uses a new password utility for password generation and maintenance. The new TDPO configuration utility, tdpoconf replaces the previous executable aobpswd.

To generate or update a password, invoke as root user tdpoconf as follows:

```
tdpoconf password [-tdpo_optfile=optfile]
```

Successful execution of tdpoconf password should generate a password file with the prefix TDPO followed by your nodename.

We used the default installation directories for the tdpo.opt file and for the password file — therefore, we did not set TDPO_PSWDPATH in tdpo.opt:

```
# gwd
/usr/tivoli/tun/client/oracle/bin
# #! env | grep -E 'DSM|TDPO'
# ./tdpoconf password

*****************************************************************************
* Tivoli Data Protection for Oracle Utility *
* Password file initialization/update program *
* ROOT privilege needed to update value *
*****************************************************************************

Please enter current password:
Please enter new password:
ANU0260I Password successfully changed.
```

```
# ls -la TDPO
-rw-r--r-- 1 root system 57 Mar 28 18:08 TDPO.brazil_oracle
```
C.2.5 Validating the setup

We used the following techniques to validate the TDP for Oracle setup:

12.3.4.1 Displaying the TSM TDP environment with `tdpoconf`

Tivoli Data Protection for Oracle includes a new configuration utility that displays library, session, and server information. This `tdpoconf` utility emulates TDP for Oracle's behavior and displays the values being used to identify setup problems.

Following is an example, `tdpoconf showenv`, to display the actual TSM TDP environment:

```
oracle@brazil => id
uid=0(oracle) gid=300(dba) groups=(staff),836(oinstall)
oracle@brazil => pwd /usr/tivoli/tsm/client/oracle/bin
oracle@brazil => ./tdpoconf showenv

TDP FOR ORACLE INFORMATION
Version: 2
Release: 2
Level: 0
Sublevel: 0
Platform: TDP Oracle 11G

TSM SERVER INFORMATION
Server Name: BRAZIL_ORACLE
Server Address: 9.1.156.57
Server Type: AS3-RS/8000
Server Port: 1500
Communication Method: TCP/IP

SESSION INFORMATION
Owner Name: oracle
Node Name: BRAZIL_ORACLE
Node Type: TDP Oracle 11G
DSM_DIR: /usr/tivoli/tsm/client/oracle/bin
DSM_ORC_CONFIG: /usr/tivoli/tsm/client/oracle/bin/dsm.ost
TDPO_OPTSFILE: /usr/tivoli/tsm/client/oracle/bin/tdpo.opt
Password Directory: /home/oracle
Compression: TRUE

POLICY DATA
Domain Name: API_DOMAIN
Management Class: API_MGMTCLASS
Active Policy Set: API_POLICY
Backup Retention: 30
```

C.2.5.1 Invoking RMAN using the new TDP for Oracle version

The RMAN crosscheck command reads the recovery catalog and crosschecks the individual backup pieces in the media manager.
This way, TDP for Oracle was invoked for the first time by RMAN and found all backup pieces created by the previous TDP for Oracle version.

The following example illustrates invoking RMAN the first time with a type sbt_tape channel:

```
Recovery Manager: Release 8.1.7.0.0 - Production
RMAN-00805: connected to target database: BERLIN (DBID=3455039196)
RMAN-00806: connected to recovery catalog database
RMAN> allocate channel for maintenance type 'sbt_tape';
2> crosscheck backup;
3> release channel;
4> RMAN-03022: compiling command: allocate
RMAN-03023: executing command: allocate
RMAN-03030: allocated channel: delete
RMAN-03030: channel delete: sid=10 devtype=SBT_TAPE
RMAN-03026: channel delete: Tivoli Data Protection for Oracle: version 2.2.0.0
RMAN-03022: compiling command: XCHECK
RMAN-03023: executing command: XCHECK
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
RMAN-03074: crosschecked backup piece: found to be 'AVAILABLE'
```

- The crosscheck backup command requires a maintenance channel.
- The allocate channel command itself needs no additional environment parameters, because the tdpo.opt file is located in the default installation directory.

### C.3 Using the new features

Here are some examples to demonstrate the new features:

#### C.3.1 Multi-buffering

Tivoli Data Protection for Oracle now includes an option to transfer Oracle data to the TSM server using multiple buffers to help alleviate network bottlenecks.
This option can be set in the TDP for Oracle option file using the following parameter:

```
TDPO_NUM_BUFFERS #
```

This option specifies the number of buffers TDP for Oracle uses to support multi-buffering. By entering a value, multi buffering is activated.

The minimum value (number of buffers) is 5, and the maximum value (number of buffers) is 15. When this option is not set, TDP for Oracle operates with a single buffer.

We measured this option with runtimes of a full database backup with values of 2, 4, 8 and without this option. We achieved the best results with only 2 buffers.

**C.3.2 Duplex copy function**

Oracle 8.1.x allows you to make up to four, exact duplicate copies of a backup that can be stored on different backup media. TDP for Oracle Version 2.2 supports this Duplex Backup feature. This Duplex Backup feature uses different management classes for each backup copy. By default, the primary management class is the one assigned to the TSM client node name.

To activate the duplexing, the following setup steps must be performed.

**12.3.4.2 Oracle configuration**

In the init.ora file, the parameter `backup_tape_io_slaves` must be set to true.

```
BACKUP_TAPE_IO_SLAVES specifies whether I/O server processes ("slaves") are used by the Recovery Manager to back up, copy, or restore data to tape. When the value is set to TRUE, Oracle uses an I/O server process to write to or read from a tape device. When the value is set to FALSE (the default), Oracle does not use I/O server process for backups. Instead, the shadow process engaged in the backup accesses the tape device.

Typically, I/O server processes are used to simulate asynchronous I/O on platforms that either do not support asynchronous I/O or implement it inefficiently. However, you can use I/O server processes even when synchronous I/O is available. In that case, the I/O server processes will use asynchronous I/O. Only one process can access a tape device at any given time. Therefore, this parameter is Boolean, and allows or disallows deployment of an I/O server process to access a tape device.
```
C.3.2.1 TDP configuration
You may define up to 3 additional management classes in the tdpo.opt file. The parameters are TDPO_MGMT_CLASS2, TDPO_MGMT_CLASS3 and TDPO_MGMT_CLASS4. They describe the corresponding management classes.

The management classes have also to be defined in TSM. Be aware of the following considerations:

- You will receive an error message if you specify `set duplex 4` in the RMAN backup script and do not define enough TDPO_MGMT_CLASS options in the tdpo.opt file.
- Make sure the storage pool information for each backup copy group within the management classes is not the same.
- Make sure backups from these different storage pools are not migrated to the same storage pool at a later time.
- Be aware that duplicate data will be sent across the network.
- Be aware that if you specify `set duplex 4` and allocate one channel in the RMAN backup script, RMAN will start four sessions to the TSM server. Likewise, if you specify `set duplex 4` and allocate two channels in the RMAN backup script, RMAN will start eight sessions to the TSM server.
- Make sure RMAN does not start more sessions than the maximum mount points allowed by the TSM server. This TSM server option restricts the use of tape at one time by the node. You can view the maximum mount points allowed by the TSM server for a particular node by entering the following command from a TSM Administrative Client prompt: `q nodef=d`.

C.3.2.2 Backup example using the RMAN duplex option
In our test environment, the TSM server has only 2 tape drives, so the client node `brazil_oracle` has a maximum mount points allowed of 2.

The TSM administrator created a new management class with the following commands:

---

Note
You cannot perform duplexed backups unless you enable this parameter. Otherwise, Oracle returns an error. When this parameter is enabled, Recovery Manager will configure as many server processes as needed for the number of backup copies requested.
In the tdpo.opt file, we added TDPO_MGMT_CLASS2 DSKPOOL1.

The RMAN cmdfile looks like this:

```sql
oracle@brazil => more backup_duplex.berlin
run {
    set duplex=2;
    allocate channel t1 type 'sbt_tape';
    set maxcorrupt for datafiles 1 to 0;
    backup
        full
        setsize = 2097152
        format '%c/backupset%k.backuppiece%p.copy%c'
        tag 'berlin online 20010329.1200' (database);
    release channel t1;
}
```

Because of the duplex option, the format parameter has to include the %c string, which specifies the copy number of the backup piece within a set of duplexed backup pieces.

As shown next, we listed the objects in TSM and RMAN and deleted them with the following steps:

- Established an RMAN session and allocated a maintenance channel.
- Started a TSM administrator session.
- Under one backup set key, there are two different backup pieces because of the set duplicate=2 option. The format parameter .copy%c created names ending with .copy1 and .copy2.
- In the TSM administrator session, we listed the first to TSM objects with a SQL select command.
- The next step was the change backupset 2773 delete, which marks the two objects BERLIN/backuppiece27.backupset1.copy1 and BERLIN/backuppiece27.backupset1.copy2 in TSM as inactive.
- In the TSM administrator session, a expire inventory command was invoked.
- A new SQL select in the TSM session returned no rows.

Here is the RMAN list backup and change delete backup:
<table>
<thead>
<tr>
<th>Key</th>
<th>Recid</th>
<th>Stamp</th>
<th>LV Set Stamp</th>
<th>Set Count</th>
<th>Completion Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2773</td>
<td>23</td>
<td>425648717</td>
<td>0</td>
<td>425648723</td>
<td>27</td>
</tr>
</tbody>
</table>

**List of Backup Pieces**

<table>
<thead>
<tr>
<th>Key</th>
<th>PC#</th>
<th>CC#</th>
<th>Status</th>
<th>Completion Time</th>
<th>Piece Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>2775</td>
<td>1</td>
<td>1</td>
<td>AVAILABLE</td>
<td>2001/03/29 11:45:17</td>
<td>BERLIN/backupset27.backuploc1.copy1</td>
</tr>
<tr>
<td>2776</td>
<td>1</td>
<td>2</td>
<td>AVAILABLE</td>
<td>2001/03/29 11:45:17</td>
<td>BERLIN/backupset27.backuploc1.copy2</td>
</tr>
</tbody>
</table>

**List of Datafiles Included**

<table>
<thead>
<tr>
<th>File Name</th>
<th>LV Type</th>
<th>Ckp</th>
<th>SCN</th>
<th>Ckp Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>/u01/oradata/berlin/system01.dbf</td>
<td>Full</td>
<td>305049</td>
<td>2001/03/29 11:14:14</td>
<td></td>
</tr>
<tr>
<td>/u02/oradata/berlin/data01.dbf</td>
<td>Full</td>
<td>305049</td>
<td>2001/03/29 11:14:14</td>
<td></td>
</tr>
</tbody>
</table>

**List of Backup Sets**

<table>
<thead>
<tr>
<th>Key</th>
<th>Recid</th>
<th>Stamp</th>
<th>LV Set Stamp</th>
<th>Set Count</th>
<th>Completion Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2774</td>
<td>24</td>
<td>425648807</td>
<td>0</td>
<td>425648723</td>
<td>28</td>
</tr>
</tbody>
</table>

**List of Backup Pieces**

<table>
<thead>
<tr>
<th>Key</th>
<th>PC#</th>
<th>CC#</th>
<th>Status</th>
<th>Completion Time</th>
<th>Piece Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>2777</td>
<td>1</td>
<td>1</td>
<td>AVAILABLE</td>
<td>2001/03/29 11:46:47</td>
<td>BERLIN/backupset28.backuploc1.copy1</td>
</tr>
<tr>
<td>2778</td>
<td>1</td>
<td>2</td>
<td>AVAILABLE</td>
<td>2001/03/29 11:46:47</td>
<td>BERLIN/backupset28.backuploc1.copy2</td>
</tr>
</tbody>
</table>

**List of Datafiles Included**

<table>
<thead>
<tr>
<th>File Name</th>
<th>LV Type</th>
<th>Ckp</th>
<th>SCN</th>
<th>Ckp Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>/u01/oradata/berlin/tools01.dbf</td>
<td>Full</td>
<td>305050</td>
<td>2001/03/29 11:45:24</td>
<td></td>
</tr>
<tr>
<td>/u01/oradata/berlin/bs01.dbf</td>
<td>Full</td>
<td>305050</td>
<td>2001/03/29 11:45:24</td>
<td></td>
</tr>
<tr>
<td>/u02/oradata/berlin/db02.dbf</td>
<td>Full</td>
<td>305050</td>
<td>2001/03/29 11:45:24</td>
<td></td>
</tr>
<tr>
<td>/u01/oradata/berlin/users01.dbf</td>
<td>Full</td>
<td>305050</td>
<td>2001/03/29 11:45:24</td>
<td></td>
</tr>
<tr>
<td>/u02/oradata/berlin/index01.dbf</td>
<td>Full</td>
<td>305050</td>
<td>2001/03/29 11:45:24</td>
<td></td>
</tr>
</tbody>
</table>

```
RMWAN> change backupset 2773 delete;
```

```
RMWAN> change backupset 2774 delete;
```
The following is a TSM administrator session showing `select` and `expire` inventory commands:

```sql
TSM BRAZIL: select * from contents where node_name = 'BRAZIL_ORACLE' and file_name like '/berlin/backupset27.backuppiece1%'

VOLUME_NAME: /tsm/dskpool/vol16
NODE_NAME: BRAZIL_ORACLE
TYPE: Brus
FILESPEC_NAME: /sdsmorc
FILE_NAME: /berlin/backupset27.backuppiece1.copy1
AGREGATED: No
FILE_SIZE: 10230812
SEGMENT: 
CACHED: No

VOLUME_NAME: /sdsmorc
NODE_NAME: BRAZIL_ORACLE
TYPE: Brus
FILESPEC_NAME: /sdsmorc
FILE_NAME: /berlin/backupset27.backuppiece1.copy2
AGREGATED: No
FILE_SIZE: 10230812
SEGMENT: 1/1
CACHED: No

TSM BRAZIL: expire inventory
ANS8031I Process number 23 started.

TSM BRAZIL: select * from contents where node_name = 'BRAZIL_ORACLE' and file_name like '/berlin/backupset27.backuppiece1%'

ANS9003E SELECT: No match found using this criteria.
ANS8001I Return code 11.
```

### C.3.3 TDPOSync

This utility allows you to repair discrepancies between the TSM server and the RMAN catalog. These discrepancies can occur as a result of incorrect management class settings. This external executable allows the DBA to resolve discrepancies between an RMAN catalog and the TSM server due to errors that can occur during RMAN file deletion. Identifying and removing backups from the TSM server that no longer exist in the Oracle catalog frees up space on the server. The `tdposync` utility uses the `tdpo.opt` file for its options.
When you invoke `tdposync`, the utility does the following:

- Prompts you for RMAN catalog owner, password and TNS alias.
- Gathers information about catalogued Oracle database(s) in a recovery catalog.
- Crosschecks TSM server objects against RMAN backup entries.
- Displays formatted output to the screen (files that exist on TSM but not in the catalog database(s)).
- Prompts you to do one of the following:
  - Delete any files found causing the discrepancy.
  - Delete all files.
  - Exit the program without deleting files from the TSM server.
Appendix D. Performance

This appendix describes performance aspects of Oracle RMAN with TDP for Oracle. There are other performance aspects such as operating system tuning, Oracle database tuning, disk I/O tuning, and network tuning, which are not covered in this chapter. All the performance considerations in this chapter should be reviewed in the light of your own system, as they can only be offered as guidelines for review.

D.1 Generic performance issues

Some of the most important considerations when it comes to performance are not the technical aspects but the environment and organizational aspects. For example, let us assume that the enterprise has more than twenty Oracle servers, every server uses the same TSM server and on every server at exactly two o’clock in the morning a database backup is started. Here it does not make sense to tune backup performance.

In this situation, it is advisable, that only a small group of persons coordinates the schedule of all the different backups.

D.1.1 TSM and TDP generic performance issues

TDP for Oracle uses the following options in the client system option file dsm.sys and client user option file dsm.opt, which affect performance:

• COMPRESSION
  - Specify COMPRESSION YES for TDP for Oracle to compress data before sending it to the TSM server. This reduces storage usage on the TSM server. COMPRESSION YES affects performance in two ways:
    - CPU utilization is higher on the machine on which TDP for Oracle is running. Network bandwidth utilization is lower because fewer bytes are transmitted. It is recommended that COMPRESSION YES be specified when any of the following conditions exist:
      • The network adapter has a data overload.
      • Communications between TDP for Oracle and the TSM server are over a low bandwidth connection.
      • There is heavy network traffic.
    - Specify COMPRESSION NO if the machine running TDP for Oracle has a CPU overload. This prevents additional CPU usage from impacting other applications, including the Oracle server.
- The value of the compression option for TDP for Oracle is honored only if the TSM administrator leaves the compression decision to the node. The TSM administrator can restrict use of the compression option by specifying certain restrictions on the TSM server side.

- TRACEFLAGS TRACEFILE
  - Once your TDP for Oracle is set up correctly, it should not be necessary to run tracing, which may produce large trace files, but during performance analysis these may be useful. Once testing is finished, be sure to remove all trace settings in the `dsm.opt` file, as these may themselves cause a performance overhead.

D.1.2 RMAN performance options

These are some performance considerations relating to RMAN options:

D.1.2.1 Allocating multiple channels
When backing up Oracle data, RMAN sends the data through a channel to a server session and finally to TSM. Using multiple channels in one backup or restore operation may increase performance.

D.1.2.2 RMAN backup with the diskratio option
The `diskratio` option directs RMAN to assign datafiles (only) to each backup set and spread them across the specified number of drives.

For example, assume that you use 10 disks, the disks supply data at 10 bytes/second, and the tape drive requires 50 bytes/second to keep streaming. You can set `diskratio 5` to spread the backup load across 5 disks for each backup set. If you set `filesperset` but not `diskratio`, then `diskratio` defaults to the same value as `filesperset`. If you specify neither parameter, `diskratio` defaults to 4.

RMAN compares the `diskratio` value to the actual number of devices involved in the backup and uses the lowest value. For example, if `diskratio` is 4 and the datafiles are on 3 disks, then RMAN attempts to spread the backup load for each set among 3 disks. The `diskratio` parameter is easier to use for datafile backups when your datafiles are striped, or they reside on separate disk spindles, and you do either of the following:

- Use a high-bandwidth tape drive that requires several datafiles to be multiplexed in order to keep the tape drive streaming.

- Make backups while the database is open and you want to spread the I/O load across several disk spindles in order to leave bandwidth for online operations.
D.2 TDP for Oracle Version 2.1.10

There is no specific option to influence performance — unless you are using the undocumented DSMO_DEBUG option. Remove DSMO_DEBUG from your RMAN allocate channel command to reduce the trace overhead.

D.2.1 TDP for Oracle Version 2.2.0

In this section we discuss two options you need to consider.

D.2.1.1 Multi-buffering of data transfers
The multi-buffering option of data transfers is, like the COMPRESSION parameter in the TSM client system option file, an option which is useful when the network load is high and network performance is poor.

More details about multi buffering can be found in C.1.5, “Multi-buffering of data transfers” on page 210.

We tested this option with different values — but in the end, we turned this option off, because we had no network problems and we detected an overhead in managing multiple buffers on a small system.

D.2.1.2 LAN-free data movement
TDP for Oracle is already enabled to use LAN-free data movement, this should bring a significant performance improvement for backup and restore operations.

However, the additional products to implement LAN free data movement were not yet available on AIX or Sun Solaris at the time of writing.

On the Windows NT platform, we tested an unsupported version and detected that backups are up to four times faster than the ordinary backups going over TCP/IP.

Note
Do not spread I/O over more than the minimum number of disks to keep the tape streaming. Otherwise, you increase restore time for a file without increasing performance.
Appendix E. TDP for IBM ESS for Oracle databases

In this appendix, we show an example of a backup and restore operation using Tivoli Data Protection (TDP) for IBM Enterprise Storage Server (ESS) for Oracle databases. We show how we set up each component and how the actual backup and restore commands were working. For further information and requirements, see Tivoli Data Protection for IBM ESS for Oracle Databases, SH26-4125-00.

Our setup uses the following components:

- Two RS/6000 with AIX 4.3.3.0
- ESS with LIC SC01206
- Oracle Version 8.1.5
- TDP for Oracle 2.1.0.10
- TDP for ESS for Oracle 1.1.0.0
- TSM server 3.7.3
- TSM client 4.1.1

E.1 IBM ESS FlashCopy

In this section, we briefly describe the FlashCopy feature of the IBM ESS. For further information, see Redbook Implementing ESS Copy Services on UNIX and Windows NT/2000, SG24-5757-00.

FlashCopy is a feature of the ESS that allows you to establish an identical and independent copy of a source ESS volume to a target ESS volume. The target volumes should be assigned to another physical machine. If the target volume is assigned to the same machine where the source volumes are, the operating system may have problems identifying the new disk volume, as it is an exact copy of the original disk. Things like unique disk identifiers, size of the disk, the copied data itself will appear twice on the same machine, which may cause problems.

See Figure 59 for an example setup.
There are two requirements to allow a FlashCopy relation between two ESS volumes to be established:

- The volumes must be in the same LSS (logical subsystem). LSS is an internal ESS structure that groups certain ESS volumes together. For a description on LSS, see Appendix B, “Understanding logical subsystems” in the redbook Implementing ESS Copy Services on UNIX and Windows NT/2000, SG24-5757-00.

- The target volume must be equal to or greater in size than the source volumes. We recommend that you select target volumes with exactly the same size as the source volumes.

At a specific time, the FlashCopy relationship between source and target volumes will be established. This is known as the \( T_0 \) (time zero) copy. The process of establishing this \( T_0 \) copy takes only a few seconds. A bitmap will be created to identify the \( T_0 \) data on the source disk.

At the time that the FlashCopy is started, the target volume is basically empty. The background copy task copies data from the source to the target. The FlashCopy bitmap keeps track of which data has been copied from source to target. If an application wants to read some data from the target that has not yet been copied to the target, the data is read from the source; otherwise, the read is satisfied from the target volume. See Figure 60.
Before an application can update a track on the source that has not yet been copied, the track is copied to the target volume. After some time all the data is copied either by the background copy process or by write operation on either the source or the target volume.

**Note**

When establishing the FlashCopy, the data on the source volume must be in a consistent state. This can be achieved by stopping the application and flushing the data from memory to disk. If the application cannot be stopped, then the application itself must provide a feature to ensure that the data is consistent while the FlashCopy is being established and that it can recover from that state afterwards on the target side.
E.2 Overview of TDP for ESS

TDP for ESS takes advantage of the FlashCopy feature of the IBM ESS to make a copy of an Oracle database so it can perform an RMAN database backup on the database copy instead of the actual database. A separate machine called the backup machine, which has access to the database copy on FlashCopy devices, is used to perform the RMAN backup.

Because resources associated with a backup process are “off-loaded” onto a separate backup machine, the availability of databases running on production machines is increased. This also reduces any performance impact that is normally caused by backup processes on production databases.

TDP for ESS also employs the use of TDP for Oracle to backup databases to TSM. The TSM server can reside on the backup machine. Figure 61 shows a typical TDP for ESS configuration.

TDP for ESS includes programs that are executed on the production machine and on the backup machine. The FlashCopy is done by the program that executes on the production machine. The program that executes on the backup machine does the Oracle backup, using the file created by the FlashCopy.

Figure 61. Typical TDP for ESS configuration
E.3 Our lab setup and experiences

In this section, we will show you how we set up TDP for ESS and TDP for Oracle. Our setup was quite simple, and we did not have a large database. Our tests involved backups using TDP for Oracle only and TDP with ESS, and we used the same RMAN catalog for both types of backup. Our database is in ARCHIVELOG mode. We tested our recovery by deleting one of the data files. We will also discuss what problems we encountered when doing backup and recovery.

This section is not intended to replace the manual *Tivoli Data Protection for IBM ESS for Oracle Databases Installation and User’s Guide Version 1 Release 1*, SH26-4125-00. This section simply complements the manual. The manual is also available online at:

http://www.tivoli.com/support/storage_mgr/tivolimain.html

We will refer you to the manual when discussing our setup in detail. We also assume that the TSM Server and ESS are already installed and operational.

E.3.1 Setup and configuration

Here we outline the setup and configuration of the systems we used, including details of the system environment and customization of option files.

E.3.1.1 Environment

Our environment consists of the following as shown in Table 5.

<table>
<thead>
<tr>
<th>System</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production machine</td>
<td>auckland.ibm.com</td>
</tr>
<tr>
<td>Backup machine</td>
<td>swiss.ibm.com</td>
</tr>
<tr>
<td>ESS Server</td>
<td>vienna01.ibm.com</td>
</tr>
<tr>
<td>TSM Server</td>
<td>swiss.ibm.com</td>
</tr>
</tbody>
</table>

E.3.1.2 TDP for Oracle setup

See Chapter 5, “Configuration and setup of TDP for Oracle on UNIX” on page 61 to better understand the setup for TDP for Oracle.

The Oracle server must be installed on both the production and backup machines.
The following is a description of our test database environment:

- The operating system DBA user is oracle8 in the production and backup machine.
- The operating system group name is dba in the production and backup machine.
- The database name is tokyo and only exists in the production machine.
- The datafiles, redo logs, and control files are in an ESS volume. Note that the redo logs and control files need not be on the same volume.
- The archive logs are in a local directory.
- The catalog database (RMAN) exists in the production machine. The name was specified in the listener.ora file.
- On the production machine, there is a tnsnames entry for the tokyo database, and the catalog database.
- In the backup machine, only the catalog database has an entry in the tnsnames.ora. There is no entry for the tokyo database.
- The ORACLE_HOME and ORACLE_BASE is /orc8/app/oracle/product/815

Our dsm.opt file in /usr/tivoli/tsm/client/api/bin in the production machine (auckland) and in the backup machine (swiss) looks like the one below. Note that the dsm.opt file need not be in this directory, but it must be in the same directory on both machines.

```
servername swiss2
```

The stanza for swiss2 in the dsm.sys file in /usr/tivoli/tsm/client/api/bin in the production machine (auckland) looks like this:

```
servername swiss2
tcps  swiss.ibm.com
tcpp  1500
passwordaccess prompt
```
The stanza for swiss2 in the dsm.sys file in /usr/tivoli/tsm/client/api/bin in the backup machine (swiss) looks like this:

```
servername swiss2
commmethod tcpip
tcps 9.11.111.11
tcpp 1500
passwordaccess prompt
shmport 1510
```

The tcpip address 9.11.111.11 points to swiss.ibm.com.

The nodename used to back up the production database tokyo to TSM is auckland.

With this configuration, we are able to backup our tokyo database to RMAN in the production environment to test that TDP for Oracle works.

### E.3.1.3 TDP for ESS setup

For the installation, hardware, software, and environment requirements, please see the *Tivoli Data Protection for IBM ESS for Oracle Databases Installation and User's Guide*, SH26-4125-00.

By default, TDP for ESS uses directory /usr/tivoli/tsm/client/ba/bin. The root user will be the one performing the TDP for ESS backup and must have the following environment variables defined:

```
DSM_CONFIG=/usr/tivoli/tsm/client/ba/bin/dsm.opt
DSM_DIR=/usr/tivoli/tsm/client/ba/bin
```

Our dsm.opt file in /usr/tivoli/tsm/client/ba/bin on the production machine (auckland) and on the backup machine (swiss) look like this:

```
servername swiss
tracefile /tmp/tokyo.trc
traceflags api api_detail appl
```

The tracefile and traceflags parameters are optional, but recommended initially during the testing of TDP for ESS. It was very valuable to us because it helped us identify any errors during a backup.
The stanza for the server swiss in the dsm.sys file in /usr/tivoli/tsm/client/ba/bin in the production machine (auckland) looks like this:

```
servername swiss
tcps swiss.ibm.com
tcpp 1500
passwordaccess generate
schedmode prompt
```

The stanza for swiss in the dsm.sys file in /usr/tivoli/tsm/client/ba/bin on the backup machine (swiss) looks like this:

```
servername swiss
COMMmethod tcpip
TPPort 1500
TCPServeraddress 9.11.111.11
schedmode prompted
passwordaccess generate
shmport 1510
```

The tcpip address 9.11.111.11 points to swiss.ibm.com.

TDP for ESS requires a setup file for a database on the production machine. For database tokyo, we put all are setup and script files in /orc8/app/oracle/product/815/admin/tokyo/adhoc directory.

Our setup file (tokyo.ess) in the production machine (auckland) initially looks like this:
Appendix E. TDP for IBM ESS for Oracle databases

For a full description of parameters used, please see the Tivoli Data Protection for IBM ESS for Oracle Databases, Installation and User’s Guide, SH26-4125-00.

The following provides additional notes on some of the parameters above:

- **database_backup_script_file**
  
  This is the RMAN backup script that will be used for the backup. Our backup script will be shown later.

- **database_backup_type**
  
  Our value here is offline. This is the backup type that will be performed on the backup machine.

```bash
#-------------------------------------#
#=== Setup for the target database which will be backed up ===#
#
# target_database_home: /orc8/app/oracle/product/815
# target_database_sid: tokyo
# target_database_username: sys
# target_database_password: change_on_install
# target_database_parameter_file: /orc8/app/oracle/product/815/admin/tokyo/pfile/initokyo.ora
# target_database_password_file: /orc8/app/oracle/product/815/dbs/orapwtokyo
# target_database_suspend: yes
#
#=== Setup for the catalog database ===#
# catalog_database_connect_string: rman.world
# catalog_database_username: rman2
# catalog_database_password: rman
#
#======Setup for other database parameters========#
# database_backup_script_file: /orc8/app/oracle/product/815/admin/tokyo/adhoc/backup
# database_backup_msglog_file: /orc8/app/oracle/product/815/admin/tokyo/adhoc/rmanlog
# database_backup_type: offline
# database_backup_incremental_level: 0
#
#======Setup for ESS copy services=====#
# java_home_directory: /usr/jdk_base
# flashcopy_type: copy
# shark_copy_service_code: /home/ibm2105cli
# shark_username: root
# shark_password: vienna01
# shark_servername: vienna01.ibm.com
# shark_target_volume: 300FCA90 - -
```
• **shark_target_volume**
  
  We only used one volume for our test. If your database uses many ESS volumes for your datafiles, and other Oracle files, you should specify this parameter for each volume you use.

  We initially put in here the serial number of the FlashCopy device, followed by two dashes (`- -`). The dashes was updated later with the source volume serial number, and the size of the volume when the backup script was run on the production machine.

  You must note that the FlashCopy device must be accessible to the backup machine, but must not be known to the operating system when the backup process is running. When the backup process starts on the backup machine, TDP for ESS runs the Configuration Manager and makes the disks and filesystems known to the operating system, runs the backup, then removes the file system and disks from the operating system.

  We had problems because of this and the errors produced in the trace file were not that obvious.

  Our backup script file as specified in `database_backup_script_file` in the production machine looks like this:

  ```
  run {
    allocate channel t1 type 'sbt_tape' parms
    'ENV=(DSMO_NODE=auckland,DSMI_ORC_CONFIG=/usr/tivoli/tsm/client/api/bin/dsm.opt,
    DSMI_LOG=/orcl/app/oracle/product/815/admin/tokyo/adhoc,DSMO_OWNER=oracle8)';
    backup
    (database);
    release channel t1;
  }
  
  Here are some things to note on the backup script:
  
  • **TDP for Oracle environment variables**
    
    The environment variables used by channel t1 are what you would normally use for TDP for Oracle.
  
  • **DSMO_NODE**
    
    This is the TSM node name to be used to backup the database. This is the node name for TDP for Oracle and not TDP for ESS.
Appendix E. TDP for IBM ESS for Oracle databases

• DSMO_OWNER

You must specify the operating system DBA as the owner. In our case, it is oracle8. If you do not specify DSMO_OWNER, the owner of the files in the TSM server will be daemon. Because oracle8 does not own the files, root will be the only one who can do RMAN crosscheck and recovery. Specifying DSMO_OWNER will ensure that the DBA will have privileges to do crosschecks and recovery for the database.

• Backup (database)

The backup command must be split into two lines as in:

```bash
backup (database)
```

TDP for ESS modifies the Backup command in the script. Note that in the production machine, it modifies it to backup only the control files. In the backup machine, it modifies it to include the incremental parameter as specified in `database_backup_incremental_level`. If the `backup (database)` is in one line, the `(database)` part gets deleted in the backup machine.

Warning: We initially have our `backup (database)` on one line, and it caused the script to fail. The trace file was helpful in finding out the problem, and it showed how TDP for ESS parsed the backup command.

We suggest that the backup script be as simple as possible as TDP for ESS needs to parse and rewrite the RMAN commands.

E.3.2 Backing up the database using TDP for ESS

Before discussing our tests, let us discuss how we did our backups on the production and backup machine. You should also read the backup and restore chapter in the Tivoli Data Protection for IBM ESS for Oracle Databases Installation and User’s Guide manual SH26-4125-00. The backup chapter also discusses how you can automate the backup using the TSM scheduler.

We first ran the backup on the production machine which only backed up the control files and some other files. Then we ran the backup on the backup machine which does the actual backup of the database.

Note that you have to use root to run the TDP for ESS backups.
E.3.2.1 Backup on the production machine

Our backup script in the production machine looks like this:

```
essorcp /orc8/app/oracle/product/815/admin/tokyo/adhoc/tokyo.ess swiss
/tmp/tokyotmp
```

The following is an explanation of the command and its parameters:

- **essorcp**
  This is the TDP for ESS backup executable for the production machine.

- **setup filename**
  This is our setup file:

- **TSM node name of the backup machine**
  Because the node name was not explicitly specified in the dsm.sys of backup machine, it defaulted to the hostname. In our case the backup machine's node name is swiss. This is the TDP for ESS node name and not the TDP for Oracle node name.

- **temporary file name**
  A temporary file is created by essorcp which contains parameters that are needed by the backup machine to perform its backup. This file is backed up in the TSM server, and later on transported to the backup machine by the backup process. We put the temporary file in /tmp/tokyotmp.

From our observation, the backup process basically performs the following:

1. Makes an RMAN backup to TSM of the control files
2. Performs a FlashCopy of the source volumes
3. Backs up the backup script and configuration files to TSM
4. Creates a temporary file
5. Performs backup of the temporary file

E.3.2.2 Backup on the backup machine

We did not use a backup script on the backup machine, We ran our backup directly on the shell, and the command looks like this:

```
[root@swiss> /] $ essorcdb auckland /tmp/tokyotmp
```
The following is an explanation of the command and its parameters:

- essorcb
  This is the TDP for ESS backup executable for the backup machine.
- TSM node name of the production machine
  Because the nodename was not explicitly specified in the dsm.sys of production machine, it defaulted to the hostname. In our case the production machine’s node name is auckland.
- temporary file name
  The backup process restores the temporary file created by the backup process in the production machine on the files specified by this parameter. We kept the temporary file name the same as the one in the production machine. In our case, this is /tmp/tokyotmp.

From our observation, the backup process basically performs the following:
1. Makes the FlashCopy volumes and filesystems available to the operating system
2. Restores the temporary file that was created by the backup process in the production machine
3. Restores the backup script and configuration files
4. Restores the controlfile
5. Performs an RMAN offline backup of the database to TSM
6. Removes the filesystems and devices from the operating system

It also appears that the admin/tokyo branch was recreated on ORACLE_HOME or ORACLE_BASE when restoring the backup script and configuration file (inittokyo.ora). It is therefore recommended that ORACLE_HOME or ORACLE_BASE exists in the backup machine before running the backup command.

Here we encountered problems as mentioned in E.3.1.3, “TDP for ESS setup” on page 233. We recommend using the trace file to determine the cause of the problem. Remember to turn off tracing later.

E.3.3 TDP for ESS testing
In this section we discuss the tests we have made for backup and recovery.
E.3.3.1 Backup

We performed both backups of the database using normal online RMAN backups and TDP for ESS backup using the same RMAN backup script. Both type of backups used the same RMAN catalog database. From the point of view of RMAN, using TDP for ESS or just TDP for Oracle for our backups is immaterial. As far as RMAN is concerned, the backups were performed using 'sbt_tape' (a media manager), and they are treated the same way.

When using only RMAN to do backup, we use the command as follows using user oracle8 to perform the backup:

```
$ rman target / catalog rman2/rman@rman cmdfile
   '/orc8/app/oracle/product/815/admin/tokyo/adhoc/backup'
```

When using TDP for ESS backups, we use the procedure as discussed above in E.3.2, “Backing up the database using TDP for ESS” on page 237. We ran them from user root.

We also backed up the archive logs without using TDP for ESS. We used user oracle8 to do the backup. The RMAN backup command looks like this:

```
$ rman target / catalog rman2/rman@rman cmdfile
   '/orc8/app/oracle/product/815/admin/tokyo/adhoc/backuplog'
```

The backuplog file that does the archive logs backup looks like this:

```
sql 'alter system archive log current';
run {
   allocate channel t1 type 'sbt_tape' parms
      'ENV=(DSMO_NODE=auckland,DSMI_ORC_CONFIG=/usr/tivoli/tsm/client/api/bin/dsm.opt,
      DSMI_LOG=/orc8/app/oracle/product/815/admin/tokyo/adhoc,DSMO_OWNER=oracle8)';
   backup (archivelog like '/orc8/app/oracle/product/815/admin/tokyo/arch/%'
       channel t1 delete input);
   release channel t1;
}
```

We also tested the TDP for ESS backup essorcp on the production machine while updates were ongoing on a USERS tablespace. The FlashCopy phase was quick. It did slow down the updates for a few seconds during the flash copy phase, but overall, the impact on the database was very small.
Because the backup machine is independent of the production machine, there was no impact on the ongoing activity on the production machine when we ran the actual backup of the database on the backup machine using the *essorc* executable.

After the backups, we verified the backups using the RMAN crosscheck utility. Initially, all backups were invalidated and expired when using user oracle8 because the owner of the files in TSM is daemon. We were only able to do crosscheck when using user root. This was corrected when we modified the backup scripts to include DSMO_OWNER as discussed in E.3.1.3, “TDP for ESS setup” on page 233.

**E.3.3.2 Recovery**

The RMAN recovery procedure is the same regardless of whether you are using TDP for ESS or just plain RMAN backup. As pointed out in E.3.3.1, “Backup” on page 240, RMAN is concerned with the device type ‘sbt_tape’ which uses a media manager. In our case, it is TDP for ORACLE. TDP for ESS is transparent to RMAN.

Note that when using plain RMAN backup, we do online backup. When we use TDP for ESS, we do offline backup. This does not matter. The datafiles in the production machine can be more up-to-date than the flash copy datafiles at the time when the backup executable *essorc* was executed in the backup machine. As long as all the archive logs needed are available, we can do full database recovery.

To understand Oracle’s recovery concepts in more depth, please see the Oracle 8i Backup and Recovery Guide.

To prepare for our recovery testing, we did the following:

1. Make a current backup of the database using TDP for ESS while transactions are being made to tablespace USERS.
2. Back up the archive logs using regular RMAN backup.
3. Verify the availability of the backup using the RMAN crosscheck utility.
4. Apply more transactions to tablespace USERS.
5. Delete the datafiles for tablespace USERS.
6. If the USERS tablespace is still accessible after the datafiles were deleted because the data buffers are in use, apply more transactions to tablespace USERS.
In order for the DBA user (oracle8) to perform recovery, the TSM environment variable DSMO_OWNER must be specified for the channel in the backup script as discussed in E.3.1.3, “TDP for ESS setup” on page 233.

Our recovery script file, named recover, looked like this:

```
run {
  allocate channel t1 type 'sbt_tape' params
  'ENV=(DSMO_NODE=auckland,DSMI_ORC_CONFIG=/usr/tivoli/tsm/client/api/bin/dsm.opt,
  DSMI_LOG=/orc8/app/oracle/product/815/admin/tokyo/adhoc)';
  restore datafile '/hewdb/tokyo/users01.dbf';
  recover tablespace "USERS" delete archivelog;
  release channel t1;
}
```

We run the recovery as user oracle8. Here is the RMAN recovery command we used:

```
rman target / catalog rman2/rman@rman cmdfile
'/orc8/app/oracle/product/815/admin/tokyo/adhoc/recover'
```

In our first test, Oracle detected immediately that the datafile was lost, so we were not able to add more transactions to tablespace USERS. With this we did an open database recovery (database remains online). Below are the steps we performed:

1. Using sqlplus or svrmgrl, make the tablespace offline:

   ```
   alter tablespace users offline temporary
   ```

2. Run the RMAN recovery command:

   ```
rman target / catalog rman2/rman@rman cmdfile
'/orc8/app/oracle/product/815/admin/tokyo/adhoc/recover'
```

3. Using sqlplus or svrmgrl, make the tablespace online again:

   ```
   alter tablespace users online;
   ```
In our second test, the tablespace USERS was still available even after the datafile was deleted. We were able to add more transactions to the tablespace. Then we restarted the instance. The startup was not able to open database because the datafile was missing. So it remained in mount mode. Here, we did a close database recovery. Below were the steps we performed:

1. Make sure that the datafile is online by querying V$DATAFILE. In our case, the datafile was online, otherwise, we would have issued this command using sqlplus or svrmgrl:

   ```
   alter database datafile '/hewdb/tokyo/users01.dbf' online;
   ```

2. Run the RMAN recovery command:

   ```
   rman target / catalog rman2/rman@rman cmdfile
   '/orc8/app/oracle/product/815/admin/tokyo/adhoc/recover'
   ```

3. Using sqlplus or svrmgrl, open the database:

   ```
   alter database open;
   ```

   Our recovery tests were able to recover the database to its latest committed transaction.

### E.3.4 Summary

In summary, TDP for ESS reduces resource contention by offloading the backup process to a backup machine. The backup process can be done offline in the backup machine which gives better performance. The production database remains online and its activity is not affected by a backup process.

It is therefore recommended to use TDP for ESS to perform database backup on databases which require maximum database availability.

If you have to do tablespace backup, the regular RMAN tablespace backup can be used for smaller tablespaces. For larger tablespaces, TDP for ESS can be used when it becomes available. As of this writing, TDP for ESS version 1.1 only supports database backup.

For backing up archive logs, the only option as the time of writing is to use regular RMAN commands.
Appendix F. TDP for Oracle using TSM LAN-free setup

In this appendix we show how we configured TDP for Oracle to send backups via the Storage Area Network (SAN).

These are the assumptions we used:

- The TSM server is already configured: For example, libraries, drives, devclasses, stgpools, and server-to-server have all been defined.
- The TSM client has Fibre Channel card installed, SAN zoning is set (if required), the client can “see” the tape drives via the operating system, and can perform regular LAN backups.

For information on how to get to the point where the tape drives are visible and the TSM server is configured, see the redbook *Using Tivoli Storage Manager in a SAN environment*, SG24-6132-00. This book also discusses the reasons for using a SAN to perform backups and the benefits to be gained.

F.1 TSM configuration for LAN-free setup

Support for the LAN-free setup is wholly contained within the TSM API. No additional code is needed in any of the products (DB2, TDP for Oracle) except to use the TSM API. The TSM Storage Agent software must be installed on the same machine as the product performing the LAN-free backup.

These are the steps necessary:

1. Define new management class for LAN-free backups.
2. Download and install storage agent.
3. Modify dsmsta.opt.
4. Check adsmscsi.
5. Define drive mappings.
6. Define server for storage agent on TSM server.
7. Run dsmsta setstorageserver on storage agent.
8. Install storage agent as service.
F.1.1 Define new management class for LAN-free backups

On the TSM server we already had a domain with a management class that had the correct policy retention for Oracle backups. So instead of going through the steps to define a new management class, we just copied an existing one, and since it already had the correct retention settings, the only thing we needed to do was to update the destination to be the SAN attached storage pool. Currently, the LAN-free setup is only supported going to SAN attached tape devices.

Here is the command we used to copy the management class:

```
tsm: BRAZIL>copy mgmtclass api_domain api_policy api_mgmtclass api_lanfree
ANR1523I Management class API_MGMTCLASS copied to class API_LANFREE in policy
domain API_DOMAIN, set API_POLICY.
```

We then need to update the backup copy group for this new api_lanfree management class so the destination is the SAN attached tape storage pool:

```
tsm: BRAZIL>update copygroup api_domain api_policy api_lanfree dest=3570san
ANR1532I Backup copy group STANDARD updated in policy domain API_DOMAIN, set
API_POLICY, management class API_LANFREE.
```

The only thing left to do is activate the policy set so this new management class can be used:

```
tsm: BRAZIL>activate policy api_domain api_policy
Do you wish to proceed? (Yes (Y)/No (N)) y
ANR1514I Policy set API_POLICY activated in policy domain API_DOMAIN.
```

Now we can check the backup copygroup for this new management class in the active policy. We can see that the retention is correct. Versions exists is set to one, the rest of the retention settings are set to zero. One of the most important things is that the copy destination is to the SAN attached tape storage.
To make sure that our Oracle backups go to this new management class, in the client options file (dsm.opt) we added this include statement:

```plaintext
include * api_lanfree
```

### F.1.2 Download and install TSM Storage Agent

The TSM Storage Agent is a key piece in doing LAN-free backups. The TSM Storage Agent is a scaled down version of the TSM server. Instead of the TSM API client sending its backups straight to the TSM server like it does in a regular LAN backup, the TSM API client sends the backup via NamedPipes to the TSM Storage Agent. The TSM Storage Agent then sends the backup data directly to the tape storage across the SAN. It also communicates with the TSM server using TCP/IP so that the data objects that the storage agent sends across the SAN are updated in the TSM server’s database.

The version 4 release 1 storage agent software for windows can be downloaded from:

```plaintext
```

The package name that was current at the time this was written was IP22268_StorageAgent.exe. Once downloaded to the client machine, we just double clicked the executable to install it. We accepted all the defaults and it installed into the directory `c:\program files\tivoli\tsm\storageagent`.

After installing the software, we rebooted the machine.
F.1.3 Modify dsmsta.opt

In the ...\storageagent directory we opened the DSMSTA.OPT file using a text editor and inserted these values. This is the minimum that is needed. These values are for a Windows environment.

The storage agent is a stripped down version of the TSM server, so this DSMSTA.OPT file is equivalent to the DSMSERV.OPT file. Therefore, the COMMmethod value is not selecting the communication method (like a client options file), but rather enabling those two communications method, just like you could do in the DSMSERV.OPT on a TSM server installation.

The DEVCONFIG value specifies the filename to use when running the DSMSTA SETSTORAGESERVER command.

The ENABLEALLCLIENTS option is an undocumented option which enables clients not yet supported to utilize the LAN-free setup:

```
COMMmethod TCPIP
COMMmethod NAMEDPIPE
DEVCONFIG devconfig.txt
ENABLEALLCLIENTS yes
```

F.1.4 Check adsmscsi

After rebooting, we checked that the TSM device driver AdsmScsi was running by running the Windows command `net start`:

```
C:\Program Files\Tivoli\TSM\storageagent>net start adsmscsi
The requested service has already been started.
More help is available by typing NET HELPMSG 2182.
```

In Windows 2000, you check Adsmscsi differently than for Windows NT. From Device Manager in Windows 2000, you must change the view to display hidden devices (Figure 62).
After selecting the view to show hidden devices, an entry **Non-Plug and Play Drives**, which you can open, will show up. When you open this window, you will see the **AdsmScsi** device (Figure 63).
Right-clicking **AdsmScsi** and selecting **Properties** presents the next screen (Figure 64).

![AdsmScsi Properties](image)

*Figure 64. AdsmScsi Properties*

One rather confusing thing is the tsmscsi.exe file located in the storageagent directory. This is **not** an updated device driver for use with TSM. This is a utility that enables and disables Windows 2000 support with the AdsmScsi device driver. When you use this utility to disable Windows 2000 support, the utility changes the startup mode for AdsmScsi to Demand. When you enable support, it changes the startup mode for AdsmScsi to Boot. When you install the storage agent, AdsmScsi defaults to a startup mode of Boot.

**F.1.5 Define drive mappings**

We then ran the TSM executable TSMDLST.EXE located in the \StorageAgent directory to determine how the storage agent “sees” the SAN attached tape library. From this output, we see that there are two tape drives.
From the TSM server administrative client, we run the command QUERY DRIVE to determine how the TSM server ‘sees’ the san attached tape library.

In a SAN, hosts access the same storage, but they may show up differently. The drive mapping makes the association between how the storage agent ‘sees’ the tape drives, with how the TSM server ‘sees’ the tape drives. Using the output from the TSMDLST.EXE and QUERY DRIVE, we will now define a drive mapping for these two drives.

F.1.6 Define server for storage agent on TSM server

Using the storage agent name that was specified during the definition of the drive mappings, we will now use the command DEFINE SERVER from the TSM administrative command line, to configure server to server communications on the TSM side. Setting the servername, serverpassword, serverhladdress, and serverlladdress had already been done. The command QUERY STATUS can be used to check if this is the case:
It is very important that you use the same server name used in this command that was used to define the drive mappings.

**F.1.7 Run dsmsta setstorageserver on storage agent**

From the `...\storageagent` directory, we need to run the DSMSTA SETSTORAGESERVER command to complete the server to server communications on the storage agent.

The values for MYNAME and MYPASSWORD come from the define server command in the previous section. The other values are for the TSM server that we are communicated with. These values can be viewed with a QUERY STATUS.

```
C:\Program Files\Tivoli\TSM\storageagent\dsmsta setstorageserver myname=jamaica_sta mypassword=stapassword servername=brazil serverpassword=brazil hladdress=193.1.1.11 lladdress=1500
ANR0900I Processing options file C:\PROGRA1\Tivoli\TSM\STORAGE1\dsmsta.opt.
Tivoli Storage Manager for Windows NT
Version 4, Release 1, Level 3.0
Licensed Materials - Property of IBM
5698-TSM (C) Copyright IBM Corporation 1999,2000. All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corporation.
ANR1432I Updating device configuration information to defined files.
ANR1433I Device configuration information successfully written to devconfig.txt.
ANR2119I The SERVERNAME option has been changed in the options file.
ANR0467I The setstorageserver command completed successfully.
```
F.1.8 install storage agent as service

To install the storage agent as a service, we use the INSTALL.EXE executable located in the ...\storageagent directory. Do NOT modify the value for the service name. You may modify the location of the dstasvc.exe file if you installed the storageagent software to a different directory.

After installing the service, start it with net start:

```bash
C:\Program Files\Tivoli\TSM\storageagent>install "TSM Storage Agent" "c:\program files\tivoli\tsm\storageagent\dstasvc.exe"
Service installed
C:\Program Files\Tivoli\TSM\storageagent>net start "TSM Storage Agent"
The TSM Storage Agent service is starting.
The TSM Storage Agent service was started successfully.
```

If you receive an error message that states that the service could not start, but did not return an error, it is most likely caused by specifying a service name other that the one listed above. In this case, use the REMOVE.EXE located in the ...\storageagent directory to remove the service, reboot your machine, and follow the instructions above like you should have done in the first place.

One the service is started, you can run a QUERY SESSION from the TSM server and see the server to server sessions that are ALWAYS running.

F.1.9 Specify enablelanfree in client options file (dsm.opt)

Now that all the previous steps have been completed, the only thing left to do is specify ENABLELANFREE YES in the dsm.opt file. Also you should have these backups going straight to the SAN attached library pool, by specifying the management class with an include statement. Our dsm.opt file looks like this:

```bash
commmethod tcpip
tcpport 1500
tcpserveraddress 193.1.1.11
nodename jamaica_oracle
passwordaccess generate
enablelanfree yes
include * api_lanfree
```
F.2 Running a backup and verifying that LAN-free setup is working

There is no difference in the backup commands for the LAN-free setup. We chose a simple RMAN backup to verify that the LAN-free setup is working.

After executing the command we connected to the storage agent using a TSM administrative client. Changed the TCPSERVERADRESS for the administrative client to be the IP address of the machine running the storage agent. We had the TSM administrative client installed on the same machine as the storage agent, so we just used localhost as our IP address. The admin name and password is the same as your TSM server name and password.

Query session shows that the LAN-free backup is working. This is indicated by the named pipe session that is sending data to the storage agent. If the LAN-free backup was not working, either the backup would fail or you would see a session on the TSM server that was sending the data:

```
C:\Program Files\Tivoli\TSM\baclient>dsmadmc -tcps=localhost
Tivoli Storage Manager
Command Line Administrative Interface - Version 4, Release 1, Level 2.12
(C) Copyright IBM Corporation, 1990, 1999, All Rights Reserved.
Enter your user id: admin
Enter your password: *****
Session established with server JAMAICA_STA: Windows NT
Server Version 4, Release 1, Level 3.0
Server date/time: 03/30/2001 15:48:45 Last access: 03/30/2001 15:37:43
tsm: JAMAICA_STA>q session
   Sess Comm.  Sess Wait  Bytes Bytes Sess Platform Client Name
Number Method State Time  Sent  Recvd Type
------ ------ ------ ------ ------- ------- ----- -------- ---------------------
   1 Tcp/Ip Start 0 S   16.5 K   23.2 K    Server- JAMAICA_STA
   3 Tcp/Ip Start 0 S   68.0 K   92.0 K    Server- JAMAICA_STA
   7 Tcp/Ip Start 0 S  128.9 K  150.6 K    Server- JAMAICA_STA
  12 Tcp/Ip Start 0 S  190.6 K  206.5 K    Server- JAMAICA_STA
 141 Tcp/Ip Start 0 S   31.1 K   31.9 K    Server- JAMAICA_STA
 167 Named RecvW 7 S   349  20.0 M Node DB2/NT JAMAICA_ORACLE Pipe
 169 Tcp/Ip Run 0 S    18.1 K    472 Admin WinNT ADMIN
```
F.3 Additional considerations

We decided that our log files should just use the LAN and not go over the SAN. It was not worth a tape mount to send a 1 Mb file via the SAN. To facilitate this, we used the format option in the RMAN script that backs up the archive redo logs:

```
sql 'alter system archive log current' ;
run {
    allocate channel t1 type 'sbt_tape' parms 'ENV=( DSMO_AVG_SIZE=1)';
    allocate channel t2 type 'sbt_tape' parms 'ENV=( DSMO_AVG_SIZE=1)';
    backup filesperset 20
        format '%d/LOGS/%t/%s/%p'
        archivelog all
        delete input ;
    release channel t1 ;
    release channel t2 ;
}
```

Now that the logs are given a different LL_NAME than the database objects, we can insert an include statement into our options file that sends just the log files to a disk storage pool. We put the include statement `include *LOGS* api_mgmtclass` below the `include *` statement so that it matches the logs first:

```
commmethod TCPIP
tcpport 1500
tcpserveraddress 193.1.1.11
nodename jamaica_oracle
passwordaccess generate
enablelanfree yes
include * api_lanfree
include *LOGS* api_mgmtclass
```
Appendix G. Using the additional material

This redbook also contains additional Web material. See the appropriate section below for instructions on using or downloading each type of material.

G.1 Locating the additional material on the Internet

The CD-ROM, diskette, or Web material associated with this redbook is also available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

ftp://www.redbooks.ibm.com/redbooks/SG246249

Alternatively, you can go to the IBM Redbooks Web site at:

ibm.com/redbooks

Select the Additional materials and open the directory that corresponds with the redbook form number.

G.2 Using the Web material

The additional Web material that accompanies this redbook includes the following:

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBA_rman.tar.Z</td>
<td>Tar file of setup + script files</td>
</tr>
</tbody>
</table>

G.2.1 System requirements for downloading the Web material

The following system configuration is recommended for downloading the additional Web material.

- **Hard disk space:** 35 Kb
- **Operating System:** Windows 98/
G.2.2 How to use the Web material

The material consists of a subdirectory (folder). Copy the contents of the Web material from this folder.

**Tar zip format:** DBA_rman.tar.Z

Run:

```
uncompress DBA_rman.tar.Z
tar -xvf DBA_rman.tar
```

This will create a directory `/tmp/DBA_rman` with eight files in it. To install the files, simply invoke:

```
/tmp/DBA_rman/DBA_first_setup
```

The `ksh` program will create `/usr/local/oracle ...` directories and load the `ksh` and configuration files.
Appendix H. Special notices

This publication is intended to help Oracle Database Administrators and Tivoli Storage Manager Administrators to manage their Oracle backup and recovery using TDP for Oracle. The information in this publication is not intended as the specification of any programming interfaces that are provided by Tivoli Storage Manager or TDP for Oracle. See the PUBLICATIONS section of the IBM Programming Announcement for Tivoli Storage Management for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee

© Copyright IBM Corp. 2001
that the same or similar results will be obtained elsewhere. Customers
attempting to adapt these techniques to their own environments do so at their
own risk.

Any pointers in this publication to external Web sites are provided for
convenience only and do not in any manner serve as an endorsement of
these Web sites.

The following terms are trademarks of the International Business Machines
Corporation in the United States and/or other countries:

<table>
<thead>
<tr>
<th>Term</th>
<th>Trademark Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>e (logo)®</td>
<td>Redbooks®</td>
</tr>
<tr>
<td>IBM ®</td>
<td>Redbooks Logo®</td>
</tr>
<tr>
<td>AIX</td>
<td>Lotus</td>
</tr>
<tr>
<td>AS/400</td>
<td>Lotus Notes</td>
</tr>
<tr>
<td>Cross-Site</td>
<td>Netfinity</td>
</tr>
<tr>
<td>Current</td>
<td>NetView</td>
</tr>
<tr>
<td>DB2</td>
<td>Notes</td>
</tr>
<tr>
<td>Domino</td>
<td>RS/6000</td>
</tr>
<tr>
<td>DPI</td>
<td>S/390</td>
</tr>
<tr>
<td>Enterprise Storage Server</td>
<td>SP</td>
</tr>
<tr>
<td>FlashCopy</td>
<td>System/390</td>
</tr>
</tbody>
</table>

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere., The Power To Manage., Anything.
Anywhere., TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet
Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli
Systems Inc., an IBM company, in the United States, other countries, or both.
In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli
A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other
countries.

Java and all Java-based trademarks and logos are trademarks or registered
trademarks of Sun Microsystems, Inc. in the United States and/or other
countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of
Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United
States and/or other countries and is used by IBM Corporation under license.
ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.
Appendix I. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

I.1 IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 265.

- Using ADSM to Back Up Databases, SG24-4335-00
- Tivoli Storage Management Concepts, SG24-4877-00
- Getting Started with Tivoli Storage Manager: Implementation Guide, SG24-5416-00

I.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at ibm.com/redbooks for information about all the CD-ROMs offered, updates and formats.

<table>
<thead>
<tr>
<th>CD-ROM Title</th>
<th>Collection Kit Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM System/390 Redbooks Collection</td>
<td>SK2T-2177</td>
</tr>
<tr>
<td>IBM Networking Redbooks Collection</td>
<td>SK2T-6022</td>
</tr>
<tr>
<td>IBM Transaction Processing and Data Management Redbooks Collection</td>
<td>SK2T-8038</td>
</tr>
<tr>
<td>IBM Lotus Redbooks Collection</td>
<td>SK2T-8039</td>
</tr>
<tr>
<td>Tivoli Redbooks Collection</td>
<td>SK2T-8044</td>
</tr>
<tr>
<td>IBM AS/400 Redbooks Collection</td>
<td>SK2T-2849</td>
</tr>
<tr>
<td>IBM Netfinity Hardware and Software Redbooks Collection</td>
<td>SK2T-8046</td>
</tr>
<tr>
<td>IBM RS/6000 Redbooks Collection</td>
<td>SK2T-8043</td>
</tr>
<tr>
<td>IBM Application Development Redbooks Collection</td>
<td>SK2T-8037</td>
</tr>
<tr>
<td>IBM Enterprise Storage and Systems Management Solutions</td>
<td>SK3T-3694</td>
</tr>
</tbody>
</table>

I.3 Other resources

These publications are also relevant as further information sources:

- Oracle 8i Recovery Manager User's Guide and Reference, A76990-01
- Oracle 8i Backup and Recovery Guide, A76993-01
- Tivoli Data Protection for IBM ESS for Oracle Databases Installation and User's Guide Version 1 Release 1, SH26-4125-00
I.4 Referenced Web sites

These Web sites are also relevant as further information sources:

  Main Tivoli Storage Manager Web site

  TDP for Oracle Web site
How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** [ibm.com/redbooks](http://ibm.com/redbooks)

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

  - In United States or Canada: pubscan@us.ibm.com
  - Outside North America: pubscan@ibmlink.ibm.com

- **Telephone Orders**

  - United States (toll free): 1-800-879-2755
  - Canada (toll free): 1-800-IBM-4YOU

- **Fax Orders**

  - United States (toll free): 1-800-445-9269
  - Canada: 1-403-267-4455

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

---

**IBM Intranet for Employees**

IBM Redbooks fax order form

Please send me the following:

<table>
<thead>
<tr>
<th>Title</th>
<th>Order Number</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

First name | Last name 

Company 

Address 

City | Postal code | Country 

Telephone number | Telefax number | VAT number 

☐ Invoice to customer number 

☐ Credit card number 

Credit card expiration date | Card issued to | Signature 

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.
## Abbreviations and acronyms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM</td>
<td>International Business Machines Corporation</td>
</tr>
<tr>
<td>ITSO</td>
<td>International Technical Support Organization</td>
</tr>
<tr>
<td>RMAN</td>
<td>Oracle Recovery Manager</td>
</tr>
<tr>
<td>TDP</td>
<td>Tivoli Data Protection</td>
</tr>
<tr>
<td>TSM</td>
<td>Tivoli Storage Manager</td>
</tr>
</tbody>
</table>
Index

A
administrative client 34, 199, 203
adsmscsi 248
aobpswd 66
API 245
Tivoli Storage Manager 4
API client 199
API sample program 207
Application Program Interface (API) 33, 34
archive log 243
archive object 38
archived redo log backup, Oracle8 83, 165
ARCHIVELOG 73
archivelog 205
ARCHIVELOG mode 20
Automatic deletion 101

B
Backup
Consistent 78
How often 90
Inconsistent 78
obsolete 94, 180
orphaned 95
backup 76
redundant 94
backup copygroup 246
backup strategy 21
backup techniques 25
backup windows 24

C
cmdfile 90, 171
commands
Oracle
rman 69, 153
commtimeout 202
Communication Channel 70
COMPRESSION 223
Configuration parameters 15
consistent restore, Oracle8 107, 189
console mode 199
Control files 15, 18
copygroup 41
archive copygroup 42
backup copygroup 41
crosscheck command 92, 214
Crosscheck utility 101

D
Data dictionary 13
data dictionary 10
data object 33, 37
data storage 12
database 9
Tivoli Storage Manager backup and restore 5
database administrator 12
database administrator (DBA) 34
database backup
partial database backup (DBA) 28
database export 27
Database recovery
inconsistent 107, 189
database recovery 189
consistent 107, 189
datafiles 17
deletion script 205
disk mirroring 26
diskratio 224
drive mappings 250
DRM 4
dsierror.log 204
dsm.opt 199, 253
DSM_CONFIG 200
DSM_DIR 36
DSM_LOG 36, 205
DSMO_DEBUG 225
DSMO_NODE 236
DSMO_OWNER 237, 242
dsmsched.log 176, 204
dsmsta 252
dsmsta.opt 248
dsoerror.log 205
Duplex Backup 216
Duplex Copy 209
enablelanfree 253
Enterprise Backup Utility 211
Enterprise Storage Server (ESS) 227
environment variables 36, 62, 204, 205
Backing Up Oracle using Tivoli Storage Management

DSMI_CONFIG 36
ESS 227
Configuration Manager 236
essorcb 239
essorcp 238
expiration 44
Export utility 73

F
Fibre Channel 245
filespace 208
FlashCopy 227
Full backup 77

I
Image copies 80
include statement 247
inconsistent restore, Oracle8 107, 189
Incremental backup 29
Cumulative 80
multilevel 78
Noncumulative 79
Indexes 10
init.ora 216
initialization files 15
Instance 16

L
LAN-free 30, 225, 245
LAN-free backup 30, 210
level 0 backup 87
libobk 70
List utility 181
Log File Backup 29
Log Files 13
checkpoint process 14
logfiles 204
logical subsystem 228
LSS 228

M
management class 40, 246
multi-buffering 210, 216, 225

N
nodename 202, 213
nomount mode 111

noncumulative backup, Oracle8 79, 161

O
Offline Backup 27
Online Backup 27
Oracle
archived redo log backup, Oracle8 83, 165
redo log backup, Oracle8 83, 165
rman command 69, 153
Oracle 8i 16
overview 16
Oracle Parallel Server (OPS) 19
oragent.log 205

P
Partial Database Backup 29
partitioning 13
partitioning option 13, 18
password 202
Password file 18
PASSWORDACCESS 66
performance 223
bandwidth 224
Planning considerations 21
policy domain 40
policy management 40
policy set 41, 246

Q
query command 203

R
readme file 205
recovery 107, 189
recovery catalog 72, 156, 221
recovery catalog, RMAN 69, 153
recovery points 25
Redo Log 17
redo log 179
redo log backup, Oracle8 83, 165
redo logs 108, 190
restore process 107, 189
RMAN 69
allocate channel 64
architecture 69, 153
archived redo log backup 83, 165
channels, communication 69, 153
communication channel 69, 153
consistent restore 107, 189
database recovery 107, 189
inconsistent restore 107, 189
multiple channels 224
noncumulative backup 79, 161
recovery catalog 69, 153
redo log backup 83, 165
restore types 107, 189
rman command 69, 153
rman scripts 69, 153
target database 69, 153
RMAN backup 204, 241
RMAN catalog 210, 231
RMAN command 206
rman command, Oracle8 69, 153
runlevel 89

S
SAN 210
SAN attached 246
sbtio.log 204
schedule
  client polling 173
  server prompted 173
scheduler service 173
scripts, RMAN 69, 153
Simulated Incremental 29
split mirror 30
SQL 'select' 37, 39
SQL select command 218
SQL*Net 89, 171
storage agent 245, 247, 253
Storage Area Network (SAN 245
storage pool 43
Structured Query Language (SQL) 9

T
Table 9
tablespace 11, 16, 108, 190
tags 186
Target Database 70
target database, Oracle8 RMAN 69, 153
target volume 227
TDP application clients 4
TDP for Domino
  command-line interface 7
TDP for Oracle 76, 211, 230

performance 223
setup 231
TDPO.OPT 209
TDPOCONF 209
tdpoconf 214
TDPOSYNC 209
TDPOSync 220
time zero 228
T0 copy 228
Tivoli Data Protection for applications 4
Tivoli Disaster Recovery Manager 4
Tivoli Space Manager 4
Tivoli Storage Management 3
Tivoli Storage Manager
  administrator 3
  API 4
  backup/archive client 5
  database backup and restore 5
  DRM 4
  HSM 4
  server 3
Tivoli Storage Manager backup/archive client 4
tracefile 201, 233
traceflag 201, 233
tracing 224, 239
troubleshooting 199
TSM administrator 35
TSM API 62
TSM client 33
  configuration 36
TSM filespace 208
  deleting 208
TSM object 206
  inconsistent state 206
TSM Storage Agent 210, 247
tsmcsi.exe 250
IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.

- Use the online Contact us review redbook form found at ibm.com/redbooks
- Fax this form to: USA International Access Code + 1 845 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

<table>
<thead>
<tr>
<th>Document Number</th>
<th>SG24-6249-00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Redbook Title</td>
<td>Backing Up Oracle Using Tivoli Storage Management</td>
</tr>
</tbody>
</table>

Review

What other subjects would you like to see IBM Redbooks address?

Please rate your overall satisfaction:

- Very Good
- Good
- Average
- Poor

Please identify yourself as belonging to one of the following groups:

- Customer
- Business Partner
- Solution Developer
- IBM, Lotus or Tivoli Employee
- None of the above

Your email address:

The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.

Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.

Questions about IBM’s privacy policy?

The following link explains how we protect your personal information.

ibm.com/privacy/yourprivacy/
Backing Up Oracle using Tivoli Storage Management
Backing Up Oracle
Using Tivoli Storage Management

This IBM Redbook is designed to assist Oracle Database administrators and system/storage administrators with Oracle backup solutions using Tivoli Storage Manager V4.1. The primary tool we used for backup and recovery is Tivoli Data Protection (TDP) for Oracle V1.1, and the book covers setup and configuration of TDP as well as day-to-day management examples. Operating environments covered by the project are AIX, Sun Solaris, and Microsoft Windows 2000. We provide an overview of relational database management systems and how to plan for protecting them.

This book will help you install, tailor, and configure Tivoli Storage Manager and Tivoli Data Protection for Oracle on UNIX and Microsoft Windows 2000 platforms in order to accomplish backup and restore of Oracle8i databases. RMAN commands and setup are covered in detail, and emphasis is placed on practical recovery scenarios.

Also featured are new solutions utilizing Tivoli’s hardware integration features for IBM’s ESS intelligent storage subsystem, as well as an update on using TDP for Oracle V2.2 new features.

Note: This redbook replaces relevant sections of the earlier redbook, Using ADSM to Back Up Databases, SG24-4335-03, by updating and re-branding the content of that book.

SG24-6249-00
ISBN 0738421855