VERITAS NetBackup™ 3.4 for Oracle Advanced BLI Agent

System Administrator's Guide

Solaris and HP

June 2000
100-001475

VERITAS

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

Copyright

Copyright © 2000 VERITAS Software Corporation. All rights reserved. VERITAS is a registered trademark of VERITAS Software Corporation. The VERITAS logo, VERITAS NetBackup, and VERITAS NetBackup BusinesServer are trademarks of VERITAS Software Corporation. All other trademarks or registered trademarks are the property of their respective owners.

Portions of this software are derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm. Copyright 1991-92, RSA Data Security, Inc. Created 1991. All rights reserved.

Printed in the USA, June 2000.

VERITAS Software Corporation
1600 Plymouth St.
Mountain View, CA 94043
Phone 650–335–8000
Fax 650–335–8050
www.veritas.com

Contents

Preface ..... **ix**
Audience x
Organization x
Related Documents xi
Conventions xii
Type Style xii
Notes and Cautions xii
Key Combinations xii
Command Usage xii
Getting Help xiii

Chapter 1. Introduction **1**
Features of NetBackup for Oracle on UNIX 2
Terminology for NetBackup for Oracle on UNIX 4
NetBackup Terms 4
Oracle Terms 5
Technical Overview of NetBackup for Oracle on UNIX 9
How Does the Oracle8 Recovery Manager Work? 10
How Does NetBackup for Oracle on UNIX Work? 10
Incremental Backup of Oracle Databases 12
Proxy Copy and Block-Level Incremental (BLI) Backup 12
Proxy Copy 12
Block-Level Incremental (BLI) Backup 13
Storage Checkpoint 14

How Does Proxy Copy and BLI Backup Work with NetBackup Advanced BLI Agent? 14
Sequence of Operation 20

Chapter 2. Installation 23
Installation Requirements 24
Installation Prerequisites 24
Install NetBackup Advanced BLI Agent 25
Remote Installation of NetBackup Advanced BLI Agent 25
Local Installation of NetBackup Advanced BLI Agent 31
Install Oracle8 Recovery Manager 34
Linking Oracle 8 with NetBackup Advanced BLI Agent 34
Linking Rollback Procedure 36
Install Oracle8 Recovery Catalog 37

Chapter 3. Configuration 39
Configure Media Manager 40
Add Oracle Classes to NetBackup 40
Expiration of Backup Files 41
Expiration of Backup Images from the NetBackup Catalog Using Retention Level 41
Environment Variables in a Scheduled Backup 41
Example Scripts 42
Maximum Jobs per Client Global Attribute 42
NetBackup Administration - Java Interface 43
`xbpadm` Interface 44
NetBackup Administration - Windows NT/2000 Interface 45
Class Configuration Procedures 46
NetBackup Administration - Java Interface 46
`xbpadm` Interface 58
NetBackup Administration - Windows NT/2000 Interface 64

Configure Run-Time Environment 76
Environment Variables Set Up by NetBackup for Oracle on UNIX 77
Environment Variables Set Up by a User 78
Multiple Stream Configuration Example 79
Create bp.conf 81
Create Scripts 82
NetBackup for Oracle Scripts 82
Create NetBackup for Oracle Script for Recovery Manager 82
RMAN Script 98
Test NetBackup for Oracle on UNIX Configuration Settings 112
NetBackup Administration - Java Interface 112
xbpadm Interface 114
NetBackup Administration - Windows NT/2000 Interface 115

Chapter 4. Using NetBackup for Oracle on UNIX 117
Maintaining a Recovery Catalog 118
Querying the Recovery Catalog 121
Performing a Backup 122
Automatic Backup of an Oracle Class 122
Manual Backup of an Oracle Class 122
User Directed Backup From the Client 123
Using xbp to Perform Backups 123
Executing the NetBackup for Oracle Script on the Client 124
Executing the rman Command on the Client 124
Browsing Backups 125
Using xbp to Browse Backups 125
Using the Recovery Catalog to Browse Backups 126
Using bplist to Browse Backups 126
Performing a Restore 127
Using xbp to Restore Backups 127

Executing an Oracle Script From a Command Line 128
rman Command 128
Alternate Client Restores 129
Server Configuration 129
Alternate Client Configuration 129
Performing Other RMAN Actions 131

Chapter 5. Troubleshooting 133
NetBackup and NetBackup for Oracle on UNIX 134
NetBackup Advanced BLI Agent Logs 134
NetBackup Server Reports 136
Oracle8 Recovery Manager Utility Logs 136
Troubleshooting Procedure 137
Backup or Restore Errors 138
Check the Logs to Determine the Source of the Error 138
Troubleshoot Each Stage of the Backup or Restore 139

Index 143

Preface

This guide describes how to install, configure and use VERITAS NetBackup for Oracle with Advanced BLI Agent on a UNIX platform. In this guide, VERITAS NetBackup for Oracle on UNIX with Advanced BLI Agent is referred to as either NetBackup for Oracle on UNIX or as NetBackup Advanced BLI Agent.

For specific information about the NetBackup server software, refer to:

- *NetBackup System Administrator's Guide - UNIX*, if you have a UNIX server.
- *NetBackup System Administrator's Guide - Windows NT/2000*, if you have a Windows NT/2000 server.

This document is the same as `NetBackup_AdminGuide_ProxyCopy.pdf` distributed with the NetBackup Advanced BLI Agent software.

For instructions on to configuring the VERITAS NetBackup for Oracle Advanced BLI Agent for Backups Without RMAN to back up and restore Oracle databases, refer to `NetBackup_AdminGuide_BLI.pdf`. This is the online version of the *NetBackup for Oracle Advanced BLI Agent System Administrator's Guide for Backups Without RMAN*.

Audience

This guide is intended for the:

- Oracle database system administrator responsible for configuring and using the Oracle8 Recovery Manager to back up and restore Oracle databases.
- NetBackup system administrator responsible for configuring NetBackup.

A system administrator is defined as a person with system administrator privileges and responsibilities.

This guide assumes:

- A basic understanding of system administration.
- A working understanding of the NetBackup client and server software.
- A familiarity with the information covered in the following NetBackup manuals:
  - *NetBackup User's Guide - UNIX*
  - *NetBackup System Administrator's Guide - UNIX* or *NetBackup System Administrator's Guide - Windows NT/2000*
  - *NetBackup Troubleshooting Guide - UNIX* or *NetBackup Troubleshooting Guide - Windows NT/2000*
- A thorough understanding of Oracle8 Recovery Manager.

Organization

This guide is organized as follows:

- The first chapter in this manual describes the technical features and concepts of NetBackup for Oracle.
- The Installation chapter explains how to install NetBackup Advanced BLI Agent on your system.
- Configuration describes how to configure NetBackup and the Oracle8 Recovery Manager for use with NetBackup Advanced BLI Agent. This information supplements the NetBackup administration manuals.
- The Using NetBackup Advanced BLI Agent chapter explains how to use NetBackup to perform backups and restores of Oracle databases. This information supplements the NetBackup manuals.
- Troubleshooting explains how to use NetBackup logs to troubleshoot NetBackup Advanced BLI Agent operations. This information supplements the NetBackup manuals.

Related Documents

The following documents provide related information. For a more detailed listing of NetBackup documents, refer to *NetBackup Release Notes*.

If you have a UNIX server, refer to these documents:

- *NetBackup System Administrator's Guide - UNIX*

  Explains how to configure and manage NetBackup on a UNIX system.

- *NetBackup Media Manager System Administrator's Guide - UNIX*

  Explains how to configure and manage the storage devices and media on UNIX NetBackup servers. Media Manager is part of NetBackup.

- *NetBackup Troubleshooting Guide - UNIX*

  Provides troubleshooting information for UNIX-based NetBackup products. You can also refer to www.veritas.com knowledge base for tech notes.

If you have a Windows NT/2000 server, refer to these documents:

- *NetBackup System Administrator's Guide - Windows NT/2000*

  Explains how to configure and manage NetBackup on a Windows NT/2000 server system.

- *NetBackup Media Manager System Administrator's Guide - Windows NT/2000*

  Explains how to configure and manage the storage devices and media on Windows NT/2000 NetBackup servers. Media Manager is part of NetBackup.

- *NetBackup Troubleshooting Guide - Windows NT/2000*

  Provides troubleshooting information for Windows NT/2000-based NetBackup products. You can also refer to www.veritas.com knowledge base for tech notes.

For this product, also refer to the following Oracle manuals:

- *Oracle Enterprise Manager Administrator's Guide*
- *Oracle8 Server Backup and Recovery Guide*

For this product, also refer to the following VERITAS manuals:

- *Database Edition for Oracle Database Administrator's Guide*
- *Storage Edition for Oracle Database Administrator's Guide*
- *NetBackup for Oracle Advanced BLI Agent System Administrator's Guide for Backups Without RMAN*

Conventions

The following explains typographical and other conventions used in this guide.

Type Style

Table 1. Typographic Conventions

Typeface	Usage
`Bold fixed width`	Input. For example, type **`cd`** to change directories.
`Fixed width`	Paths, commands, filenames, or output. For example: The default installation directory is `/opt/VRTSxx`.
Italics	Book titles, new terms, or used for emphasis. For example: *Do not* ignore cautions.
Sans serif (italics)	Placeholder text or variables. For example: Replace *filename* with the name of your file.
Sans serif (no italics)	Graphical user interface (GUI) objects, such as fields, menu choices, etc. For example: Enter your password in the Password field.

Notes and Cautions

Note This is a Note and is used to call attention to information that makes it easier to use the product or helps you to avoid problems.

Caution This is a Caution and is used to warn you about situations that can cause data loss.

Key Combinations

Some keyboard command sequences use two or more keys at the same time. For example, you may have to hold down the Ctrl key before you press another key. When this type of command is referenced, the keys are connected by plus signs. For example:

Press Ctrl+t

Command Usage

The following conventions are frequently used in the synopsis of command usage.

brackets [ ]

The enclosed command line component is optional.

Vertical bar or pipe (|)

Separates optional arguments from which the user can choose. For example, when a command has the following format:

command *arg1*|*arg2*

the user can use either the *arg1* or *arg2* variable.

Getting Help

For updated information about this product, including system requirements, supported platforms, supported peripherals, and a list of current patches available from Technical Support, visit our web site:

`http://www.veritas.com/`

For product assistance, contact VERITAS Customer Support.

US and Canadian Customers: 1-800-342-0652

International Customers: +1 (650) 335-8555

VERITAS Customer Support can also be reached through electronic mail at:

`support@veritas.com`

Introduction 1

NetBackup for Oracle on UNIX integrates the database backup and recovery capabilities of Oracle8 Recovery Manager with the backup and recovery management capabilities of NetBackup and its Media Manager.

This chapter introduces NetBackup Advanced BLI Agent and how it relates to Oracle8 Recovery Manager and NetBackup. Read this chapter for a description of:

- Features of NetBackup for Oracle on UNIX
- Terminology for NetBackup for Oracle on UNIX
- Technical Overview of NetBackup for Oracle on UNIX

In this guide, VERITAS NetBackup for Oracle on UNIX Advanced BLI Agent is referred to as NetBackup for Oracle on UNIX or as NetBackup Advanced BLI Agent.

Features of NetBackup for Oracle on UNIX

This section describes the NetBackup for Oracle on UNIX main features.

Feature	Description
Media and device management	All devices supported by Media Manager are available to NetBackup for Oracle on UNIX.
Scheduling facilities	NetBackup scheduling facilities on the master server can be used to schedule automatic and unattended Oracle backups. This also lets you choose the times when these operations can occur. For example, to prevent interference with normal daytime operations, you can schedule your database backups to occur only at night.
Multiplexed backups and restores	NetBackup for Oracle on UNIX lets you take advantage of NetBackup's multiplexing capabilities. Multiplexing directs multiple data streams to one backup device, thereby reducing the time necessary to complete the operation.
Transparent execution of both Oracle and regular file system backup and restore operations	All backups and restores are executed simultaneously and transparently without any action from the NetBackup administrator. A database administrator can execute database backup and restore operations through NetBackup or use Oracle8 Recovery Manager as if NetBackup were not present. An administrator or any other authorized user can use NetBackup to execute database backups and restores.
Sharing the same devices and tapes used during other file backups	It is possible to share the same devices and media used for other backups or to give Oracle exclusive use of certain devices and media.
Centralized and networked backup operations	From the NetBackup master server, you can schedule database backups or start them manually for any client. The Oracle databases can also reside on hosts that are different from the devices on which NetBackup stores the backups.

Feature	Description
Graphical user interfaces	NetBackup provides the following graphical user interfaces for client users and administrators: ◆ Client user motif interface, `xbp` ◆ Administrator user interface on Java, `jnbSA` ◆ Administrator user interface, `xbpadm` ◆ Administrator user interface on Windows NT/2000 A database administrator or NetBackup administrator can start backup or restore operations for Oracle from the NetBackup graphical user interface on the master server.
Parallel backup and restore operations	NetBackup for Oracle on UNIX supports the parallel backup and restore capabilities of the Oracle8 Recovery Manager. This permits the user to run more than one tape device at a time for a single Oracle backup or restore, thereby reducing the time necessary to complete the operation.

Terminology for NetBackup for Oracle on UNIX

This section explains important terms that may be new to an Oracle database administrator or a NetBackup administrator.

NetBackup Terms

This section describes NetBackup terms as they apply to NetBackup for Oracle on UNIX.

NetBackup	NetBackup backs up and restores files, directories, raw partitions, and databases on client systems that have Oracle databases.
NetBackup schedule	NetBackup schedules control NetBackup operations such as: when backups can occur, the type of backup (full, incremental) to perform, and how long NetBackup retains the image (retention level).
Administrator directed backups	NetBackup administrators are able to perform remote backups of all files, directories, databases, and raw partitions contained on client systems within a client/server network.
User-directed backups and restores	NetBackup for Oracle on UNIX users are able to perform backups of all files, directories, databases, and raw partitions contained on client systems within a client system.
Graphical interfaces	Graphical user interfaces are available for both users and administrators.
Media Manager	The Media Manager provides complete management and tracking of all devices and media used during backups and restores.
Block Level Incremental (BLI) Backup	Block-Level Incremental (BLI) Backup interface extends the capabilities of NetBackup to backup only changed data blocks of Oracle database files. BLI Backup accomplishes this using the Storage Checkpoint facility on the VERITAS File System (VxFS). BLI Backup uses Storage Checkpoint to identify all the data blocks that have changed since the last Storage Checkpoint, then backs up only those changed data blocks. This reduces both the time required to complete a database backup and the amount of data transferred during backups. BLI Backup also allows more frequent backups, resulting in more up-to-date backup images. When restoring from backups, the restore time is increased only by the extra time required to apply the incremental backups after a full restore completes.

Oracle Terms

This section describes Oracle terms as they apply to the NetBackup for Oracle on UNIX product.

Oracle Enterprise Manager The Oracle Enterprise Manager-Backup Manager is a graphical user interface for the Recovery Manager (`rman`). This interface allows you to use the point and click method to perform backups and recoveries. For more information about the Oracle Enterprise Manager, see the *Oracle Enterprise Manager Administrator's Guide*.

Oracle8 Recovery Manager Oracle8 Recovery Manager (RMAN) is used to back up, restore, and recover database files. Oracle8 Recovery Manager starts Oracle8 server processes on the target database. These Oracle8 server processes actually perform the backup and restore. Oracle8 Recovery Manager performs important backup and recovery procedures, and greatly simplifies the tasks administrators perform during these processes. However, it cannot directly manage the storage devices and media used in its backups and must be integrated with an application that has these capabilities. NetBackup for Oracle on UNIX provides device and media management capability by integrating the Oracle8 Recovery Manager with NetBackup and its media management software. Other advantages are access to NetBackup's automatic scheduling facilities and the graphical interfaces.

NetBackup Advanced BLI Agent currently supports Oracle8 Recovery Manager, Release 8.1.5.

Recovery Catalog The recovery catalog is a repository of information that is used and maintained by Oracle8 Recovery Manager. Oracle8 Recovery Manager uses this information to determine how to execute requested backup and restore actions.

Oracle recommends that you use Oracle8 Recovery Manager with a recovery catalog, especially if you have 20 (or more) datafiles. However, you are not required to maintain a recovery catalog with Oracle8 Recovery Manager.

Because most information in the recovery catalog is also available in the target database's control file, Oracle8 Recovery Manager supports an operational mode where it uses the target database control file instead of a recovery catalog. This operational mode is appropriate for small databases where installation and administration of another database, for the sole purpose of maintaining the recovery catalog, would be burdensome.

For a detailed description of the recovery catalog, refer to Chapter 7, "Recovery Manager Concepts," in the *Oracle8 Server Backup and Recovery Guide*.

rman command

The `rman` command is used to start a backup or restore by the Oracle8 Recovery Manager. Recovery Manager is an integral part of Oracle8. This command is described in the *Oracle8 Server Backup and Recovery Guide*.

RMAN script

The RMAN script specifies the commands that Oracle8 Recovery Manager will perform (for example, backups, and restores). This command file is described in the *Oracle8 Server Backup and Recovery Guide* (available from the Oracle Corporation).

The *Oracle8 Server Backup and Recovery Guide* explains the `rman` commands and the command script files. Refer to that guide for details on command syntax and parameters for performance.

There are example RMAN script in the directory named *install_path*`/netbackup/ext/db_ext/oracle/samples/rman`. These scripts are executable Oracle8 Recovery Manager command files and are fully commented to explain the features used. We recommend that you review these examples. You can use them as a starting point for developing backup, restore and recovery scripts.

Full Backup

A full backup copies all blocks into the backup set, skipping only datafile blocks that have never been used. Note that a full backup is not the same as a whole database backup; full is an indicator that the backup is not incremental.

A full backup has no effect on subsequent incremental backups, which is why it is not considered part of the incremental strategy. In other words, a full backup does not affect which blocks are included in subsequent incremental backups.

Incremental Backup

An incremental backup is a backup of only those blocks that have changed since a previous backup. Oracle allows you to create and restore incremental backups of datafiles, tablespaces, and a database. You can include a control file in an incremental backup set, but the control file is always included in its entirety, no blocks are skipped.

Multi-Level Incremental Backup

Oracle8 Recovery Manager allows you to create multi-level backups. Each level is denoted by an integer, e.g., 0, 1, 2, etc. A level 0 incremental backup, which is the base of subsequent incremental backups, copies all blocks containing data.

When you generate a level n incremental backup in which n is greater than 0, you back up:

- All blocks that have been modified since the most recent backup at a level n or lower (this is the default type of incremental backup, called Differential Incremental Backup)
- All blocks that have been modified since the most recent backup at level n-1 or lower (called a Cumulative Incremental Backup)

The benefit of performing multi-level incremental backups is that you do not back up all of the blocks all of the time. Incremental backups at level greater than 0 only copy blocks that were modified, which means that the backup size may be significantly smaller and require much less time. The size of the backup file depends solely upon the number of blocks modified and the incremental backup level.

Differential Incremental Backup

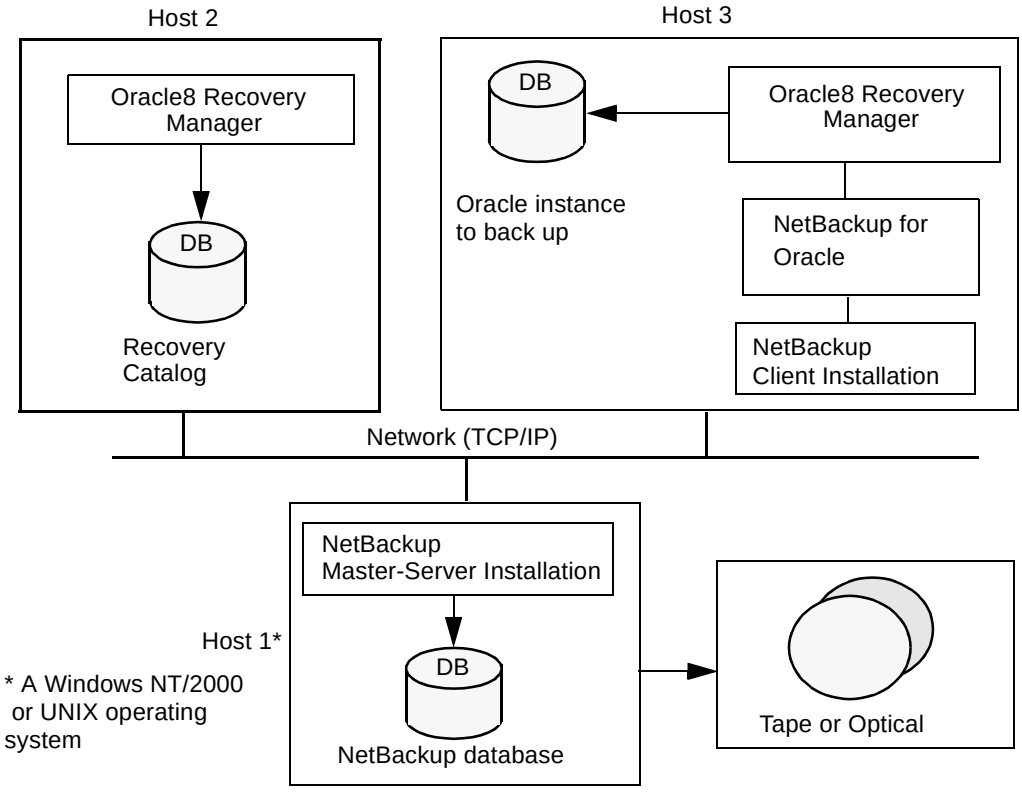
In a differential level n incremental backup, you back up all blocks that have changed since the most recent backup at level n or lower. For example, in a differential level 2 backup, you back up all blocks modified since the last level 2, level 1, or level 0 backup. Incremental backups are differential by default.

Cumulative Incremental Backup	In a cumulative level n incremental backup, you back up all blocks that have changed since the most recent backup at level n-1 or lower. For example, in a cumulative level 2 backup, you back up all blocks changed since the most recent level 1 or level 0 backup. Cumulative incremental backups reduce the work needed for a restore by ensuring that you only need one incremental backup from any particular level at restore time. Cumulative backups require more space and time than Differential Incremental Backups, however, because they duplicate the work done by previous backups at the same level.
proxy copy	Proxy copy is an extension to the Oracle8i Media Management API, which gives media management software more control over database backups and restores. With proxy copy feature, instead of simply writing to or reading from the backup devices, the media management software will manage the entire data transfer. With proxy copy, RMAN provides a list of files requiring backup or recovery to the media management software, which in turn makes all decisions regarding how and when to move the data. Proxy copy allows NetBackup for Oracle on UNIX to take advantage of these special features to reduce or eliminate the impact of a backup on an Oracle server. Backups and restores remain tightly integrated with RMAN and its catalog, greatly simplifying administration tasks.

Technical Overview of NetBackup for Oracle on UNIX

The example network below shows the major components in a NetBackup for Oracle on UNIX configuration.

The storage devices are connected to the NetBackup master server. The host with the database must be a NetBackup client and have NetBackup for Oracle on UNIX installed.



How Does the Oracle8 Recovery Manager Work?

The Oracle8 Recovery Manager (RMAN) performs a wide variety of automated backup and recovery functions. During backup or restore, RMAN provides the interface to the databases and performs the actual extraction and insertion of data for non-proxy operations.

To start a database backup or restore, the database administrator must execute a command called `rman`. This command can be executed from the command line, a shell script, or an application such as NetBackup. The RMAN script is used as a parameter to the `rman` command and defines the commands to be performed (for example, backup or restore). The RMAN script also defines other components of the operation such as the database objects to be backed up or restored.

During a non-proxy backup or restore, the Oracle8 Recovery Manager controls the data streams going into or out of a database. The Oracle8 Recovery Manager can access storage devices when it is integrated with a media management system, such as that provided by NetBackup and its Media Manager.

For more information, read the *Oracle8 Server Backup and Recovery Guide.*

How Does NetBackup for Oracle on UNIX Work?

Following are prerequisites for performing backups to tape:

- Linking with NetBackup and its Media Manager
- Generating unique file names

NetBackup for Oracle on UNIX includes a library of functions that enable the Oracle8 Recovery Manager to use NetBackup and its Media Manager. To link with this library, see “Linking Oracle 8 with NetBackup Advanced BLI Agent” on page 34.

When using the `backup` command, each resulting backup set contains at least one backup piece (datafile, datafile copy, control file, or archive log) from the target database. You must give each backup piece a unique name using the `format` operand. Several substitution variables are available to aid in generating unique names. You can specify the `format` operand in the `backup` command or in the `allocate channel` command. NetBackup considers the backup piece name as the file being backed up, so this name must be unique in the catalog. In Oracle8i, the RMAN backup will fail with a `file already exists in catalog` error.

Note For performance reasons, NetBackup for Oracle on UNIX strongly recommends that an RMAN timestamp be placed at the end of each backup piece name. Without this timestamp, there will be performance degradation as the NetBackup catalog grows. To guarantee uniqueness and optimal performance, specify each backup file name as:

*tag*_%U%_%t for Oracle8i

where
tag is any user-defined string
%s specifies the backup set number
%p specifies the backup piece number with the backup set
%t specifies the backup set timestamp
%U specifies a shorthand for %u_%p_%c that guarantees uniqueness in generated file names. (Oracle8i).

For more information on generating unique file names, refer to the *Oracle8 Server Backup and Recovery Guide.*

NetBackup users or schedules start database backups or restores by specifying a NetBackup for Oracle script in the file list of the Oracle class. The shell executes NetBackup for Oracle script. The NetBackup for Oracle script instructs the shell to execute the rman command with the RMAN script as a parameter.

For a backup:

1. The rman command, with the RMAN script as a parameter, starts the requested operation on the databases.

2. When the process requires media to store backup data, RMAN starts a user-directed backup by issuing a backup request.

3. The NetBackup master server connects to NetBackup for Oracle on UNIX on the client and transfers the database data to secondary storage.

A restore works in essentially the same manner except that RMAN issues a restore request. This causes NetBackup to retrieve the data from secondary storage and send it to NetBackup for Oracle on UNIX on the client.

Since the Oracle8 Recovery Manager supports parallel operations, a single rman execution can start more than one backup or restore on the NetBackup system.

The status for an RMAN operation is stored in the Oracle8 Recovery Manager catalog. This is the only status that a database administrator must check to verify that a backup or restore has been successful.

NetBackup also logs status, but only for its own part of the operation. The database administrator cannot use the NetBackup status to determine whether `rman` was successful. Errors can occur in `rman` that do not affect NetBackup and are not recorded in its logs.

Incremental Backup of Oracle Databases

The Oracle8 Recovery Manager provides true incremental backups. An incremental backup will back up datafiles, which include only the blocks that have been changed since the last incremental.

Refer to the *Oracle8 Server Administrator's Guide* and the *Oracle8 Server Backup and Recovery Guide* for more complete details on the backup and recovery process.

Proxy Copy and Block-Level Incremental (BLI) Backup

NetBackup Advanced BLI Agent implements the proxy copy feature of the Oracle Media Management API and performs BLI Backups of Oracle8i database files. To allow BLI Backups, the datafiles must reside on VxFS file systems using Version 4 layout.

On Solaris, you also must have a valid VERITAS Database Edition for Oracle license.

On HP-UX, you also must have a valid VERITAS Storage Edition for Oracle license.

Use the `fstyp -v` device command to determine the layout version of your file system. See the `vxupgrade(1M)` manual page for information on upgrading file systems to the Version 4 layout.

Proxy Copy

Proxy copy is an extension to Oracle8i's Media Management API. It gives NetBackup Advanced BLI Agent more control over database backups and restores. Instead of simply writing to or reading from the backup devices, NetBackup Advanced BLI Agent manages the entire data transfer between the disks containing Oracle datafiles and Netbackup and its Media Manager.

Note Currently, Oracle8i supports proxy backups for datafiles only. Backups of control files and archive logs must be performed using RMAN non-proxy backups.

With proxy copy, RMAN provides a list of files, requiring backup or recovery, to NetBackup Advanced BLI Agent. NetBackup Advanced BLI Agent makes all decisions regarding how the data is moved and when to move the data.

Backups and restores remain tightly integrated with RMAN and its catalog, greatly simplifying administration tasks.

Block-Level Incremental (BLI) Backup

If only a small portion of a database changes on a daily basis, full database backups are costly in terms of time and media. BLI Backup interface extends the capabilities of NetBackup to back up only changed data blocks of Oracle database files.

BLI Backup accomplishes this backup methodology using the Storage Checkpoint facility in the VERITAS File System (VxFS) available through the Database Edition on Solaris and the Storage Edition on HP-UX.

The VxFS Storage Checkpoint facility keeps track of data blocks modified by the database since the last backup. NetBackup with BLI Backup leverages this facility to back up only changed blocks, not the entire database, for an incremental backup.

A database BLI Backup is done at the file system block level, which means only changed blocks, *not* the changed files, are backed up. Because the VxFS Storage Checkpoint facility identifies changed blocks in real time, BLI Backup does not need to search the entire database for the modified blocks. BLI Backup saves time, decreases the amount of backup media required, and significantly reduces CPU and network overhead during backups. In addition, BLI Backup allows more frequent backups, making backup images more up-to-date.

BLI Backup supports full, as well as block-level incremental backups of Oracle databases. Two types of block-level incremental backups are supported: *differential* and *cumulative*. Full, differential incremental, and cumulative incremental backups are specified as part of the backup schedule. When restoring Oracle database files, NetBackup restores an appropriate full backup, then applies the changed blocks from the incremental backups. This restore process is performed automatically by NetBackup and is completely transparent.

Cumulative BLI Backup

This is backup of all the data blocks of database files that changed since the last full backup. Since a cumulative BLI Backup image contains only the data blocks of database files that changed since the last full backup, a cumulative BLI Backup can reduce the number of incremental backup images that must be applied to a restore operation, improving restore times.

Differential BLI Backup

A backup where NetBackup performs a backup of only those data blocks of database files that changed since the last backup of any type (full, cumulative incremental, or differential incremental backup) was performed.

Full Backup

A backup where NetBackup backs up the entire database file, not just data blocks changed since the last full or incremental backup.

Storage Checkpoint

VxFS Storage Checkpoint is a disk- and I/O-efficient snapshot of file systems. A Storage Checkpoint provides a consistent, stable view of a file system at the point in time when the file system was snapped or checkpointed. Instead of making a physically separate copy of the file system, a Storage Checkpoint identifies and maintains only changed file system blocks, saving disk space and significantly reducing I/O overhead.

By keeping track of changed blocks, the VxFS Storage Checkpoint enables BLI Backups. VxFS Storage Checkpoint facility provides a consistent view of file systems, allowing BLI Backup to freeze the database image for consistent database backups.

The Storage Checkpoint operation is similar to the snapshot file system mechanism. However, unlike a snapshot, the Storage Checkpoint persists after a system reboot. Also, the Storage Checkpoint operation is totally transparent to administrators. The Checkpoint image is managed and available only through NetBackup or the VxDBA utility for database backup.

For more information on Storage Checkpoint, see one of the following:

- The *VERITAS Database Edition for Oracle Database Administrator's Guide* for Solaris.
- The *VERITAS Storage Edition for Oracle Database Administrator's Guide* for HP-UX.

When NetBackup initiates full database backups, followed by block level incremental backups, it takes appropriate Storage Checkpoints of the Oracle datafile file systems. These Storage Checkpoints identify and maintain a list of modified blocks.

File System and Storage Checkpoint Space Management

To support BLI Backup, the VxFS file systems need extra disk space to keep track of the block change information. In general, about 1% of a file system size is sufficient.

How Does Proxy Copy and BLI Backup Work with NetBackup Advanced BLI Agent?

NetBackup Advanced BLI Agent extends the functionality of NetBackup for Oracle on UNIX and integrates the proxy copy feature to allow BLI Backups of Oracle8i datafiles residing on VxFS file systems.

Note Oracle executables do not have to be installed on the VxFS file system.

To use NetBackup Advanced BLI Agent you must first have a thorough understanding of NetBackup for Oracle on UNIX. Refer to "How Does NetBackup for Oracle on UNIX Work?" on page 10.

Review this section for a short overview of the way NetBackup Advanced BLI Agent works.

NetBackup users or schedules start database backups or restores by specifying a NetBackup for Oracle script in the file list of the Oracle class. The NetBackup for Oracle script instructs the shell to execute the `rman` command with the RMAN script as a parameter.

With proxy copy, RMAN provides a list of files, requiring backup or recovery, to NetBackup Advanced BLI Agent. NetBackup Advanced BLI Agent uses the BLI Backup interface to perform the actual extraction and insertion of data.

When Oracle performs proxy copy backups, it puts the datafile being backed up into backup mode. NetBackup Advanced BLI Agent uses Nodata Storage Checkpoint for proxy copy BLI backups. As a result, all datafiles on the file system that are being backed up will be put into backup mode for the duration of the backup.

On the initial call, NetBackup Advanced BLI Agent returns a special entry to RMAN indicating that it supports proxy copy. It also indicates to RMAN that it supports an unlimited number of files to be proxy copied in a single proxy copy session. This means that the degree of parallelism for proxy backups will not be controlled by the number of channels allocated for the RMAN `backup` command unless the user specifies a name of a particular channel with a `channel` operand for a *backup_specification* as demonstrated in "Example 4b" on page 18.

Due to the fact that Storage Checkpoint takes a snapshot of a file system for a particular database instance, you must perform backups of all database datafiles residing on the same file system at the same time. To insure that NetBackup always has consistent images of the database files, NetBackup Advanced BLI Agent will back up files on the same file system at the same time. At the beginning of the backup, NetBackup Advanced BLI Agent groups all files it received from RMAN based on the file system to which they belong. It then starts backup streams for all files in each file system at the same time. A number of proxy copy backup streams to be started on each file system is controlled by the `NB_ORA_PC_STREAMS` variable. However, the number of parallel streams started will never be larger than the number of files on the file system. The number of file systems to be backed up in parallel is controlled by the `NB_ORA_PC_FILESYSTEMS` variable. Refer to "Environment Variables Set Up by a User" on page 78.

The following configuration examples demonstrate how to make sure all datafiles on the file system get backed up in one RMAN backup set.

Caution Use caution when configuring a proxy backup of a subset of a database. Backing up tablespaces may result in inconstant backup images for all subsequent backups of the datafiles.

Example 1

```
run {
   allocate channel t1 type 'SBT_TAPE';
   send 'NB_ORA_PC_STREAMS=3';
   backup proxy
      format  'bk_%U_%t'
      (database);
   release channel t1;
}
```

For example, the database consists of two datafiles (df1 and df2) residing on file system F1.

RMAN will open one channel and provide a list of all datafiles to NetBackup Advanced BLI Agent.

Although the NB_ORA_PC_STREAMS variable value is three, there are only two datafiles. Therefore, NetBackup Advanced BLI Agent will start only two parallel BLI Backup jobs, one for datafile df1 and the other for datafile df2. After the proxy backup is done, RMAN will start a non-proxy backup of the database control file.

Example 2

```
run {
   allocate channel t1 type 'SBT_TAPE';
   allocate channel t2 type 'SBT_TAPE';
   send 'NB_ORA_PC_STREAMS=3';
   backup proxy
      format  'bk_%U_%t'
      (database);
   release channel t1;
   release channel t2;
}
```

In this example the database is configured as in Example 1. RMAN will open two channels. It will use one channel to proxy backup all datafiles and another channel to start a non-proxy back up of the control file in parallel with the proxy backup.

The following examples show how to proxy back up a subset of a database.

Example 3

```
run {
   allocate channel t1 type 'SBT_TAPE';
   send 'NB_ORA_PC_STREAMS=3';
   backup proxy
      format  'bk_%U_%t'
```

```
      (tablespace tbs1, tbs2);
   release channel t1;
}
```

For example, tablespaces tbs1 and tbs2 consist of datafiles residing on file system F1. In this example RMAN will open channel t1 to start a backup of all datafiles of tablespaces tbs1 and tbs2. NetBackup Advanced BLI Agent will decide how many parallel backup streams to start based on the values of NB_ORA_PC_STREAMS variable and the number of datafiles. If the number of datafiles is larger than three, then NetBackup Advanced BLI Agent will group all datafiles into three backup streams based on size.

Example 4

These examples illustrate potential problems you may encounter when performing a proxy backup of a subset of a database. Depending upon your database layout, your backup may result in inconsistent backup images.

Caution Use caution when configuring a proxy backup of a subset of a database. Backing up tablespaces may result in inconstant backup images for all subsequent backups of the datafiles.

Example 4a

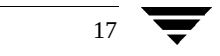
```
run {
   allocate channel t1 type 'SBT_TAPE';
   backup proxy
      format  'bk_%U_%t'
      (tablespace tbs1);
   backup proxy
      format  'bk_%U_%t'
      (tablespace tbs2);
   release channel t2;
}
```

In this example, RMAN will use the channel to start a backup of tablespace tbs1 datafiles. After the backup is done, it will start a backup of tablespace tbs2 datafiles.

This is *not* a problem when:

- All tablespace tbs1 datafiles reside on file system F1.
- All tablespace tbs2 datafiles reside on files system F2.

  In this case, a Storage Checkpoint of file system F1 and a Storage Checkpoint of file system F2 will be done only once. The backup images of the datafiles will be consistent.



This *is* a problem when:

- Tablespace tbs1 datafiles reside on file system F1 and file system F2.
- Tablespace tbs2 datafiles reside on file system F1 and file system F2.

In this case, a Storage Checkpoint of each file system will be taken twice: first, when the backup of tbs1 is done and then latter, when the backup of tbs2 is done. This will result in inconstant backup images for all subsequent backups of the datafiles.

Example 4b

```
run {
   allocate channel t1 type 'SBT_TAPE';
   allocate channel t2 type 'SBT_TAPE'
   backup proxy
      format  'bk_%U_%t'
      (tablespace tbs1 channel t1)
      (tablespace tbs2 channel t2);
   release channel t1;
   release channel t2;
}
```

In this example the user specifically indicates which channel to use for each tablespace backup. RMAN will use channel t1 to backup tablespace tbs1 datafiles and channel t2 to backup tablespace tbs2 datafiles.

As in the previous example, this *is* a problem when:

- Tablespace tbs1 datafiles reside on file system F1 and file system F2.
- Tablespace tbs2 datafiles reside on file system F1 and file system F2.

In this case, a Storage Checkpoint of each file system will be taken twice: first, when the backup of tbs1 is done and then latter, when backup of tbs2 is done. This will result in inconstant backup images for all subsequent backups of the datafiles.

Oracle Class Configuration Issues

When the Oracle class is configured, the user needs to enable the Block level incremental attribute to allow BLI Backup. Refer to "NetBackup Administration - Java Interface" on page 46 for instructions.

All proxy backups are BLI Backups performed by *Automatic Full Backup, Automatic Differential Incremental Backup,* and *Automatic Cumulative Incremental Backup* schedules. Even if proxy backup is initiated by a user, NetBackup always starts one of these schedules to perform BLI Backup. If the proxy schedule name is not specified on the request, the NetBackup server will start an *Automatic Full Backup* schedule by default.

NetBackup Advanced BLI Agent will always check that a full backup has been performed before proceeding with an incremental backup. If the NetBackup scheduler or user initiates an incremental backup, and NetBackup Advanced BLI Agent finds no record of a full backup, it will perform a full backup.

To insure that it has a proper set of images to restore, NetBackup will perform a full backup when it encounters the following situations:

- If the number of backup streams specified with NB_ORA_PC_STREAMS has changed from the previous backup
- If NetBackup does not have a valid full backup image in its database (for example, if images were expired)
- If a new file, created before but not included in a previous full backup, was added to an incremental backup
- If there is no valid full Storage Checkpoint information available

NetBackup Advanced BLI Agent will always initiate a full backup under these conditions, even if the user wants to perform an incremental backup.

Currently, Oracle8i does not support proxy backups of database control files and archived logs. To be able to perform a whole database proxy backup, which automatically includes a backup of the control file, you must configure:

- One or more automatic backup schedules to perform proxy BLI Backups of the datafiles
- *Backup Policy* schedule type to backup the control file and the archived logs

To make sure that NetBackup Advanced BLI Agent uses the proper class and schedule, use RMAN send command to pass the NB_ORA_CLASS and NB_ORA_PC_SCHED variables. Refer to “Example 4, hot_database_backup_proxy.sh” on page 94.

Quick I/O Oracle Datafiles

NetBackup Advanced BLI Agent fully supports backups and restores of the Quick I/O Oracle datafiles. A Quick I/O file consists of two components: a hidden file with a space allocated for it, and a link that points to the Quick I/O interface of the hidden file. On the backup, NetBackup Advanced BLI Agent follows the symbolic link and backs up both components of the Quick I/O file: the symbolic link and the hidden file.

During the restore job, NetBackup Advanced BLI Agent will restore both components from the backup image. If either one or both of the components is missing, NetBackup Advanced BLI Agent will create the missing component(s).

Script Examples for Proxy Copy Backups

Refer to the following NetBackup Advanced BLI Agent scripts for proxy database backup.

- "Example 4, hot_database_backup_proxy.sh" on page 94
- "Example 8, hot_database_backup_proxy.rcv" on page 110

Sequence of Operation

The backup sequence of operation is:

1. NetBackup Advanced BLI Agent receives a list of datafiles to back up.

   Each datafile is identified by a unique backup file name in the NetBackup catalog. To insure this, you must give each datafile a unique name using the `format` operand. You can use the same technique as with non-proxy backups.

Note For performance reasons, NetBackup for Oracle on UNIX strongly recommends that an RMAN timestamp be placed at the end of each backup piece name. Without this timestamp, there will be performance degradation as the NetBackup catalog grows. To guarantee uniqueness and optimal performance, specify each backup file name as:

<*tag*>_%U%_%t

where
<*tag*> is any user-defined string
`%s` specifies the backup set number
`%p` specifies the backup piece number with the backup set
`%t` specifies the backup set timestamp
`%U` specifies a shorthand for `%u_%p_%c` that guarantees uniqueness in generated file names.

2. NetBackup Advanced BLI Agent checks that:
   - each datafile in the file list is a VxFS file

     Oracle executables do not need to be installed on VxFS file systems.
   - valid VERITAS Database Edition for Oracle license is installed

Note If you try to perform proxy backups of Oracle datafiles residing on file systems other than VxFS, NetBackup Advanced BLI Agent will return the following message to RMAN: `Could not proxy backup file` *filename*. If the user specifies the `proxy` (versus `proxy only`) operand of the RMAN `backup` command, then RMAN will try to perform a non-proxy backup.

3. NetBackup Advanced BLI Agent groups the datafiles based on the file system they belong to.

4. NetBackup Advanced BLI Agent uses the value of the NB_ORA_PC_FILESYSTEMS variable to determine how many file systems to back up in parallel.

5. NetBackup Advanced BLI Agent uses the value of the NB_ORA_PC_STREAMS variable to determine the number of backup streams to start for each file system.

6. For each file, the Storage Checkpoint facility identifies all the data blocks that changed since the last Storage Checkpoint.

Any of the datafiles proxy backed up in one session may be restored individually later, or they may be restored in the same proxy copy session along with other backup files created in different proxy copy sessions.

Restoring of any of the incremental backup images requires NetBackup to restore the last full backup image and all the subsequent incremental backups until the specified incremental backup image is restored. This restore process is performed automatically by NetBackup and is completely transparent. The media that stored the last full backup and the subsequent incremental backups must be available; otherwise, the restore cannot proceed.

Note that restoring a file rewrites all blocks in that file. The first subsequent differential incremental backup and all subsequent cumulative incremental backups will back up all of the blocks in the restored file. After restoring an entire database, note that the first subsequent backup will result in a full backup.

The restore destination can be a VxFS, UFS (Solaris), or HFS (HP-UX) file system. The destination VxFS file system does not need to support the Storage Checkpoint feature to restore files; however, a VxFS file system with the Storage Checkpoint feature is needed to perform BLI Backups of the restored data.

The restore sequence of operation is:

1. NetBackup Advanced BLI Agent receives a list of datafiles to restore.

2. NetBackup Advanced BLI Agent sends a restore request to the NetBackup server for all files in the list.

3. The NetBackup server restores the following in sequence for each file.

   a. The last full backup that occurred before the target date and time.

   b. All incremental backups that occurred since the last full backup.

Installation 2

This chapter describes the NetBackup Advanced BLI Agent installation procedure. It includes a section on installation prerequisites. It also contains sections on installing Oracle7 Enterprise Backup Utility and Oracle8 Recovery Catalog.

If you are going to perform BLI Backups with RMAN after installing this software, refer to "Configuration" on page 39.

If you are going to perform BLI Backups without RMAN after installing this software, refer to `NetBackup_AdminGuide_BLI.pdf`. This is the online version of the *NetBackup for Oracle Advanced BLI Agent System Administrator's Guide for Backups Without RMAN*.

To determine which Oracle version levels are supported, refer to the Database Extension Matrix in the *NetBackup Release Notes.*

Installation Requirements

Make sure your system meets the following requirements:

1. The following products are properly installed and configured:
   - NetBackup 3.4
   - For Solaris, VERITAS Database Edition for Oracle Release 2.0 or later
   - For HP-UX, VERITAS Storage Edition for Oracle Release 2.0 or later
   - Oracle8i
2. VERITAS products must have valid licenses. To check for licenses, enter:

   `vxlicense -p`

   This command displays all of the valid VERITAS licenses installed on the system.
   - On Solaris, the Storage Checkpoint feature [83] and Database Edition product [100] appear in the list along with a date.
   - On HP-UX, the Storage Checkpoint feature [83] and Storage Edition product [108] appear in the list along with a date.

Installation Prerequisites

Before installing NetBackup Advanced BLI Agent, be sure to complete the following procedures:

1. Install NetBackup server software on the server.

   The NetBackup server platform can be any of those that NetBackup supports.

   For a DataCenter installation, refer to the *NetBackup DataCenter Installation Guide - UNIX* or the *NetBackup DataCenter Installation Guide - Windows NT/2000*.
2. Install the NetBackup client software on the client where you will be backing up the databases.

   For a DataCenter installation, refer to the *NetBackup DataCenter Installation Guide - UNIX* for installation instructions on UNIX clients.

Now you are ready to install NetBackup Advanced BLI Agent on the client where you will be backing up the databases. Refer to the next section for detailed instructions on installing NetBackup Advanced BLI Agent.

Note If you are going to perform BLI Backups with RMAN after installing this software, refer to "Configuration" on page 39.
If you are going to perform BLI Backups without RMAN after installing this software, refer to `NetBackup_AdminGuide_BLI.pdf`. This is the online version of the *NetBackup for Oracle Advanced BLI Agent System Administrator's Guide for Backups Without RMAN*.

Install NetBackup Advanced BLI Agent

There are two ways to install database extension software.

- Remote Installation

  Loads the software on a master server. The user will then push the database software out to affected clients.

  Refer to the following section.

- Local Installation

  Loads and installs the software only to the local machine.

  Refer to "Local Installation of NetBackup Advanced BLI Agent" on page 31.

Note During the installation, you will be prompted to enter a UNIX database administrators' group name. This is the name of the group that the Oracle software owner belongs to. In most cases the name is **dba**. If you have multiple Oracle installations on your system, with different software owner names, make sure that they all belong to a single UNIX database administrators' group.

Remote Installation of NetBackup Advanced BLI Agent

During a remote installation, NetBackup Advanced BLI Agent files are loaded onto the current machine, which must be a master server. The software will then be distributed to the clients and installed.

Before performing a remote install, make sure:

- There is adequate disk space on each client that will receive the software.

  Less than one megabyte of additional disk space is required in the client's *install_path*`/netbackup` directory. However, more disk space may be necessary at run time.

- NetBackup version 3.4 client software is installed and operational on each Oracle on UNIX Advanced BLI Agent client.

  This also means that the directory *install_path*/netbackup already exists on each Oracle on UNIX Advanced BLI Agent client.

Remote Install Procedure

1. Log in as the root user on the server.

   If you are already logged in, but are not the root user, execute the following command.

   ```
   su - root
   ```

2. Make sure a valid license key for NetBackup Advanced BLI Agent has been registered.

   Use the command *install_path*/netbackup/bin/admincmd/get_license_key to list and add keys.

3. Insert the CD-ROM into the drive.

4. Change the working directory to the CD-ROM directory.

   ```
   cd /CD_mount_point
   ```

5. Load the software on the server by executing the install script.

   ```
   ./install
   ```

   The following prompt will appear:

   ```
   Do you want to do a local installation? (y/n) [n]
   ```

   a. Answer **n**.

      You are presented with a menu of all database extensions available on the CD-ROM.

   b. Select the NetBackup for Oracle on UNIX Advanced BLI Agent option.

   c. Enter **q** to quit selecting options.

      A prompt will appear asking if the list is correct.

   d. Answer **y**.

      The install script identifies the types of client software loaded during the installation of the NetBackup server. By default, any matching NetBackup Advanced BLI Agent software will automatically be loaded. If there are more

platforms available, the script displays a menu giving you the opportunity to add more client types to the default list. Once the list is complete, database extension version files, compressed tar files and the `install_dbext` script are copied to directory *install_path*`/netbackup/dbext`.

6. Distribute and install the NetBackup Advanced BLI Agent software on each client.

Note It is expected that the NetBackup version level (for example, 3.4) running on each client that you wish to update matches the version level of the database extension being installed.

a. Execute the command to distribute the NetBackup Advanced BLI Agent software to the clients. This command varies, depending upon the type of install you will perform.

There are two types of installs.

- *initial install*
  Use an initial install if the clients you intend to update have not been configured into classes of type Oracle.
- *upgrade install*
  Use an upgrade install if all the clients you intend to update already have been configured into classes of type Oracle.

Initial Install Procedure

1. Execute the following command to create a file containing a list of clients currently configured in the NetBackup database.

`cd` *install_path*`/netbackup/bin`
`./admincmd/bpclclients -allunique -noheader >` *filename*

where *filename* is the name of the file to contain the list of unique clients. If no clients have been configured in the NetBackup database, and therefore *filename* is empty, create *filename* using the same format as that generated by `bpclclients`.

`bpclclients` generates output in following format:

hardware operating_system client_name

where

hardware is the hardware name. For examples, execute the `ls` command in directory *install_path*`/netbackup/client`.

operating_system is the operating system name. For examples, execute the `ls` command in directory *install_path*`/netbackup/client/`*hardware*.

client_name is the name of the client.

For example, the contents of *filename* might look like this:

```
Solaris Solaris2.6 curry.min.ov.com
RS6000 AIX4.3 cypress
```

2. Edit *filename*.

   This is an optional step. Use it if the contents of *filename* need to be changed. Edit *filename* to contain only those clients you wish to update with NetBackup Advanced BLI Agent software.

3. Specify *filename* on the `update_dbclients` command.

   For example:

`cd` *install_path*`/netbackup/bin`
`./update_dbclients Oracle_on_UNIX_Advanced_BLI_Agent -ClientList` *filename*

Only clients listed in *filename* will be updated.

Upgrade Install Procedure

Execute the following command.

`cd` *install_path*`/netbackup/bin`
`./update_dbclients Oracle_on_UNIX_Advanced_BLI_Agent ALL ALL`

This command will look at all possible clients and only update the ones currently in an Oracle class type.

Instead of `ALL ALL`, you may use `-ClientList` *filename* as explained in "Initial Install Procedure" on page 27.

Note With an initial or upgrade install, some clients may be skipped and not updated. Possible reasons are:
- the client is a PC client (which cannot be updated from a UNIX server),
- NetBackup Advanced BLI Agent does not support that client's platform type,
- the NetBackup Advanced BLI Agent software for that client type was not loaded onto the server in step 5,

- (if using the `ALL ALL` method) the client does not belong to an Oracle class type. All skipped clients are available in a file whose name is displayed by `update_dbclients`.

b. The number of updates required to distribute the software to the clients is displayed.

If more than one update will occur, you will see the following prompt:

`Enter the number of simultaneous updates you wish to take place. 1` *`max dflt`*

where:

max is the maximum number of simultaneous updates that is allowed. The value displayed will be a number ranging from 1 to 30.

dflt is the default number the program will use if you press Enter. The value displayed will be a number ranging from 1 to 15.

Example 1

If three client updates will be performed, the *max* and *dflt* values shown would be 3.

Example 2

If 50 client updates will be performed, the *max* value shown would be 30 and the *dflt* value shown would be 15.

`update_dbclients` will start the number of updates that you specify. If this number is less than the total number of client updates to be performed, new updates will start as the previous updates finish until all of the updates have been completed.

Based on your answer, the time it will take to update the clients is displayed, followed by this question:

`Do you want to upgrade the clients now? (y/n) [y]`

c. Enter **y** or **n** for the prompt.

If you answer **n**, `update_dbclients` will quit and leave the list of clients it would have updated in a file. This file can later be used by the `-ClientList` parameter mentioned previously.

Answer **y** to continue the installation process.

- If the configuration information for a client is unknown, you will be prompted for it. For example:

```
-----> Client curry.min.ov.com

Please specify the UNIX database administrators' group for
the Oracle software owner:
```

Type the UNIX group to which the Oracle software owner belongs. In most cases, the group entered will be dba. You will have an opportunity to validate your answer.

- If the `update_dbclients` command is able to determine a client's configuration, it will not prompt.

To change a client's Oracle software owner group configuration information later, you must log onto the client and execute the command *install_path*`/netbackup/bin/install_oracle`.

- If more than one client is being updated, the configuration information for the previous client (whether determined by asking explicitly or by `update_dbclients` itself) establishes a default answer for the next client. This makes configuration simpler when the clients have been similarly configured. Therefore, you may see a display similar to the following:

```
-----> Client guava.min.ov.com

Please specify the UNIX database administrators' group for
the Oracle software owner:

The previous response to this question was:
dba

Use the previous response? (y/n) [y]
```

Example: Let's assume that you want to update the following three clients.

`curry.min.ov.com`

`guava.min.ov.com`

`hat.min.ov.com`

As far as the script is concerned, the configuration information for all three clients is unknown.

Assume the Oracle software owner belongs to group **dba** on `curry.min.ov.com`, so when you are prompted for configuration information for `curry.min.ov.com`, specify **dba** as a valid group for the Oracle software owner.

The next client is `guava.min.ov.com`.

- If client `guava.min.ov.com` also has **dba** as its Oracle software owner group, the answer to the `Use the previous response?` prompt will be **y**.
- If client `guava.min.ov.com` has a different configuration, the answer to the `Use the previous response?` prompt will be **n**. The `Please specify the UNIX database administrators' group for the Oracle software owner:` prompt will re-display. At this point you specify a different group.

The next client is `hat.min.ov.com`. The Oracle software owner group you entered for `guava.min.ov.com` becomes the "previous response" displayed for `hat.min.ov.com`.

If the `update_dbclients` command was successful in distributing the software to the client, it will automatically run the `install_dbext` script on the client. Configuration information determined in step c on page 29 is used to complete the installation. If `install_dbext` has successfully completed, there will be a version file in directory *install_path*`/netbackup/ext` that contains the version of NetBackup Advanced BLI Agent that was installed and an installation timestamp. The `update_dbclients` command displays a note on whether the update was successful for each client. When the `update_dbclients` command has completed, it displays a file name that contains a complete log of what happened for each client. If the update failed for any client, the log file should be examined to determine the problem.

Local Installation of NetBackup Advanced BLI Agent

During a local installation, the NetBackup Advanced BLI Agent files are extracted and installed. You also are prompted for configuration information. The local machine can be a client or a master server that also happens to be a client.

Before performing a local install, make sure:

- The local machine has adequate disk space.

  Less than one megabyte of additional disk space is required in the *install_path*`/netbackup` directory. However, more disk space may be necessary at run time.

- NetBackup version 3.4 client software is installed and operational.

  This also means that the *install_path*`/netbackup` directory already exists.

Local Install Procedure

1. Log in as the root user on the machine.

   If you are already logged in, but are not the root user, execute the following command.

   ```
   su - root
   ```

   - If the local machine is a client, go to step 3.
   - If the local machine is a server, go to step 2.

2. Make sure a valid license key for NetBackup Advanced BLI Agent has been registered.

   Use the command *install_path*`/netbackup/bin/admincmd/get_license_key` to list and add keys.

3. Insert the CD-ROM into the drive.

4. Change the working directory to the CD-ROM directory.

   ```
   cd /CD_mount_point
   ```

5. Load and install the software by executing the `install` script.

Note It is expected that the NetBackup version level (for example, 3.4) running on the local machine matches the version level of the database extension being installed.

```
./install
```

The following prompt will appear:

```
Do you want to do a local installation? (y/n) [n]
```

a. Answer **y**.

   You are presented with a menu of all database extensions available on the CD-ROM.

b. Select the NetBackup for Oracle on UNIX Advanced BLI Agent option.

c. Enter **q** to quit selecting options.

   A prompt will appear asking if the list is correct.

d. Answer **y**.

   The following actions will occur:

- The version file, compressed tar file and `install_dbext` script will be loaded to directory *install_path*`/netbackup/dbext`.
- The `install` script will automatically execute the `install_dbext` script.
- After the install_dbext script unbundles the NetBackup Advanced BLI Agent compressed tar file, you will be prompted for the following configuration information.

  ```
  Please specify the UNIX database administrators' group for
  the Oracle software owner:
  ```

  Type the UNIX group to which the Oracle software owner belongs. In most cases, the group entered will be dba.
- If `install_dbext` has successfully completed, there will be a version file in directory *install_path*`/netbackup/ext/` that contains the version of NetBackup Advanced BLI Agent that was installed and an installation timestamp.

Install Oracle8 Recovery Manager

To configure the Oracle8 Recovery Manager, refer to the *Oracle8 Server Backup and Recovery Guide*.

To be able to write to sequential devices such as tape, Oracle must have access to a shared library provided by NetBackup Advanced BLI Agent. Oracle uses this shared library when it needs to write to or read from devices supported by Media Manager. Linking RMAN with a library provided by NetBackup for Oracle on UNIX is the only NetBackup requirement. Before writing to sequential devices such as tape, you must link the Oracle8 Server software with the media management API library installed by NetBackup for Oracle on UNIX.

The default location for the NetBackup API library is *install_path*`/netbackup/bin`.

The library name is

- `libobk.so.1` on Solaris
- `libobk.sl` on HP-UX (32-bit)

Linking Oracle 8 with NetBackup Advanced BLI Agent

This section documents the preferred way to link Oracle with the NetBackup API library. The linking process is not consistent across hardware platforms and Oracle Server release levels. On some platforms, it is not consistent across OS levels.

Run-time problems may occur if these linking instructions are not followed.

1. Become the Oracle user by typing the following command.

   ```
   su - oracle
   ```

2. Shut-down all of the Oracle instances (SIDs).

   ```
   svrmgrl
   connect internal
   shutdown
   exit
   ```

   Change the ORACLE_SID environment variable to any other SID that uses this Oracle installation and repeat above commands.

Starting with version 8.0.4, Oracle is shipped with default media management software. To disable this media management software, you must remove or rename the `libobk` file from *${ORACLE_HOME}*`/lib`. It may be a file or a symbolic link. To satisfy this and other linking requirements, follow the instructions below for your specific platform and version of Oracle.

- Solaris

  a. Set the Oracle user's library search path.

  ${ORACLE_HOME}*/lib must be in the Oracle user's library search path to prevent linking and run-time load errors. If *${ORACLE_HOME}/lib is not in the search path, the NetBackup API shared library (libobk) will not be found when linking and a run-time load error will occur when an application, such as svrmgrl, is executed. On Solaris, make sure that the environment variable, LD_LIBRARY_PATH, is properly set. If you are using the C shell, edit the Oracle user's .cshrc login file to include the line:

  ```
  setenv LD_LIBRARY_PATH=${ORACLE_HOME}/lib:$LD_LIBRARY_PATH
  ```

  Otherwise, edit the Oracle user's .profile login file to include the lines:

  ```
  LD_LIBRARY_PATH=${ORACLE_HOME}/lib:$LD_LIBRARY_PATH
  export LD_LIBRARY_PATH
  ```

  b. Create a symbolic link from Oracle to the NetBackup API library.

  ```
  cd ${ORACLE_HOME}/lib
  ls -l libobk.so
  If libobk.so exists.
  mv libobk.so libobk.so.orig

  ln -s /usr/openv/netbackup/bin/libobk.so.1 libobk.so
  ```

  This is the only required step for 8.1.5. You no longer have to make a new Oracle executable to use the NetBackup for Oracle API library. With Oracle8i, the Oracle executable always looks for the following shared library: *${ORACLE_HOME}*/lib/libobk.so.

  c. Make a new Oracle executable.

  This step is relevant only for **Oracle8 Release 8.1.6.**

  ```
  cd ${ORACLE_HOME}/rdbms/lib
  make -f ins_rdbms.mk ioracle LLIBOBK=-lobk
  ```

- HP-UX (32-bit)

  a. Set the Oracle user's library search path

  ${ORACLE_HOME}*/lib must be in the Oracle user's library search path to prevent linking and run-time load errors. If *${ORACLE_HOME}/lib is not in the search path, the NetBackup API shared library (libobk) will not be found when linking and a run-time load error will occur when an application, such as svrmgrl, is executed. On HP-UX, make sure that the environment variable, SHLIB_PATH, is properly set. If you are using the C shell, edit the Oracle user's .cshrc login file to include the line:

  ```
  setenv SHLIB_PATH=${ORACLE_HOME}/lib:$SHLIB_PATH
  ```

  Otherwise, edit the Oracle user's .profile login file to include the lines:

  ```
  SHLIB_PATH=${ORACLE_HOME}/lib:$SHLIB_PATH
  export SHLIB_PATH
  ```

  b. Create a symbolic link from Oracle to the NetBackup API library.

  ```
  cd ${ORACLE_HOME}/lib
  ls -l libobk.sl

  If libobk.sl exists.
  mv libobk.sl libobk.sl.orig

  ln -s /usr/openv/netbackup/bin/libobk.sl libobk.sl
  ```

  This is the only required step. You no longer have to make a new Oracle executable to use the NetBackup for Oracle API library. With Oracle8i, the Oracle executable always looks for the shared library: *${ORACLE_HOME}*/lib/libobk.sl.

Linking Rollback Procedure

If you run into problems and are unable to re-link Oracle with NetBackup Advanced BLI Agent, you can roll back what you have done by entering the following command:

```
cd $ORACLE_HOME/lib
mv libobk.so.orig libobk.so
```

(or whatever the API library file was originally called)

Install Oracle8 Recovery Catalog

The recovery catalog is a repository of information that is used and maintained by Oracle8 Recovery Manager. Oracle8 Recovery Manager uses the information in the recovery catalog to determine how to execute requested backup and restore actions.

The recovery catalog contains information about:

- Datafile and archivelog backup sets and backup pieces
- Datafile copies
- Archived redo logs and their copies
- Tablespaces and datafiles on the target database
- Stored scripts, which are named, user-created sequences of RMAN and SQL commands

Oracle recommends you use Oracle8 Recovery Manager with a recovery catalog, especially if you have 20 (or more) datafiles. However, you are not required to maintain a recovery catalog with Oracle8 Recovery Manager.

Because most information in the recovery catalog is also available in the target database's control file, Oracle8 Recovery Manager supports an operational mode where it uses the target database control file instead of a recovery catalog. This mode is appropriate for small databases where installation and administration of another database for the sole purpose of maintaining the recovery catalog would be burdensome.

If a recovery catalog is not used, the following features are not supported:

- Tablespace point-in-time recovery
- Stored scripts
- Restore and recovery when the control file is lost or damaged

Because recovery catalog installation procedures are version specific, refer to your *Oracle8 Server Backup and Recovery Guide* for details.

Configuration 3

Note If you are going to perform BLI Backups with RMAN after installing this software, refer to this chapter for configuration instructions.
If you are going to perform BLI Backups without RMAN after installing this software, refer to *NetBackup for Oracle Advanced BLI Agent System Administrator's Guide for Backups Without RMAN*.

Before attempting to configure NetBackup Advanced BLI Agent, complete the installation procedure as described in the Installation chapter.

The following is the configuration procedure.

1. Configure Media Manager.
2. Add Oracle Classes to NetBackup
3. Configure Run-Time Environment
4. Create Scripts
5. Test NetBackup for Oracle on UNIX Configuration Settings

The following sections in this chapter describe each of these steps in detail.

Configure Media Manager

Use the Media Manager to configure tapes or other storage units for a NetBackup for Oracle on UNIX configuration.

- Refer to the *Media Manager for NetBackup System Administrator's Guide - UNIX* if the NetBackup server is UNIX.
- Refer to the *Media Manager for NetBackup System Administrator's Guide - Windows NT/2000* if the NetBackup server is Windows NT/2000.

The number of volumes required will depend on the devices used, the size of the Oracle databases that you are backing up and the frequency of backups.

Add Oracle Classes to NetBackup

This section provides an overview of how to configure NetBackup to perform backup and restore operations.

To use NetBackup for Oracle on UNIX, you must add at least one Oracle class to NetBackup, then define the appropriate schedules for that class. This section contains the following:

- Issues to remember while configuring a class for NetBackup for Oracle on UNIX.
- Class configuration procedures for the NetBackup Java Interface and the NetBackup Windows NT/2000 interface

Most requirements for Oracle classes are the same as for file system backups. Refer to the *NetBackup System Administrator's Guide - UNIX* or the *NetBackup System Administrator's Guide - Windows NT/2000* for detailed configuration instructions.

Some issues of importance are:

- Expiration of Backup Files
- Environment Variables in a Scheduled Backup
- Example Scripts
- Maximum Jobs per Client Global Attribute
- Class Configuration Procedures

Expiration of Backup Files

NetBackup and Oracle keep backup image information in catalogs. Currently, automatic expiration of backup images from both catalogs is not supported.

The following describes how to expire images and synchronize the NetBackup and the Oracle catalogs.

Expiration of Backup Images from the NetBackup Catalog Using Retention Level

NetBackup automatically controls the expiration of the Oracle backup images from its catalog using the retention setting in a class schedule.

For non-proxy backups, use the *Backup Policy* schedule retention setting to specify the length of time before NetBackup expires a backup image.

For proxy backups, use the corresponding automatic backup schedule retention setting to specify the length of time before NetBackup expires a backup image.

When you use the retention setting to expire backup images, you must perform periodic Oracle catalog maintenance to remove catalog references to expired backup files.

Oracle8 Recovery Manager Catalog Maintenance

Use RMAN catalog maintenance commands to remove references to backup files. You can use these commands to delete backup image information from both the Oracle recovery catalog and the NetBackup catalog. For more information on the catalog maintenance commands, refer to “Maintaining a Recovery Catalog” on page 118.

For non-proxy backups, when a request is issued to delete a backup file from the recovery catalog, RMAN sends the request to NetBackup Advanced BLI Agent to delete the corresponding image from the NetBackup catalog, regardless of the retention level.

In contrast, all requests to expire proxy backups are ignored by NetBackup Advanced BLI Agent. To restore proxy backups, NetBackup needs to have a full backup and all subsequent incremental backups. To insure that NetBackup has all of the backup images it needs, all requests to expire are ignored. The only way to expire proxy backup images from the NetBackup catalog is through the automatic backup schedule retention level.

Environment Variables in a Scheduled Backup

When a schedule is executed, NetBackup sets environment variables based on which automatic backup schedule is selected. These environment variables can then be used by Oracle scripts to conditionally start a backup. Refer to “Environment Variables Set Up by NetBackup for Oracle on UNIX” on page 77.

Example Scripts

The scripts provided with this product are not intended for production. You must modify these scripts to suit your needs.

Caution NetBackup does not interpret a NetBackup for Oracle script. Be sure to use the correct NetBackup for Oracle script name in the file list to prevent an error or possibly a wrong operation.

Refer to "Create Scripts" on page 82 for instructions on using the example scripts to create scripts of your own.

Maximum Jobs per Client Global Attribute

The Maximum jobs per client global attribute value is figured with the following formula.

`Max Jobs per Client =` *Number of Streams* `x` *Number of Classes*

Where:

- *Number of Streams* is the number of backup jobs between the database server and NetBackup. Each separate stream starts a new backup job on the client.
- *Number of Classes* is the number of classes that may back up this client at the same time. This number can be greater than one. For example, a client may be in two classes in order to back up two different databases. These backup windows may overlap.

For Oracle non-proxy backups and restores, the number of jobs is hard to determine, as Oracle internally determines when and how many streams to run in parallel to optimize performance. For Oracle proxy backups, the number of streams is determined by two variables: NB_ORA_PC_FILESYSTEMS and NB_ORA_PC_STREAMS. Refer to "Environment Variables Set Up by a User" on page 78 for details.

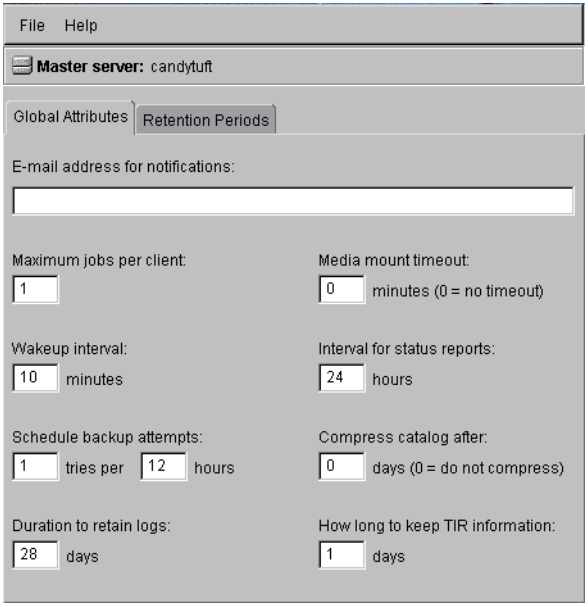
To avoid any problems, we recommend that you enter a value of 99 for the Maximum jobs per client global attribute.

NetBackup Administration - Java Interface

Use this procedure to set the Maximum Jobs per Client global attribute on the NetBackup Administration - Java Interface for HP or Solaris operating systems.

1. On the Configure menu in the NetBackup Administration dialog box, click NetBackup System Configuration. The System Configuration dialog box appears.

2. In the Master Server dialog box, click the Global Attributes tab.



   The default value is 1 for Maximum jobs per client.

3. Change the Maximum jobs per client value to a value equal to the maximum number of backups allowed per client.

Tip To avoid any problems, we recommend that you enter a value of 99 for the Maximum jobs per client global attribute.

xbpadm Interface

Use this procedure to set the Maximum Jobs per Client global attribute on a UNIX NetBackup master server.

1. Log onto the server as root.
2. Start the NetBackup xbpadm administrator interface.
   - If the DISPLAY variable is set, type:

     ```
     /usr/openv/netbackup/bin/goodies/xbpadm &
     ```

   - If the DISPLAY variable is not set, use the -d option:

     ```
     /usr/openv/netbackup/bin/goodies/xbpadm -d (your_machine_name):0 &
     ```

   The NetBackup Administration dialog box will open.
3. From the File menu, click Change NetBackup Configuration. The NetBackup Configuration dialog box will appear.

   The default value is 1 for Maximum jobs per client.
4. Change the Maximum jobs per client value to a value equal to the maximum number of backups allowed per client.

Tip To avoid any problems, we recommend that you enter a value of 99 for the Maximum jobs per client global attribute.

5. Click OK.

NetBackup Administration - Windows NT/2000 Interface

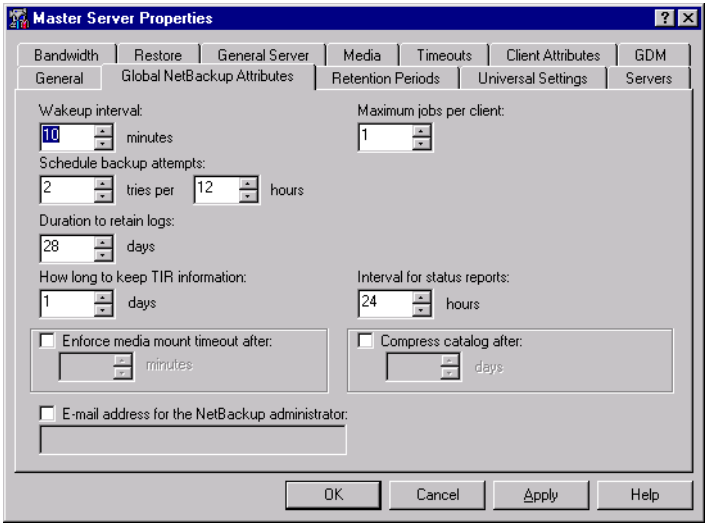
Use this procedure to set the Maximum Jobs per Client global attribute on a Windows NT / 2000 server or on the NetBackup Administration Client host.

1. On the Start menu in the NetBackup Administration window, click Configure NetBackup. The Configure-NetBackup dialog box appears.

2. In the left pane, right-click on the server and on the shortcut menu click Properties (Read/Write).

   The Master Server Properties dialog box appears.

3. In the Master Server Properties dialog box, click the Global NetBackup Attributes tab.



   The default value is 1 for Maximum jobs per client.

4. Change the Maximum jobs per client value to a value equal to the maximum number of backups allowed per client.

Tip To avoid any problems, we recommend that you enter a value of 99 for the Maximum jobs per client global attribute.

Class Configuration Procedures

NetBackup classes define the criteria for the backup. These criteria include:

- clients and the NetBackup for Oracle on UNIX script files to be executed on the clients
- storage unit and media to use
- backup schedules

Procedures in this section describe how to configure a class for NetBackup for Oracle on UNIX on a NetBackup server. There are other attributes for a class to consider. Refer to the *NetBackup System Administrator's Guide - UNIX* or the *NetBackup System Administrator's Guide - Windows NT/2000* for details on how to configure all the attributes.

NetBackup Administration - Java Interface

Use this procedure to configure a class on the NetBackup Administration - Java Interface on HP or Solaris operating systems.

1. Log onto the server as root.
2. Add the ALLOW_BLOCK_INCREMENTALS option to the `bp.conf` file on the NetBackup server. This will allow BLI Backups.

   For example:

   ```
   SERVER = netbackup-server
   ALLOW_BLOCK_INCREMENTALS
   ```

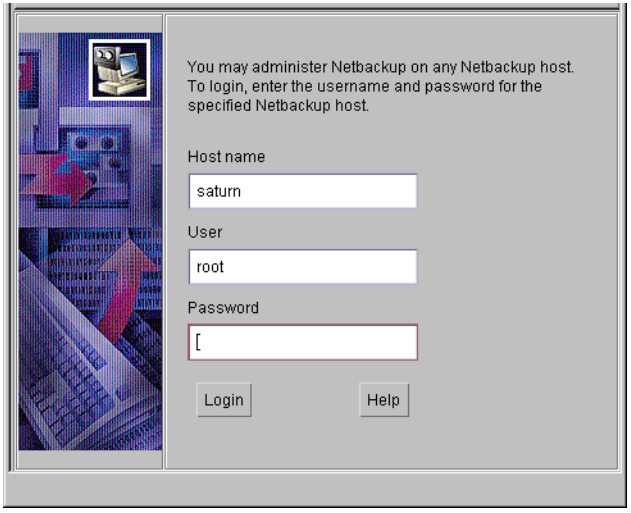
3. Start the NetBackup administrator interface by entering:

   install_path`/netbackup/bin/jnbSA &`

   For additional usage information, enter:

   `jnbSA -h`

The Login dialog box appears.

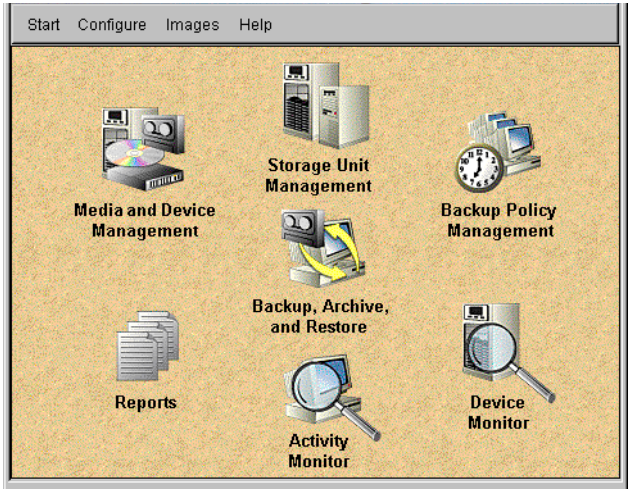


4. Type the password.

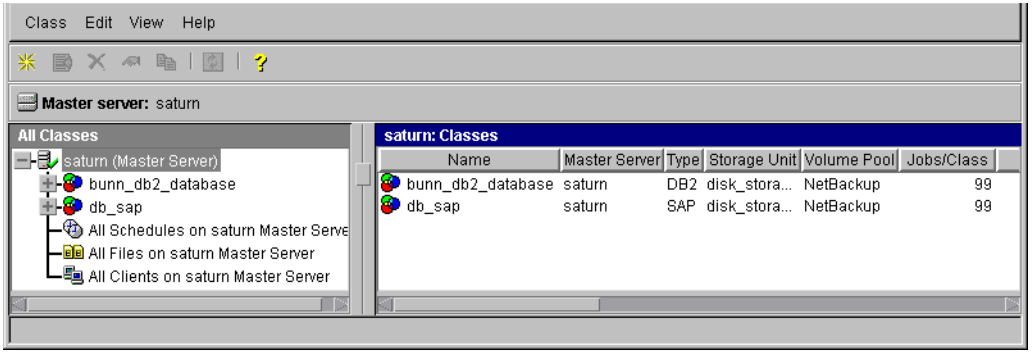
5. Press Login.

   The Login dialog closes and the NetBackup Assistant displays.

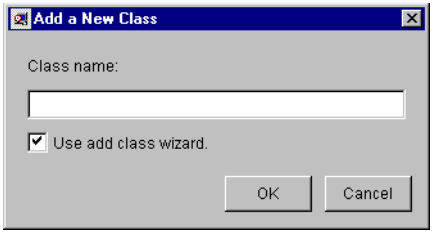
6. Click Close. The launch screen displays.



7. Click the Backup Policy Management icon. The Backup Policy Management (Classes) - NetBackup dialog appears.



8. On the Edit menu click New. The Add a New Class dialog box appears.

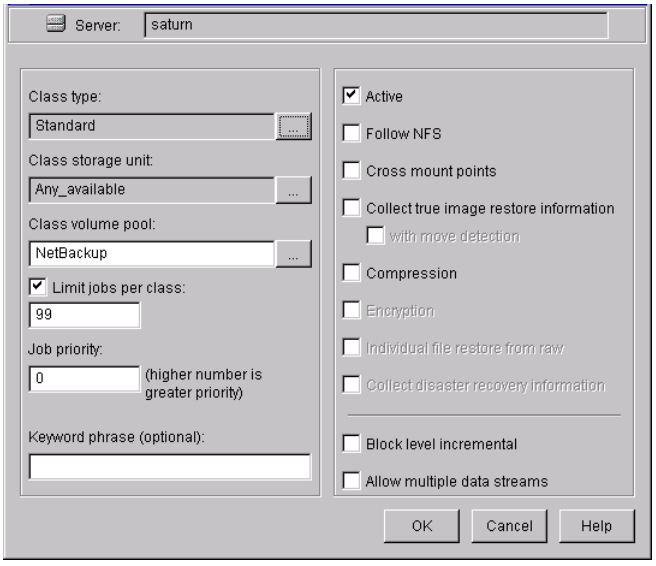


The class wizard automates the class configuration process. To configure classes without using the class wizard, use the following instructions.

9. Clear the Use add class wizard check box.

10. Type the new class name in the Class name box.

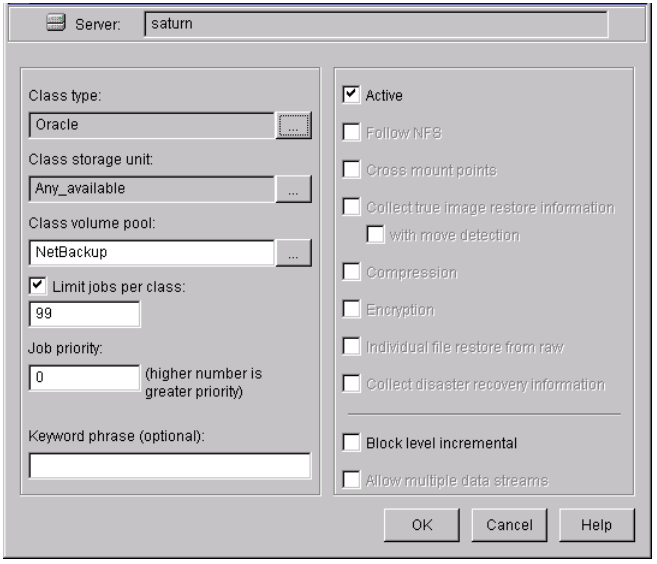
11. Click OK. The Change Attributes dialog box will appear.



a. Select the Oracle class type for Oracle.

b. Click OK.

The Change Attributes dialog box will change as follows.



Refer to the following table to configure class attributes.

Class storage unit:
: Select the storage unit for this class. A storage unit is a group of one or more storage devices configured to store information from a backup.

Class volume pool:
: Select the volume pool for this class. A volume pool is a group of volumes (removable media) configured for use by NetBackup only. These volumes are protected from being used by other applications.

Limit jobs per class:
: Type the maximum number of concurrent jobs for this class. If the Limit jobs per class checkbox is clear, the maximum number of backup and restore jobs that NetBackup will perform concurrently for this class can be up to a limit of 999. To specify a lower limit, select the checkbox and specify a value from 1 to 999 (the default is 99).

Job priority
: Select a value for the job priority NetBackup will assign to automatic backup jobs for this class. When a drive becomes available, NetBackup assigns it to the first client in the highest priority class.

Keyword phrase:
: For NetBackup for Oracle on UNIX, the keyword phrase entry is ignored.

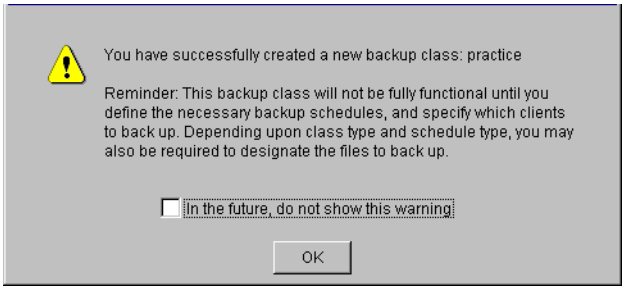
Active

Select the checkbox to perform scheduled operations defined in this class. The class must be active for NetBackup to execute automatic backup schedules or allow user backups or archives.

Block level incremental

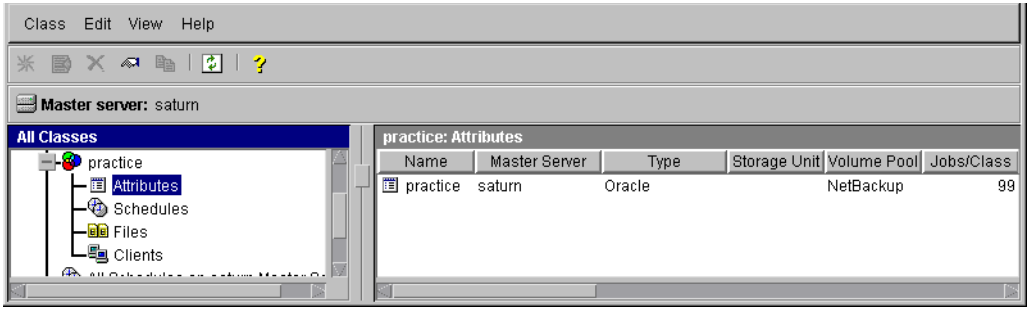
Select the checkbox to enable BLI Backups.

c. Click OK to close the Change Attributes dialog box. The following Warning will appear.



d. Click OK to close the Warning box.

Notice that the newly created class appears in the All Master Servers pane in the Backup Policy Management (Classes) - NetBackup dialog box.



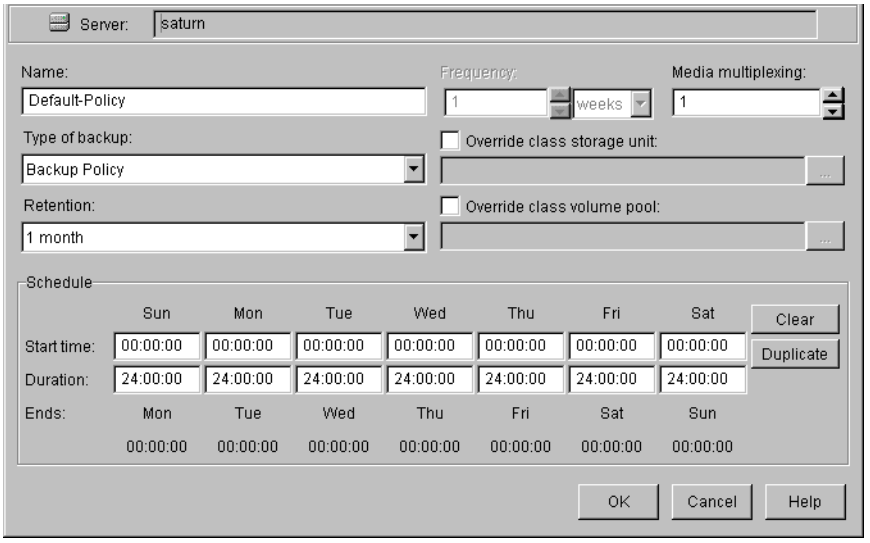
Also notice that the configuration settings you entered in the Change Attributes dialog box are displayed in the *class: Attributes* pane. Use the scroll bar at the bottom of the *class: Attributes* pane to view all settings.

12. Refer to the following instructions to configure schedules for your class.

a. Click Schedules in the All Master Servers pane of the Backup Policy Management (Classes) - NetBackup dialog box.

Notice that a Default-Policy appears in the practice:Schedules pane.

b. Double click the Default-Policy schedule. The Change Schedules dialog box will appear.



c. Configure a *Backup Policy* schedule.

All Oracle non-proxy database operations are performed through NetBackup for Oracle on UNIX using a *Backup Policy* schedule. This includes those backups started automatically.

You must configure a *Backup Policy* schedule for each Oracle class you create. If you do not do this, you will not be able to perform a non-proxy backup. To help satisfy this requirement, a *Backup Policy* schedule named Default-Policy is automatically created when you configure a new class.

Refer to the following table when configuring a Backup Policy schedule.

Name:

Each schedule requires a unique name.

Type of backup:

A *Backup Policy* schedule enables user-controlled NetBackup non-proxy operations performed on the client.
At least one *Backup Policy* schedule must be configured in each Oracle class. The Default-Policy schedule is configured as a *Backup Policy* schedule.

Retention:

The retention period for a *Backup Policy* schedule refers to the length of time that NetBackup keeps backup images. Set the time period to retain at least two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on. For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Note Oracle is not notified when NetBackup expires a backup image. You must use Oracle catalog maintenance commands to periodically delete expired backup sets from the Oracle catalog.

Start:

Specifies the day and time when the backup windows will open.

Duration:

Specifies the period of time (backup window) during which the backup job can take place.

The backup window for a *Backup Policy* schedule must encompass the time period during which all NetBackup jobs, scheduled and unscheduled, will occur. This is necessary because the *Backup Policy* schedule starts processes that are required for all non-proxy NetBackup for Oracle on UNIX backups, including those started automatically.

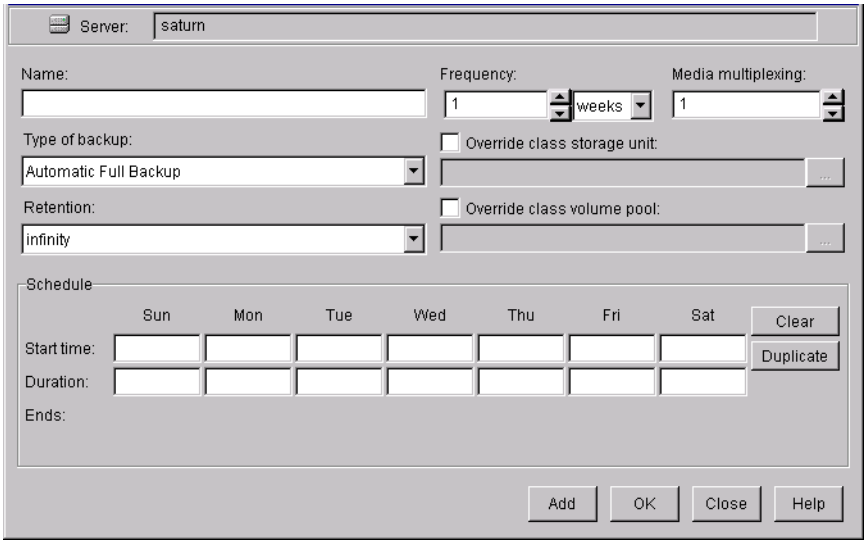
For example, assume that you:
- expect users to perform NetBackup operations during business hours, 0800 to 1300.
- configured automatic backups to start between 1800 and 2200.

The *Backup Policy* schedule must have a start time of 0800 and a duration of 14 hours.

Tip Set the time period for the *Backup Policy* schedule for 24 hours per day, seven days per week. This will ensure that your NetBackup for Oracle on UNIX operations are never locked out due to the *Backup Policy* schedule.

d. Configure an automatic backup.

Double click on Schedules in the All Master Servers pane of the Backup Policy Management (Classes) - NetBackup dialog box. The Add Schedule - Class *classname* property sheet will appear.



Refer to the following table to configure an automatic backup.

Name:

Each schedule requires a unique name.

Type of backup:

An *Automatic Full Backup, Automatic Differential Incremental Backup, or Automatic Cumulative Incremental Backup* schedule specifies the dates and times when NetBackup will automatically start backups by running the NetBackup for Oracle scripts in the order that they appear in the file list. If there is more than one client in the Oracle class, the NetBackup for Oracle scripts are executed on each client.
This schedule is also responsible for running the actual proxy copy BLI backups.

Retention:

NetBackup Advanced BLI Agent uses this retention period to specify how long to keep proxy backup images.

The retention period refers to the length of time that NetBackup keeps backup images. Set the time period needed to retain at lease two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on.

For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Frequency

Refers to the time period to wait between backups.

Start:

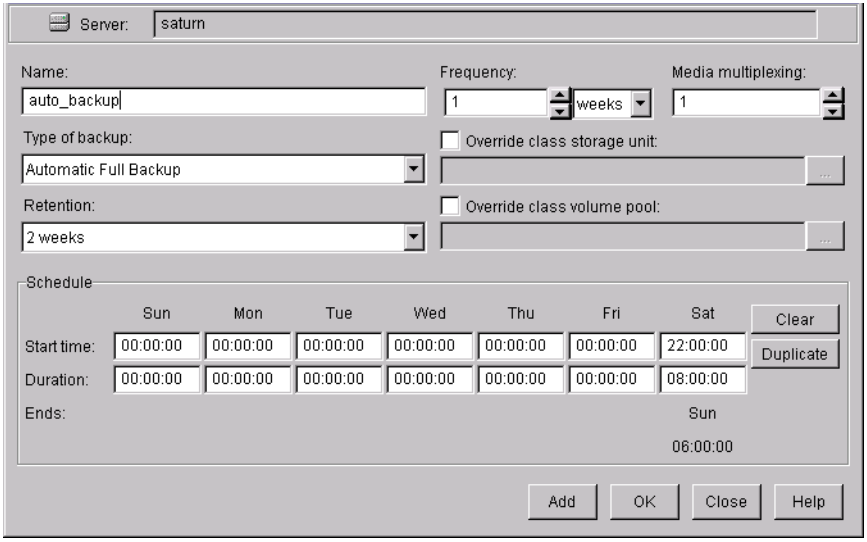
Specifies the day and time when the backup windows will open.

Duration:

Specifies the period of time (backup window) during which the backup job can take place.

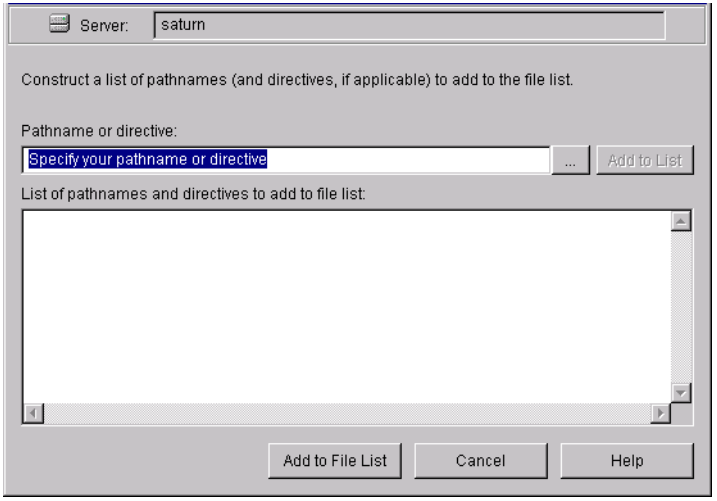
These settings define the backup window, which is the time period during which backups can occur for this schedule. The start time defines the times and days of the week when the window opens. The duration defines how long the window stays open.

The following illustrates how an automatic backup schedule might be configured.



13. Refer to the following instructions to configure the list of NetBackup for Oracle scripts.

a. Double click on Files in the All Master Servers pane of the Backup Policy Management (Classes) - NetBackup dialog box. The Add File Class will appear.



The File list in a database class has a different meaning than for other classes. Normally, in a Standard class, you would list files and directories to be backed up. But since you are now configuring a database class, you will list NetBackup for Oracle scripts.

Refer to “Create Scripts” on page 82 for details.

b. Type in the NetBackup for Oracle script. Specify the full pathname.

For example:

install_path`/netbackup/ext/db_ext/oracle/scripts/`*script_name*`.sh`

c. Click Add.

Since all NetBackup for Oracle scripts specified in the file list execute during automatic backups, you must make sure that only one type of backup is executed on the same database.

To make sure the correct script gets executed, use the environment variables set up by NetBackup when it initiates an automatic backup schedule. Code in the NetBackup for Oracle script can use the environment variables to determine what RMAN command file to use for the backup.

For example:

If an automatic cumulative incremental type of backup is executed, the NetBackup for Oracle script would use an RMAN script containing commands to perform a cumulative incremental backup. Only one NetBackup for Oracle script is necessary to handle different schedules.

For more details on using environmental variables with automatic backup, refer to "Environment Variables Set Up by NetBackup for Oracle on UNIX" on page 77.

14. Refer to the following instructions to configure the Client list.

a. Double click on Clients in the All Master Servers pane of the Backup Policy Management (Classes) - NetBackup dialog box. The Add Client Class dialog box will appear.

b. Type the name of the client. This client should have:

- the database installed
- NetBackup for Oracle on UNIX installed
- the backup or restore NetBackup for Oracle script(s)

c. Click Add to add the client to the client list.

d. Click OK.

The Add Client Class dialog box will close. The NetBackup Administration dialog box will remain open.

xbpadm Interface

Use this procedure to configure a class on a UNIX NetBackup master server.

1. Log onto the server as root.
2. Add the ALLOW_BLOCK_INCREMENTALS option to the bp.conf file on the NetBackup server. This will allow BLI Backups.

   For example:

   ```
   SERVER = netbackup-server
   ALLOW_BLOCK_INCREMENTALS
   ```

3. Start the NetBackup xbpadm administrator interface.
   - If DISPLAY variable is set, type:

     ```
     /usr/openv/netbackup/bin/goodies/xbpadm &
     ```

   - If the DISPLAY variable is not set, use the -d option:

   ```
   /usr/openv/netbackup/bin/goodies/xbpadm -d (your_machine_name):0 &
   ```

   The NetBackup Administration dialog box will open.
4. Create a new class.
   a. On the Actions menu, select New, then Classes. The Creating a Class dialog box will open.
   b. In the Class Name box, type the new class name.

      When you configure the Oracle class on your NetBackup installation, you will use a unique class name. For this procedure, we are going to use the word ***practice*** as the class name.

   c. Under Select one of, select New Class. The Class Type list box will enable.
   d. Select the Oracle class from the list box.
   e. Click OK. The Changing Class dialog box will open.
5. Check the Class Attribute settings.

   Refer to the following table to configure class attributes.

   Class storage unit:
   : Select the storage unit for this class. A storage unit is a group of one or more storage devices configured to store information from a backup.

   Class volume pool:
   : Select the volume pool for this class. A volume pool is a group of volumes (removable media) configured for use by NetBackup only. These volumes are protected from being used by other applications.

Limit jobs per class:
: Type the maximum number of concurrent jobs for this class. If the Limit jobs per class checkbox is clear, the maximum number of backup and restore jobs that NetBackup will perform concurrently for this class can be up to a limit of 999. To specify a lower limit, select the checkbox and specify a value from 1 to 999 (the default is 99).

Job priority
: Select a value for the job priority NetBackup will assign to automatic backup jobs for this class. When a drive becomes available, NetBackup assigns it to the first client in the highest priority class.

Keyword phrase:
: For NetBackup for Oracle on UNIX, the keyword phrase entry is ignored.

Active
: Select the checkbox to perform scheduled operations defined in this class. The class must be active for NetBackup to execute automatic backup schedules or allow user backups or archives.

Block level incremental
: Select the checkbox to enable BLI Backups.

6. Refer to the following instructions to configure the schedules for your class.

a. Click Schedules to change the display.

b. Click New to open the Creating a Schedule dialog box.

c. In the Name of Schedule box, type the new schedule name.

d. Click OK. The Creating a Schedule dialog box will open.

e. Configure a *Backup Policy* schedule.

All Oracle non-proxy database operations are performed through NetBackup for Oracle on UNIX using a *Backup Policy* schedule. This includes those backups started automatically.

You must configure a *Backup Policy* schedule for each Oracle class you create. If you do not do this, you will not be able to perform a non-proxy backup. To help satisfy this requirement, a *Backup Policy* schedule named Default-Policy is automatically created when you configure a new class.

Refer to the following table when configuring *Backup Policy* schedules.

Name:
: Each schedule requires a unique name.

Type of backup:
: A *Backup Policy* schedule enables user-controlled NetBackup non-proxy operations performed on the client.

At least one *Backup Policy* schedule must be configured in each Oracle class. The Default-Policy schedule is configured as a *Backup Policy* schedule.

Retention:

The retention period for a *Backup Policy* schedule refers to the length of time that NetBackup keeps backup images. Set the time period to retain at least two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on. For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Note Oracle is not notified when NetBackup expires a backup image. You must use Oracle catalog maintenance commands to periodically delete expired backup sets from the Oracle catalog.

Media Multiplexing

The media multiplexing box sets the number of jobs from this schedule that NetBackup can multiplex onto any one drive.

Start:

Specifies the day and time when the backup windows will open.

Duration:

Specifies the period of time (backup window) during which the backup job can take place.

The backup window for a *Backup Policy* schedule must encompass the time period during which all NetBackup jobs, scheduled and unscheduled, will occur. This is necessary because the *Backup Policy* schedule starts processes that are required for all non-proxy NetBackup for Oracle on UNIX backups, including those started automatically.

For example, assume that you:
- expect users to perform NetBackup operations during business hours, 0800 to 1300.
- configured automatic backups to start between 1800 and 2200.

The *Backup Policy* schedule must have a start time of 0800 and a duration of 14 hours.

Tip Set the time period for the *Backup Policy* schedule for 24 hours per day, seven days per week. This will ensure that your NetBackup for Oracle on UNIX operations are never locked out due to the *Backup Policy* schedule.

f. Configure an automatic backup.

Refer to the following table when configuring *Automatic Full Backup, Automatic Differential Incremental Backup, or Automatic Cumulative Incremental Backup* schedules.

Name:

Each schedule requires a unique name.

Type of backup:

An *Automatic Full Backup, Automatic Differential Incremental Backup, or Automatic Cumulative Incremental Backup* schedule specifies the dates and times when NetBackup will automatically start backups by running the NetBackup for Oracle scripts in the order that they appear in the file list. If there is more than one client in the Oracle class, the NetBackup for Oracle scripts are executed on each client.
This schedule is also responsible for running the actual proxy copy BLI backups.

Retention:

NetBackup Advanced BLI Agent uses this retention period to specify how long to keep proxy backup images.
The retention period refers to the length of time that NetBackup keeps backup images. Set the time period needed to retain at lease two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on.
For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Frequency

Refers to the time period to wait between backups.

Start:

Specifies the day and time when the backup windows will open.

Duration:

Specifies the period of time (backup window) during which the backup job can take place.
These settings define the backup window, which is the time period during which backups can occur for this schedule. The start time defines the times and days of the week when the window opens. The duration defines how long the window stays open.

7. Refer to the following instructions to configure the Files list for your class.

Perform this procedure if unattended schedule backups are going to be performed. Otherwise this step can be skipped.

a. Click Files.

The File list in a database class has a different meaning than for other classes. Normally, in a Standard class, you would list files and directories to be backed up. But since you are now configuring a database class, you will list NetBackup for Oracle scripts.

Refer to "Create Scripts" on page 82 for more details on scripts.

b. Specify the full path name for a NetBackup for Oracle script in the file list. For example:

install_path`/netbackup/ext/db_ext/oracle/samples/rman/cold_database_backup.sh`

Note The scripts must be modified to suit your needs. Also, the scrtipts must be installed on each machine that is in the client list.

Since all NetBackup for Oracle scripts specified in the file list execute during automatic backups, you must make sure that only one type of backup is executed on the same database.

To make sure the correct script gets executed, use the environment variables set up by NetBackup when it initiates an automatic backup schedule. Code in the NetBackup for Oracle script can use the environment variables to determine what RMAN command file to use for the backup.

For example:

If an automatic cumulative incremental type of backup is executed, the NetBackup for Oracle script would use an RMAN script containing commands to perform a cumulative incremental backup. Only one NetBackup for Oracle script is necessary to handle different schedules.

For more details on using environmental variables with automatic backup, refer to "Environment Variables Set Up by NetBackup for Oracle on UNIX" on page 77.

8. Refer to the following instructions to configure the Clients list for your class.

   a. Click Clients to change the display.

   b. Click New to open the Adding Clients dialog box.

   c. Select the hardware and operating system from the Hardware and Operating System: scroll box.

   d. Type the name of the client. This client should have:

      - the database installed
      - NetBackup for Oracle on UNIX installed
      - the backup or restore NetBackup for Oracle script(s)

Note The Install NetBackup Client Software checkbox will install NetBackup client software on a remote client. There is no option to install NetBackup for Oracle on UNIX software. Refer to the installation instructions in this guide to install NetBackup for Oracle on UNIX.

   e. Click OK.

9. Click OK.

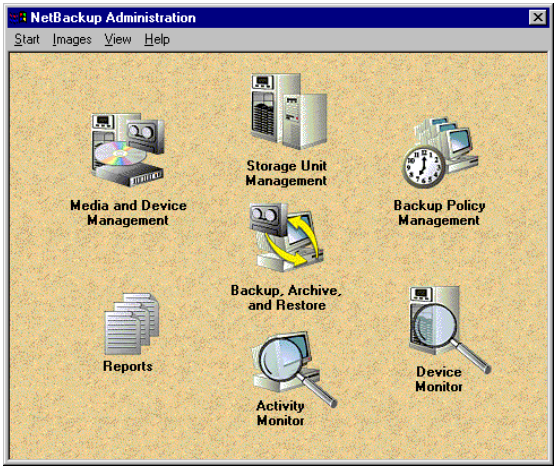
   The Changing Class dialog box will close. The NetBackup Administration dialog box will remain open.

NetBackup Administration - Windows NT/2000 Interface

Use this procedure when configuring a class from a Windows NT / 2000 server or from the NetBackup Administration Client host.

1. Log onto the server as Administrator.

2. From the Start menu, select Programs, VERITAS NetBackup, NetBackup Administration. The NetBackup Administration interface appears.

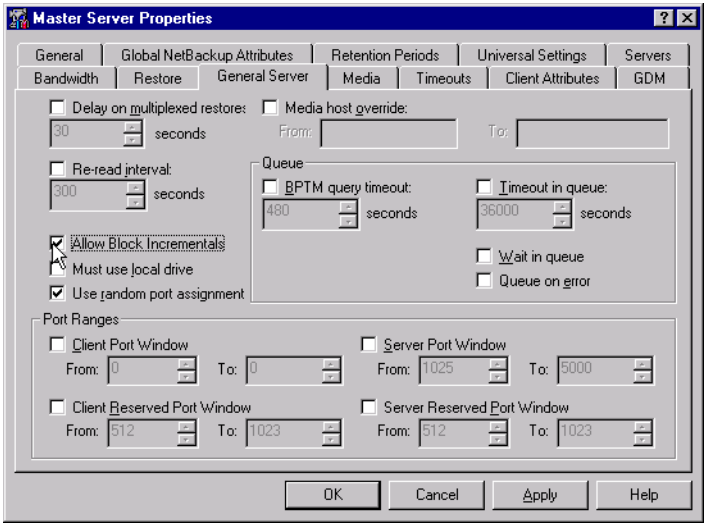


3. On the Start menu in the NetBackup Administration window, click Configure NetBackup. The Configure-NetBackup dialog box appears.

4. In the left pane, right-click on the server and on the shortcut menu click Properties (Read / Write).

   The Master Server Properties dialog box appears.

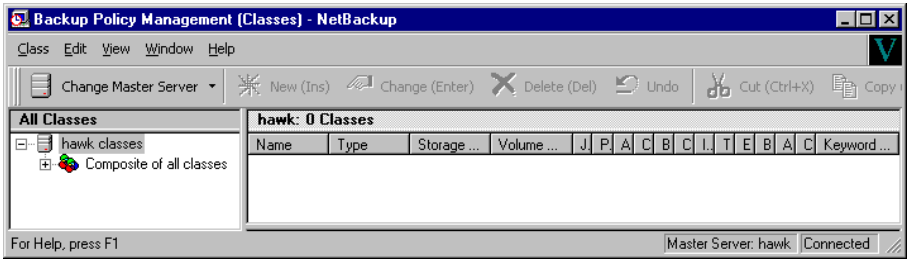
5. In the Master Server Properties dialog box, click the General Server tab and click Allow Block Incrementals.



6. Click OK to close the Master Server Properties dialog box.

7. Click the Backup Policy Management icon.

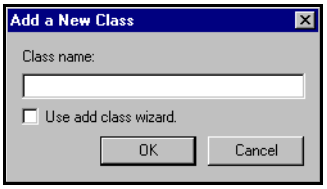
   The Backup Policy Management (Classes) - NetBackup dialog appears.



8. Perform the following steps to add a new class.

The class wizard automates the class configuration process. To configure classes without using the class wizard, perform the following steps.

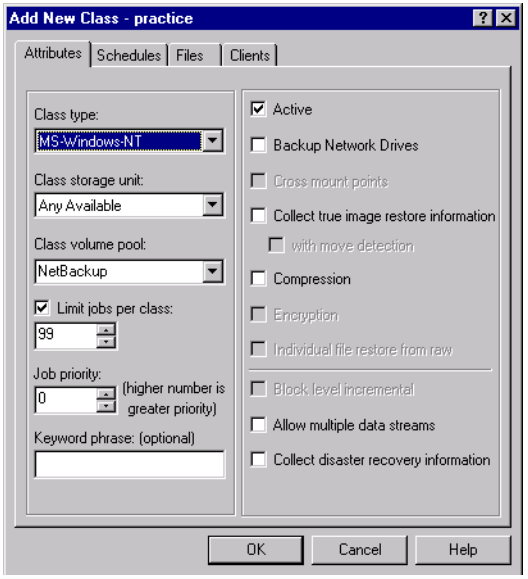
a. On the Class menu click New. The Add a New Class dialog box appears.



b. Confirm that the Use add class wizard check box is not checked.

c. Type the new class name in the Class name box.

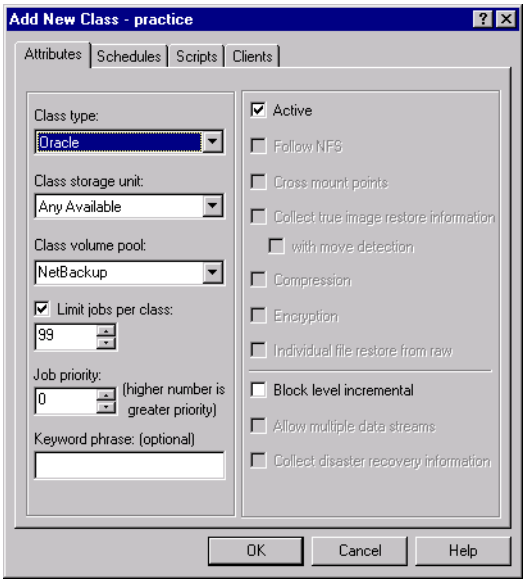
d. Click OK. The Add New Class dialog box appears. The class name you specified appears in the title bar.



9. Use the following instructions to configure the general attributes for the class.

a. Select the Oracle class type.

The Add New Class dialog box will change as follows.



Refer to the following table to configure class attributes.

Class storage unit:
: Select the storage unit for this class. A storage unit is a group of one or more storage devices configured to store information from a backup.

Class volume pool:
: Select the volume pool for this class. A volume pool is a group of volumes (removable media) configured for use by NetBackup only. These volumes are protected from being used by other applications.

Limit jobs per class:
: Type the maximum number of concurrent jobs for this class. If the Limit jobs per class checkbox is clear, the maximum number of backup and restore jobs that NetBackup will perform concurrently for this class can be up to a limit of 999. To specify a lower limit, select the checkbox and specify a value from 1 to 999 (the default is 99).

Job priority
: Select a value for the job priority NetBackup will assign to automatic backup jobs for this class. When a drive becomes available, NetBackup assigns it to the first client in the highest priority class.

Keyword phrase:
: For NetBackup for Oracle on UNIX, the keyword phrase entry is ignored.

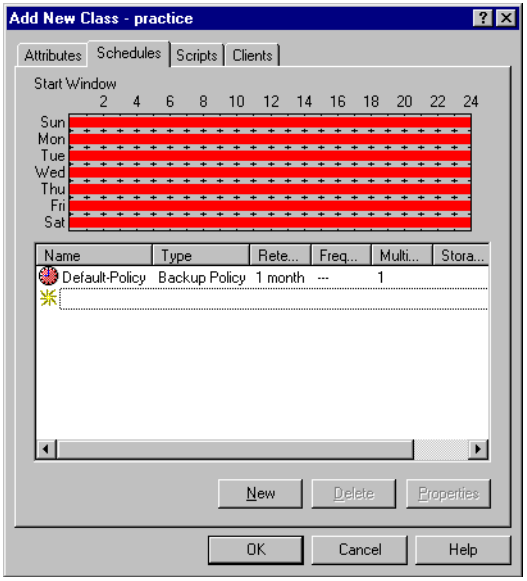
Active
: Select the checkbox to perform scheduled operations defined in this class. The class must be active for NetBackup to execute automatic backup schedules or allow user backups or archives.

Block level incremental
: Select the checkbox to enable BLI Backups.

b. Select the Block level incremental checkbox.

10. Use the following instructions to configure the class schedules.

a. Click Schedules tab. The Schedules property sheet appears.



Notice that a schedule named Default Policy of type *Backup Policy* has been created. This schedule can be used as is, or it can be modified to individual needs.

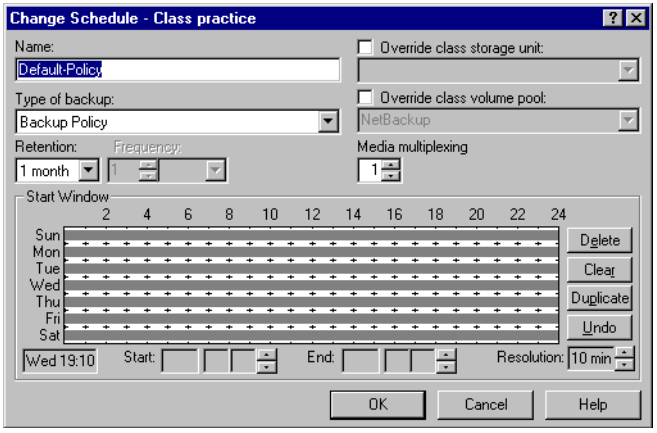
b. Configure a *Backup Policy* schedule.

All Oracle non-proxy database operations are performed through NetBackup for Oracle on UNIX using a *Backup Policy* schedule. This includes those backups started automatically.

You must configure a *Backup Policy* schedule for each Oracle class you create. If you do not do this, you will not be able to perform a non-proxy backup. To help satisfy this requirement, a *Backup Policy* schedule named Default-Policy is automatically created when you configure a new class.

c. Double-click on Backup Schedule in the Add New Class dialog box.

The Change Schedules dialog box appears.



Refer to the following table when configuring *Backup Policy* schedules.

Name:

Each schedule requires a unique name.

Type of backup:

A *Backup Policy* schedule enables user-controlled NetBackup non-proxy operations performed on the client.

At least one *Backup Policy* schedule must be configured in each Oracle class. The Default-Policy schedule is configured as a *Backup Policy* schedule.

Retention:

The retention period for a *Backup Policy* schedule refers to the length of time that NetBackup keeps backup images. Set the time period to retain at least two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on. For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Note Oracle is not notified when NetBackup expires a backup image. You must use Oracle catalog maintenance commands to periodically delete expired backup sets from the Oracle catalog.

Start:

Specifies the day and time when the backup windows will open.

End:

Specifies the day and time when the backup windows will close. These settings define the backup window, which is the time period during which backups can occur for this schedule. The start time defines the times and days of the week when the window opens. The duration defines how long the window stays open.

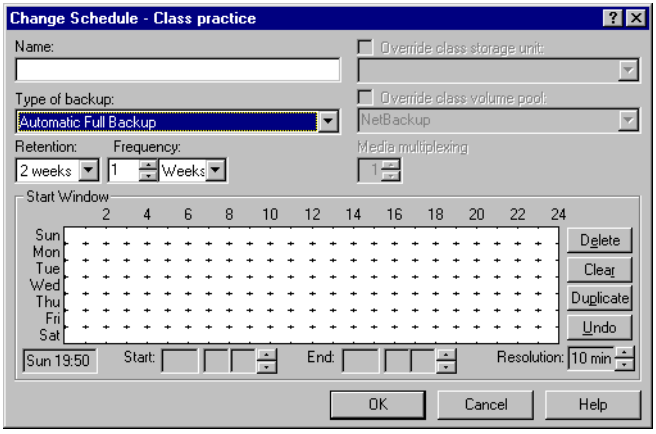
The backup window for a *Backup Policy* schedule must encompass the time period during which all NetBackup jobs, scheduled and unscheduled, will occur. This is necessary because the *Backup Policy* schedule starts processes that are required for all non-proxy NetBackup for Oracle on UNIX backups, including those started automatically.

For example, assume that you:
- expect users to perform NetBackup operations during business hours, 0800 to 1300.
- configured automatic backups to start between 1800 and 2200.

The *Backup Policy* schedule must have a start time of 0800 and a duration of 14 hours.

Tip Set the time period for the *Backup Policy* schedule for 24 hours per day, seven days per week. This will ensure that your NetBackup for Oracle on UNIX operations are never locked out due to the *Backup Policy* schedule.

d. Click New to configure an automatic backup schedule. The Change Schedules dialog box appears.



Refer to the following table when configuring *Automatic Full Backup, Automatic Differential Incremental Backup, or Automatic Cumulative Incremental Backup* schedules.

Name:

Each schedule requires a unique name.

Type of backup:

An *Automatic Full Backup, Automatic Differential Incremental Backup, or Automatic Cumulative Incremental Backup* schedule specifies the dates and times when NetBackup will automatically start backups by running the NetBackup for Oracle scripts in the order that they appear in the file list. If there is more than one client in the Oracle class, the NetBackup for Oracle scripts are executed on each client.

This schedule is also responsible for running the actual proxy copy BLI backups.

Retention:

NetBackup Advanced BLI Agent uses this retention period to specify how long to keep proxy backup images.

The retention period refers to the length of time that NetBackup keeps backup images. Set the time period needed to retain at lease two full backups of your database. In this way, if one full backup has been lost, you will have another full backup to fall back on.

For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.

Frequency

Refers to the time period to wait between backups.

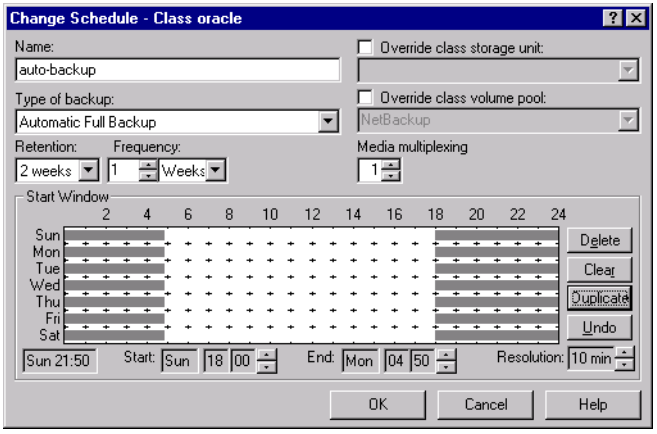
Start:

Specifies the day and time when the backup windows will open.

End:

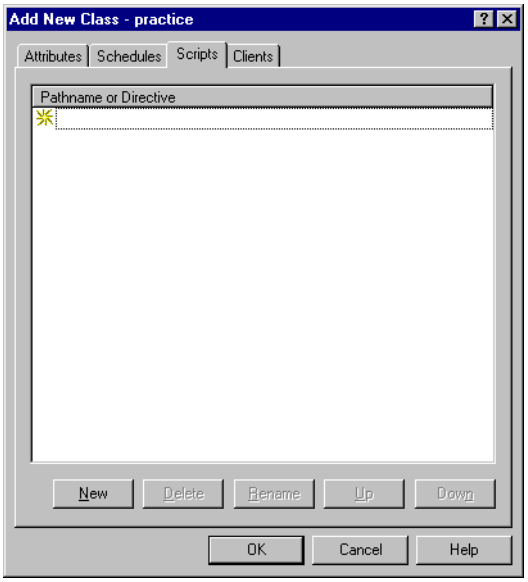
Specifies the day and time when the backup windows will close. These settings define the backup window, which is the time period during which backups can occur for this schedule. The start time defines the times and days of the week when the window opens. The duration defines how long the window stays open.

The following is an example of an automatic backup schedule.



11. Refer to the following instructions to configure the list of scripts.

a. Click Scripts tab. The Scripts property sheet appears.



The File list in a database class has a different meaning than for other classes. Normally, in a Standard class, you would list files and directories to be backed up. But since you are now configuring a database class, you will list NetBackup for Oracle scripts.

Refer to “Create Scripts” on page 82 for more details on scripts.

b. Click New.

c. Type the NetBackup for Oracle script. Specify the full pathname for the NetBackup for Oracle script in the file list.

For example:

install_path`/netbackup/ext/db_ext/oracle/samples/rman/cold_database_backup.sh`

d. Click Enter.

Since all NetBackup for Oracle scripts specified in the file list execute during automatic backups, you must make sure that only one type of backup is executed on the same database.

To make sure the correct script gets executed, use the environment variables set up by NetBackup when it initiates an automatic backup schedule. Code in the NetBackup for Oracle script can use the environment variables to determine what RMAN command file to use for the backup.

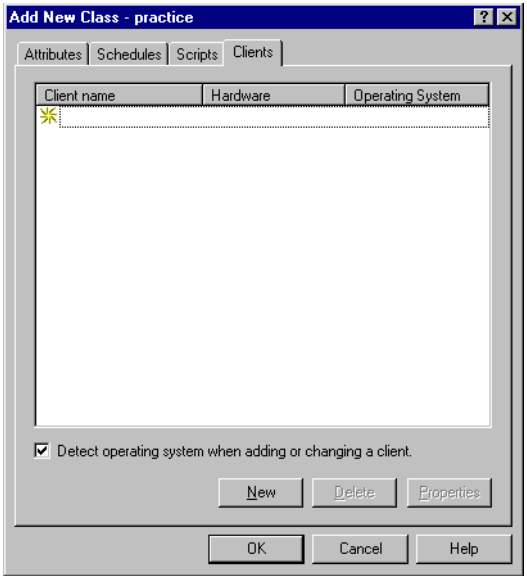
For example:

If an automatic cumulative incremental type of backup is executed, the NetBackup for Oracle script would use an RMAN script containing commands to perform a cumulative incremental backup. Only one NetBackup for Oracle script is necessary to handle different schedules.

For more details on using environmental variables with automatic backup, refer to “Environment Variables Set Up by NetBackup for Oracle on UNIX” on page 77.

12. Refer to the following instructions to configure the Client list.

a. Click Clients tab. The Clients property sheet appears.



b. Click New.

c. Type the name of the "Create Scripts" on page 82 client that has:

- the database installed
- NetBackup for Oracle on UNIX installed
- the backup or restore NetBackup for Oracle script

d. Click OK.

The Client Hardware and Operating System dialog box appears.

e. Select the hardware and operating system for the client.

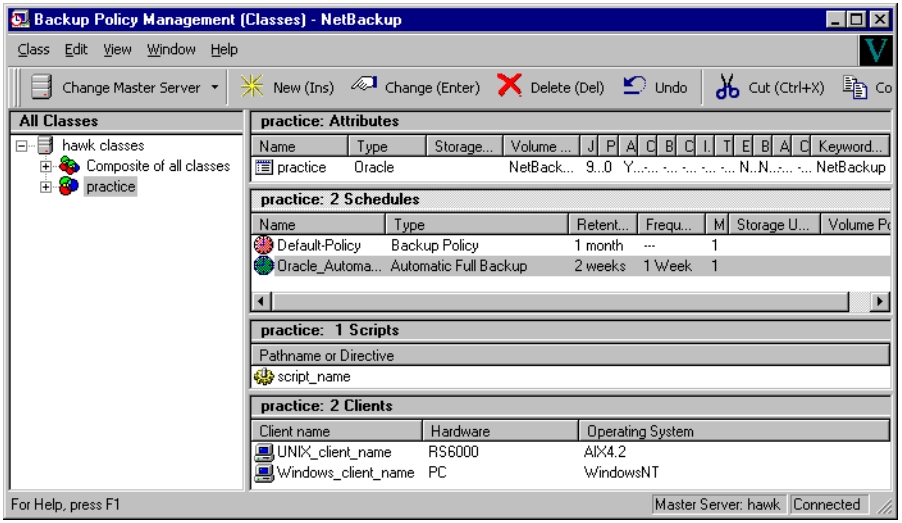
f. Click OK.

The Client Hardware and Operating System dialog box closes.

13. Click OK.

The Changing Class dialog box will close. The Backup Policy Management (Classes) - NetBackup dialog box will remain open.

Example Oracle Class on a NetBackup for Windows NT/2000 Server



Configure Run-Time Environment

The following is the order of precedence for the run-time configuration variable settings.

1. vendor-specific string with RMAN `send` command

   Refer to “RMAN send Command” on page 79 for details.

2. RMAN `parms` operand

   Refer to “RMAN parms Operand” on page 80 for details.

3. environment variables

   Refer to “Environment Variables Set Up by NetBackup for Oracle on UNIX” on page 77 and to “Environment Variables Set Up by a User” on page 78 for details.

4. user `bp.conf`

   Refer to “Create bp.conf” on page 81 for details.

5. master `bp.conf`

   Refer to “Create bp.conf” on page 81 for details.

Environment Variables Set Up by NetBackup for Oracle on UNIX

When a schedule executes, NetBackup sets environment variables for the local Oracle scripts to use when performing a NetBackup operation. These variables can be used to perform conditional operations inside the script. These variables are set only if the backup is started from the server (either automatically by the NetBackup scheduler or manually through the administrator interface).

NB_ORA_PC_SCHED
: Name of the Oracle schedule used for a proxy copy BLI backup.

NB_ORA_SERV
: Name of the NetBackup Server.

NB_ORA_CLASS
: Name of the Oracle class.

NB_ORA_FULL
: Set to 1 if this is an *Automatic Full Backup*.

NB_ORA_INCR
: Set to 1 if this is an *Automatic Differential Incremental Backup*.

NB_ORA_CINC
: Set to 1 if this is an *Automatic Cumulative Incremental Backup*.

NB_ORA_SCHEDULED
: Set to 1 if a scheduled backup (*Automatic Full Backup, Automatic Differential Incremental Backup,* or *Automatic Cumulative Incremental Backup*).

Refer to the following examples:

- “Example 1, cold_database_backup.sh” on page 83
- “Example 2, hot_database_backup.sh” on page 88
- “Example 3, cold_duplex_database_backup_full.sh” on page 91
- “Example 4, hot_database_backup_proxy.sh” on page 94

Environment Variables Set Up by a User

The NetBackup for Oracle on UNIX variables that follow can be set for use in the Oracle user environment.

NB_ORA_SERV
: Specifies the name of NetBackup master server.

NB_ORA_CLIENT
: Specifies the name of the Oracle client. It is especially useful for an alternate client restore operation.

NB_ORA_CLASS
: Specifies the name of the class to use for the Oracle backup.

NB_ORA_SCHED
: Specifies the name of the *Backup Policy* schedule to use for the Oracle backup.

NB_ORA_PC_SCHED
: Specifies the name of the schedule to use for the Oracle proxy backup. To insure that backup uses the correct schedule for proxy backups, use the `send` command as explained in "RMAN send Command" on page 79.

NB_ORA_PC_FILESYSTEMS
: Specifies the number of file systems that NetBackup Advanced BLI Agent will back up in parallel in each proxy copy session.
The default value is equal to the number of target file systems. Target file systems contain the datafiles to be backed up.
Use the RMAN `send` command to pass this variable as explained "RMAN send Command" on page 79.

NB_ORA_PC_STREAMS
: Specifies the number of backup streams per file system that NetBackup Advanced BLI Agent starts simultaneously in each proxy copy session. When a backup starts, NetBackup Advanced BLI Agent groups all datafiles by the file system. Based on the value of this variable, it then organizes the files into a specified number of backup streams based on the file size. NetBackup Advanced BLI Agent attempts to create streams of equal size.
The default value for `NB_ORA_PC_STREAMS` is 1.
Use the RMAN `send` command to pass this variable as explained "RMAN send Command" on page 79.

Multiple Stream Configuration Example

In this example there are three file systems:

FS1 with 10 MB of datafiles

FS2 with 1 GB of datafiles

FS3 with 10 MB of datafiles

A backup is configured using the following values:

NB_ORA_PC_FILESYSTEMS =2

NB_ORA_PC_STREAMS =3

These settings allow three backup streams per file system. Only two file systems can be backed up at a time. As a result, the following sequence will take place.

a. NetBackup Advanced BLI Agent will initiate parallel backups of FS1 and FS2. Each file system will have three backup streams. Overall, there will be six backup jobs running at the same time.

b. The backup of FS1 will finish before the backup of FS2, since FS1 contains only 10 MB of datafiles. The backup of FS2 will continue.

c. NetBackup Advanced BLI Agent will initiate the backup of FS3 when the backup of FS1 is finished. FS3 will have three backup streams. In other words, parallel backups of FS2 and FS3 will take place. Once again, there will be six backup jobs running at the same time.

Note When connecting to a database using a TNS alias, you must use a `send` command or a `parms` operand to specify environment variables. In other words, when accessing a database through a listener, the environment variables set at the system level are not visible when RMAN is running.

RMAN `send` Command

You can use the `send` command to send a vender-specific quoted string to NetBackup for Oracle on UNIX. This section describes the different ways to specify any of the variables.

- Specify the variables in the string in the RMAN script after all channels have been allocated and before the `backup` command.

```
run {
   allocate channel t1 type 'sbt_tape';
   allocate channel t2 type 'sbt_tape';
   send 'NB_ORA_CLASS=your_class, NB_ORA_SERV=your_server';
   send 'NB_ORA_PC_FILESYSTEMS=number_of_filesystems';
```

```
        send 'NB_ORA_PC_STREAMS=number_of_streams';
        backup
        (database format 'bk_%U_%t');
    }
```

- Use the `send device type` command to send the variables to all channels of the specified type.
- Use the `send channel` command to send the variables to specific channels.
- Specify variables in the string on the RMAN command line to send them to all allocated channels.

  Example 1:

  See Example 4, `hot_database_backup_proxy.sh` on how to use it in your Oracle script.

  Example 2:

  To call RMAN from the UNIX prompt:

${ORACLE_HOME}`/bin/rman nocatalog send "'NB_ORA_CLASS=`*your_class*`, NB_ORA_PC_SCHED=`*your_proxy_copy_schedule*`, NB_ORA_PC_STREAMS=`*number_of_streams*`, NB_ORA_PC_FILESYSTEMS=`*number_of_file_systems*`'"`

RMAN `parms` Operand

Use the RMAN command operand, `parms`, with each `allocate channel` command in the RMAN script to specify any of the variables.

The following example uses the `parms` operand to specify what class and server to use for a database backup. The `parms` are set per allocated channel.

```
run {
    allocate channel t1 type 'sbt_tape'
    parms="ENV=(NB_ORA_CLASS=your_class, NB_ORA_SERV=your_server)";
    allocate channel t2 type 'sbt_tape'
    parms="ENV=(NB_ORA_CLASS=your_class, NB_ORA_SERV=your_server)";
    backup
    (database format 'bk_%s_%p_%t');
}
```

For more information on the `send` command and `parms` operand of an `rman` command, see the *Oracle8 Server Backup and Recovery Guide*.

Create bp.conf

A NetBackup for Oracle on UNIX user can create an Oracle client bp.conf file in Oracle user's home directory on the NetBackup for Oracle on UNIX client host. When a NetBackup for Oracle on UNIX operation is started, the user's bp.conf file is searched before the master configuration file (*install_path*/netbackup/bp.conf) on the client. Any option found at the user level takes precedence over the same option in the master level configuration.

The following options can be set in the user's bp.conf file:

BPBACKUP_CLASS
: This option specifies the name of the class to use for the Oracle backup.

BPBACKUP_SCHED
: This option specifies the name of the Backup Policy type of schedule to use for the Oracle backup.

CLIENT_NAME
: This option specifies the name of the Oracle client. It is especially useful for an alternate client restore operation.

CLIENT_READ_TIMEOUT
: This option specifies the number of seconds the Oracle client will initially wait for a response from the NetBackup server. The default value is 900.

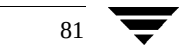
SERVER
: This option specifies the name of NetBackup master server.

VERBOSE
: This option causes NetBackup to include more information in its logs.

See the *NetBackup System Administrator's Guide - UNIX* for more details and default values.

The following shows example bp.conf entries for an Oracle user:

```
SERVER=jupiter
CLIENT_READ_TIMEOUT=1200
VERBOSE
```

Create Scripts

Script files must be created on the client before NetBackup for Oracle on UNIX backups can be performed automatically. This section describes the following types of scripts.

- NetBackup for Oracle script is necessary when an unattended scheduled backup is performed. These scripts are specified in the Oracle class File List. Refer to "NetBackup for Oracle Scripts" on page 82.
- RMAN script is necessary for an Oracle8 database. This script is specified as a parameter to the `rman` command. The database administrator must create this script if it does not already exist. Create this script *before* using the Oracle8 Recovery Manager's `rman` command. Refer to "RMAN Script" on page 98.

For more information on scripts, see the *Oracle8 Server Backup and Recovery Guide*.

NetBackup for Oracle Scripts

This section describes how to create a NetBackup for Oracle script for the Recovery Manager.

Create NetBackup for Oracle Script for Recovery Manager

When NetBackup for Oracle on UNIX was initially installed, example scripts were placed in the following directory:

install_path`/netbackup/ext/db_ext/oracle/samples/rman`

The Oracle example scripts installed are:

```
pit_database_restore.
hot_tablespace_backup.
hot_database_backup.
database_restore.
cold_duplex_database_backup_full.
cold_database_backup.
pit_database_restore.rcv
hot_tablespace_backup_level1_differential.rcv
hot_tablespace_backup_level1_cumulative.rcv
hot_tablespace_backup_level0.rcv
hot_database_backup_level1_differential.rcv
hot_database_backup_level1_cumulative.rcv
hot_database_backup_level0.rcv
database_restore.rcv
cold_duplex_database_backup_full.rcv
cold_database_backup_level1_differential.rcv
cold_database_backup_level1_cumulative.rcv
cold_database_backup_level0.rcv
```

```
hot_database_backup_proxy.sh
hot_tablespace_backup_proxy.sh
hot_database_backup_proxy.rcv
hot_tablespace_backup_proxy.rcv
```

The following scripts show how to perform proxy backups of a database:

- `hot_database_backup_proxy.sh`
- `hot_tablespace_backup_proxy.sh`
- `hot_database_backup_proxy.rcv`
- `hot_tablespace_backup_proxy.rcv`

1. Copy the example scripts to a different directory on your client. Oracle scripts can be located anywhere on the client.

2. Modify each script for your environment.

3. Make sure the `su` command logs into the correct user.

   If you do not include an `su - user` (*user* is Oracle administrator account) in your Oracle scripts, they will not run with the proper permissions and environment variables. The result will be problems with your database backups or restores.

Example 1, `cold_database_backup.sh`

This example sets the environment and calls `rman` with the appropriate command file to perform a whole database backup. It is used for both full backups and incremental backups. When a schedule executes, NetBackup sets environment variables that the script uses to perform the backup.

```
#!/bin/sh
  # $Header: cold_database_backup.sh,v 1.2 99/04/20 13:57:08 fma Stab $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  cold_database_backup.sh
  # ---------------------------------------------------------------------------
  #  This script uses Recovery Manager to take a cold (consistent) database
  #  backup. A cold backup is one where the database is shutdown cleanly before
  #  performing the backup. It is assumed that this script will be executed
  #  by user root. In order for the Oracle utilities, svrmgrl and rman, to
```

```
#  work properly we switch user (su -) to the oracle dba account before
#  each execution. If this script runs under a user account that has Oracle
#  dba privilege you should remove the switch user logic.
#
#  Note: As written, this script will bring down the target database and
#        open it when done.
# ------------------------------------------------------------------------

# ------------------------------------------------------------------------
# Put output in <this file name>.out. Change as desired.
# Note: output directory requires write permission.
# ------------------------------------------------------------------------

OUTF=${0}.out

# ------------------------------------------------------------------------
# You may want to delete the output file so that backup information does
# not accumulate.  If not, delete the following lines.
# ------------------------------------------------------------------------

if [ -f "$OUTF" ]
then
rm -f "$OUTF"
fi

{ # output block

echo "‘date‘ ----------------Beginning of Script------------"
echo "Script name: $0"

# ------------------------------------------------------------------------
# Replace /db/oracle/product/8.0.5, below, with the Oracle home path.
# ------------------------------------------------------------------------

ORACLE_HOME=/db/oracle/product/8.0.5
export ORACLE_HOME

# ------------------------------------------------------------------------
# Replace ORA805, below, with the Oracle SID of the target database.
# ------------------------------------------------------------------------

ORACLE_SID=ORA805
export ORACLE_SID

# ------------------------------------------------------------------------
# Replace ora805, below, with the Oracle DBA user id (account).
# ------------------------------------------------------------------------

ORACLE_USER=ora805

# ------------------------------------------------------------------------
# Replace ${ORACLE_HOME}/scripts with the NetBackup Oracle script path.
# Since subsequent installs or deinstalls will remove this file, you will
# want to move this script from its installed location before making updates.
```

```
# ------------------------------------------------------------------------

NB_ORA_SCRIPTS=${ORACLE_HOME}/scripts

# ------------------------------------------------------------------------
# Set the Oracle Server Manager name.
# ------------------------------------------------------------------------

SVRMGR=svrmgrl

# ------------------------------------------------------------------------
# Set the Oracle Recovery Manager name.
# ------------------------------------------------------------------------

RMAN=rman

# ------------------------------------------------------------------------
# Print out the value of the variables set by this script.
# ------------------------------------------------------------------------

echo
echo   "RMAN: $RMAN"
echo   "SVRMGR: $SVRMGR"
echo   "ORACLE_SID: $ORACLE_SID"
echo   "ORACLE_USER: $ORACLE_USER"
echo   "ORACLE_HOME: $ORACLE_HOME"
echo   "NB_ORA_SCRIPTS: $NB_ORA_SCRIPTS"

# ------------------------------------------------------------------------
# Print out the value of the variables set by bphdb.
# ------------------------------------------------------------------------

echo
echo   "NB_ORA_FULL: $NB_ORA_FULL"
echo   "NB_ORA_INCR: $NB_ORA_INCR"
echo   "NB_ORA_CINC: $NB_ORA_CINC"
echo   "NB_ORA_SERV: $NB_ORA_SERV"
echo   "NB_ORA_CLASS: $NB_ORA_CLASS"
echo   "NB_ORA_PC_SCHED: $NB_ORA_PC_SCHED"
echo   "NB_ORA_SCHEDULED: $NB_ORA_SCHEDULED"
echo   "NB_ORA_USER_INITIATED: $NB_ORA_USER_INITIATED"

# ------------------------------------------------------------------------
# Call Server Manager to shutdown the target database in immediate priority.
# This lets current calls to the database complete, but prevents further
# logons or calls.
#
# The shutdown-startup logic of this script can be commented out if you know
# that the database will always be mounted and in a consistent state before
# this script is executed.
# ------------------------------------------------------------------------

echo
echo "% ${ORACLE_HOME}/bin/$SVRMGR"
```

```
echo "connect internal"
echo "shutdown immediate"
echo "exit"
echo

su - $ORACLE_USER -c "
ORACLE_SID=$ORACLE_SID
export ORACLE_SID
${ORACLE_HOME}/bin/$SVRMGR <<!
connect internal
shutdown immediate
exit
!
"

# ------------------------------------------------------------------------
# Startup the database in case it had crashed or was not shutdown cleanly
# prior to starting this script.  This will perform a crash recover if
# it is needed. Use the RESTRICT option because we are going to shutdown
# again right away and we don't want to let users in during the short
# interval. The default init<SID>.ora file will be used by startup. If it was
# moved or the name was changed use the pfile= option on the startup command.
# ------------------------------------------------------------------------

echo
echo "% ${ORACLE_HOME}/bin/$SVRMGR"
echo "connect internal"
echo "startup restrict open"
echo "shutdown immediate"
echo "exit"
echo

su - $ORACLE_USER -c "
ORACLE_SID=$ORACLE_SID
export ORACLE_SID
${ORACLE_HOME}/bin/$SVRMGR <<!
connect internal
startup restrict open
shutdown immediate
exit
!
"

# ------------------------------------------------------------------------
# Now we know that the database is cleanly closed and is ready for a
# cold backup. RMAN requires that the database be started and mounted
# to perform a backup.
# ------------------------------------------------------------------------

echo
echo "% ${ORACLE_HOME}/bin/$SVRMGR"
echo "connect internal"
echo "startup mount"
echo "exit"
```

```
echo

su - $ORACLE_USER -c "
ORACLE_SID=$ORACLE_SID
export ORACLE_SID
${ORACLE_HOME}/bin/$SVRMGR <<!
connect internal
startup mount
exit
!
"

echo

if [ "$NB_ORA_INCR" = "1" ]
then
echo "cold database differential incremental backup requested"
CMDFILE=${NB_ORA_SCRIPTS}/cold_database_backup_level1_differential.rcv

elif [ "$NB_ORA_CINC" = "1" ]
then
echo "cold database cumulative incremental backup requested"
CMDFILE=${NB_ORA_SCRIPTS}/cold_database_backup_level1_cumulative.rcv

else # default
echo "cold database backup requested (incremental level 0)"
CMDFILE=${NB_ORA_SCRIPTS}/cold_database_backup_level0.rcv
fi

# ----------------------------------------------------------------------------
# Call Recovery Manager to initiate the backup. This example does not use a
# Recovery Catalog. If you choose to use one, remove the option, nocatalog,
# from the rman command line below and add a 'connect rcvcat' statement to
# the corresponding command file ($CMDFILE). An alternative to putting the
# connect statements in the command file would be to add them to the rman
# command line.
#
# Note: Any environment variables needed at run time by RMAN or $CMDFILE
#       must be set and exported within the switch user (su) command.
# ----------------------------------------------------------------------------

echo
echo "% ${ORACLE_HOME}/bin/$RMAN nocatalog cmdfile '$CMDFILE'"
echo

su - $ORACLE_USER -c "
ORACLE_SID=$ORACLE_SID
export ORACLE_SID
${ORACLE_HOME}/bin/$RMAN nocatalog cmdfile "$CMDFILE"
"
RETURN_STATUS=$?

echo
echo "'date' ----------------End of Script------------------"
```

```
echo
echo "exit $RETURN_STATUS"
exit $RETURN_STATUS

} >> $OUTF
```

Example 2, `hot_database_backup.sh`

This example sets up the environment and calls `rman` with the appropriate command file, to perform a whole database backup. It is used for both full backups and incremental backups. When a schedule executes, NetBackup sets environment variables that the script uses to perform the backup.

With the proper schedules this script could be used to automatically execute a backup every week on Friday night and a incremental backup each night for the rest of the week.

```
#!/bin/sh
  # $Header: hot_database_backup.sh,v 1.2 99/04/20 13:58:31 fma Stab $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  hot_database_backup.sh
  # ---------------------------------------------------------------------------
  #  This script uses Recovery Manager to take a hot (inconsistent) database
  #  backup. A hot backup is inconsistent because portions of the database are
  #  being modified and written to the disk while the backup is progressing.
  #  You must run your database in ARCHIVELOG mode to make hot backups. It is
  #  assumed that this script will be executed by user root. In order for RMAN
  #  to work properly we switch user (su -) to the oracle dba account before
  #  execution. If this script runs under a user account that has Oracle dba
  #  privilege you should remove the switch user logic.
  # ---------------------------------------------------------------------------

  # ---------------------------------------------------------------------------
  # Put output in <this file name>.out. Change as desired.
  # Note: output directory requires write permission.
  # ---------------------------------------------------------------------------

  OUTF=${0}.out

  # ---------------------------------------------------------------------------
  # You may want to delete the output file so that backup information does
  # not accumulate.  If not, delete the following lines.
  # ---------------------------------------------------------------------------

  if [ -f "$OUTF" ]
  then
```

```
rm -f "$OUTF"
fi

{ # output block

echo "‘date‘ ----------------Beginning of Script------------"
echo "Script name: $0"

# ------------------------------------------------------------------------
# Replace /db/oracle/product/8.0.5, below, with the Oracle home path.
# ------------------------------------------------------------------------

ORACLE_HOME=/db/oracle/product/8.0.5
export ORACLE_HOME

# ------------------------------------------------------------------------
# Replace ORA805, below, with the Oracle SID of the target database.
# ------------------------------------------------------------------------

ORACLE_SID=ORA805
export ORACLE_SID

# ------------------------------------------------------------------------
# Replace ora805, below, with the Oracle DBA user id (account).
# ------------------------------------------------------------------------

ORACLE_USER=ora805

# ------------------------------------------------------------------------
# Replace ${ORACLE_HOME}/scripts, below, with the NetBackup Oracle script
# path. Since subsequent installs or deinstalls will remove this file,
# you will want to move this script from its installed location before
# making updates.
# ------------------------------------------------------------------------

NB_ORA_SCRIPTS=${ORACLE_HOME}/scripts

# ------------------------------------------------------------------------
# Set the Oracle Recovery Manager name.
# ------------------------------------------------------------------------

RMAN=rman

# ------------------------------------------------------------------------
# Print out the value of the variables set by this script.
# ------------------------------------------------------------------------

echo
echo   "RMAN: $RMAN"
echo   "ORACLE_SID: $ORACLE_SID"
echo   "ORACLE_USER: $ORACLE_USER"
echo   "ORACLE_HOME: $ORACLE_HOME"
echo   "NB_ORA_SCRIPTS: $NB_ORA_SCRIPTS"
```

```
# ------------------------------------------------------------------------
# Print out the value of the variables set by bphdb.
# ------------------------------------------------------------------------

echo
echo   "NB_ORA_FULL: $NB_ORA_FULL"
echo   "NB_ORA_INCR: $NB_ORA_INCR"
echo   "NB_ORA_CINC: $NB_ORA_CINC"
echo   "NB_ORA_SERV: $NB_ORA_SERV"
echo   "NB_ORA_CLASS: $NB_ORA_CLASS"
echo   "NB_ORA_PC_SCHED: $NB_ORA_PC_SCHED"
echo   "NB_ORA_SCHEDULED: $NB_ORA_SCHEDULED"
echo   "NB_ORA_USER_INITIATED: $NB_ORA_USER_INITIATED"

# ------------------------------------------------------------------------
# This script assumes that the database is properly opened. If desired,
# this would be the place to verify that.
# ------------------------------------------------------------------------

echo

if [ "$NB_ORA_INCR" = "1" ]
then
echo "hot database differential incremental backup requested"
CMDFILE=${NB_ORA_SCRIPTS}/hot_database_backup_level1_differential.rcv

elif [ "$NB_ORA_CINC" = "1" ]
then
echo "hot database cumulative incremental backup requested"
CMDFILE=${NB_ORA_SCRIPTS}/hot_database_backup_level1_cumulative.rcv

else # default
echo "hot database backup requested (incremental level 0)"
CMDFILE=${NB_ORA_SCRIPTS}/hot_database_backup_level0.rcv
fi

# ------------------------------------------------------------------------
# Call Recovery Manager to initiate the backup. This example does not use a
# Recovery Catalog. If you choose to use one, remove the option, nocatalog,
# from the rman command line below and add a 'connect rcvcat' statement to
# the corresponding command file ($CMDFILE). An alternative to putting the
# connect statements in the command file would be to add them to the rman
# command line.
# ------------------------------------------------------------------------

echo
echo "% ${ORACLE_HOME}/bin/$RMAN nocatalog cmdfile '$CMDFILE'"
echo

su - $ORACLE_USER -c "
${ORACLE_HOME}/bin/$RMAN nocatalog cmdfile "$CMDFILE"
"
RETURN_STATUS=$?
```

```
echo
echo "‘date‘ ----------------End of Script-----------------"
echo
echo "exit $RETURN_STATUS"
exit $RETURN_STATUS

} >> $OUTF
```

Example 3, `cold_duplex_database_backup_full.sh`

This example sets up the environment and calls `rman` to execute a command file that will make three copies of a cold (consistent) database backup.

```
#!/bin/sh
  # $Header: cold_duplex_database_backup_full.sh,v 1.1 99/04/23 17:13:20 fma Exp
  $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  cold_duplex_database_backup_full.sh
  # ---------------------------------------------------------------------------
  # This script uses Recovery Manager to make multiple copies of a cold
  # (consistent) database backup. A cold backup is one where the database
  # is shutdown cleanly before performing the backup. It is assumed that
  # this script will be executed by user root. In order for RMAN to work
  # properly we switch user (su -) to the oracle dba account. If this script
  # runs under a user account that has Oracle dba privilege you should
  # remove the switch user logic.
  #
  # Note that you must enable (set to TRUE in the init.ora file) the
  # BACKUP_TAPE_IO_SLAVES initialization parameter to perform duplexed
  # backups. RMAN will configure as many slaves as needed for the number
  # of backup copies you request. For more information on
  # BACKUP_TAPE_IO_SLAVES, see the Oracle8 Reference.
  # ---------------------------------------------------------------------------

  # ---------------------------------------------------------------------------
  # Put output in <this file name>.out. Change as desired.
  # Note: output directory requires write permission.
  # ---------------------------------------------------------------------------

  OUTF=${0}.out

  # ---------------------------------------------------------------------------
  # You may want to delete the output file so that backup information does
  # not accumulate.  If not, delete the following lines.
  # ---------------------------------------------------------------------------
```

```
if [ -f "$OUTF" ]
then
rm -f "$OUTF"
fi

{ # output block

echo "‘date‘ ----------------Beginning of Script------------"
echo "Script name: $0"

# ------------------------------------------------------------------------
# Replace /db/oracle/product/8.1.5 and later, below, with the Oracle home path.
# ------------------------------------------------------------------------

ORACLE_HOME=/db/oracle/product/8.1.5 and later
export ORACLE_HOME

# ------------------------------------------------------------------------
# Replace ora815, below, with the Oracle DBA user id (account).
# ------------------------------------------------------------------------

ORACLE_USER=ora815

# ------------------------------------------------------------------------
# Replace productionDB, below, with the target database TNS alias (service)
# name from the tnsnames.ora file.
# ------------------------------------------------------------------------

TARGET_TNS=productionDB
export TARGET_TNS

# ------------------------------------------------------------------------
# Replace rcvcatDB, below, with the recovery catalog database TNS alias
# (service) name from the tnsnames.ora file.
# ------------------------------------------------------------------------

RCVCAT_TNS=rcvcatDB
export RCVCAT_TNS

# ------------------------------------------------------------------------
# Replace ${ORACLE_HOME}/scripts, below, with the NetBackup Oracle script
# path. Since subsequent installs or deinstalls will remove this file,
# you will want to move this script from its installed location before
# making updates.
# ------------------------------------------------------------------------

NB_ORA_SCRIPTS=${ORACLE_HOME}/scripts

# ------------------------------------------------------------------------
# Set the Oracle Recovery Manager name.
# ------------------------------------------------------------------------

RMAN=rman
```

```
# ------------------------------------------------------------------------
# Print out the value of the variables set by this script. Won't be needing
# any of the variables set in bphdb.
# ------------------------------------------------------------------------

echo
echo   "RMAN: $RMAN"
echo   "TARGET_TNS: $TARGET_TNS"
echo   "RCVCAT_TNS: $RCVCAT_TNS"
echo   "ORACLE_USER: $ORACLE_USER"
echo   "ORACLE_HOME: $ORACLE_HOME"
echo   "NB_ORA_SCRIPTS: $NB_ORA_SCRIPTS"

echo
echo "cold duplexed full database backup requested"
# ------------------------------------------------------------------------
# Note: the CMDFILE contains commands that are only valid with Oracle
# release 8.1.5 or greater.
# ------------------------------------------------------------------------

CMDFILE=${NB_ORA_SCRIPTS}/cold_duplex_database_backup_full.rcv

# ------------------------------------------------------------------------
# Call Recovery Manager to initiate the backup. This example uses a
# Recovery Catalog. If you choose not to use one, replace the option
# 'rcvcat <userid>/<passwd>@<tns alias>' with the 'nocatalog' option.
# An alternative to using the 'target' and 'rcvcat' command line options
# is to connect to the target and catalog databases from within the
# $CMDFILE script, using the 'connect' command.
#
# Change user ids and passwords accordingly.
#
# Note: Any environment variables needed at run time by RMAN or $CMDFILE
#       must be set and exported within the switch user (su) command.
# ------------------------------------------------------------------------

echo
echo  "%  ${ORACLE_HOME}/bin/$RMAN  target  internal/oracle@$TARGET_TNS  rcvcat
rman/rman@$RCVCAT_TNS cmdfile $CMDFILE"
echo

su - $ORACLE_USER -c "
${ORACLE_HOME}/bin/$RMAN     target     internal/oracle@$TARGET_TNS     rcvcat
rman/rman@$RCVCAT_TNS cmdfile "$CMDFILE"
"
RETURN_STATUS=$?

echo
echo "'date' ----------------End of Script------------------"
echo
echo "exit $RETURN_STATUS"
exit $RETURN_STATUS

} >> $OUTF
```

Example 4, `hot_database_backup_proxy.sh`

This example sets the environment and calls `rman` with the appropriate command file to perform a whole database proxy backup. It gets executed for both full and incremental backups.

When a schedule executes, NetBackup sets environment variables that can be used by NetBackup Advanced BLI Agent.

The script shows how to use the RMAN `send` command to pass the following variables with vendor-specific quoted string:

NB_ORA_CLASS

Name of the Oracle class to be used for non-proxy and proxy backups.

NB_ORA_PC_SCHED

Name of the Oracle schedule to be used for proxy backup.

NB_ORA_PC_STREAMS

Specifies a number of parallel backup streams to be started for each file system in the proxy backup session.

NB_ORA_PC_FILESYSTEMS

Specifies a number of file systems to be backed up in parallel. The default is to start backups of all file systems in parallel.

Refer to “Multiple Stream Configuration Example” on page 79.

```
#!/bin/sh
  # $Header: hot_database_backup_proxy.sh,v 1.4 2000/02/22 21:42:24 exb Stab $
  #
  #bcpyrght
  #***************************************************************************
  #* $VRTScprght:  Copyright 1993 - 2000 VERITAS Software Corporation, All Rights
  Reserved  $
  ***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  hot_database_backup_proxy.sh
  # ---------------------------------------------------------------------------
  #  This script uses Recovery Manager to take a hot (inconsistent) database
  #  backup. A hot backup is inconsistent because portions of the database are
  #  being modified and written to the disk while the backup is progressing.
  #  You must run your database in ARCHIVELOG mode to make hot backups. It is
  #  assumed that this script will be executed by user root. In order for RMAN
  #  to work properly we switch user (su -) to the oracle dba account before
  #  execution. If this script runs under a user account that has Oracle dba
  #  privilege you should remove the switch user logic.
```

```
# ------------------------------------------------------------------------

# ------------------------------------------------------------------------
# Put output in <this file name>.out. Change as desired.
# Note: output directory requires write permission.
# ------------------------------------------------------------------------

OUTF=${0}.out

# ------------------------------------------------------------------------
# You may want to delete the output file so that backup information does
# not accumulate.  If not, delete the following lines.
# ------------------------------------------------------------------------

if [ -f "$OUTF" ]
then
rm -f "$OUTF"
fi

{ # output block

echo "‘date‘ ----------------Beginning of Script------------"
echo "Script name: $0"

# ------------------------------------------------------------------------
# Replace /db/oracle/product/8.1.5, below, with the Oracle home path.
# ------------------------------------------------------------------------

ORACLE_HOME=/db/oracle/product/8.1.5

# ------------------------------------------------------------------------
# Replace ORA815, below, with the Oracle SID of the target database.
# ------------------------------------------------------------------------

ORACLE_SID=ORA815

# ------------------------------------------------------------------------
# Replace ora815, below, with the Oracle DBA user id (account).
# ------------------------------------------------------------------------

ORACLE_USER=ora815

# ------------------------------------------------------------------------
# Replace ${ORACLE_HOME}/scripts, below, with the NetBackup Oracle script
# path. Since subsequent installs or deinstalls will remove this file,
# you will want to move this script from its installed location before
# making updates.
# ------------------------------------------------------------------------

NB_ORA_SCRIPTS=${ORACLE_HOME}/scripts

# ------------------------------------------------------------------------
# Set the Oracle Recovery Manager name.
# ------------------------------------------------------------------------
```

```
RMAN=rman

# ------------------------------------------------------------------------
# Replace the values of the following variables:
#
#    NB_ORA_PC_FILESYSTEMS - specifies a number of file systems to be backed up
#                        in parallel.
#       NB_ORA_PC_STREAMS - specifies a number of parallel backup streams to
#                        start within each file system.
# ------------------------------------------------------------------------


NB_ORA_PC_FILESYSTEMS=1

NB_ORA_PC_STREAMS=3

# ------------------------------------------------------------------------
# Print out the value of the variables set by this script.
# ------------------------------------------------------------------------

echo
echo   "RMAN: $RMAN"
echo   "ORACLE_SID: $ORACLE_SID"
echo   "ORACLE_USER: $ORACLE_USER"
echo   "ORACLE_HOME: $ORACLE_HOME"
echo   "NB_ORA_SCRIPTS: $NB_ORA_SCRIPTS"

echo   "NB_ORA_PC_STREAMS: $NB_ORA_PC_STREAMS"
echo   "NB_ORA_PC_FILESYSTEMS: $NB_ORA_PC_FILESYSTEMS"

# ------------------------------------------------------------------------
# Print out the value of the variables passed from the server by bphdb.
# ------------------------------------------------------------------------

echo
echo   "NB_ORA_FULL: $NB_ORA_FULL"
echo   "NB_ORA_INCR: $NB_ORA_INCR"
echo   "NB_ORA_CINC: $NB_ORA_CINC"
echo   "NB_ORA_SERV: $NB_ORA_SERV"
echo   "NB_ORA_CLASS: $NB_ORA_CLASS"
echo   "NB_ORA_PC_SCHED: $NB_ORA_PC_SCHED"
echo   "NB_ORA_SCHEDULED: $NB_ORA_SCHEDULED"

# ------------------------------------------------------------------------
# This script assumes that the database is properly opened. If desired,
# this would be the place to verify that.
# ------------------------------------------------------------------------

# ------------------------------------------------------------------------
# Print out the type of the backup that is initiated by the NBU server.
# ------------------------------------------------------------------------
echo
```

```
if [ "$NB_ORA_INCR" = "1" ]
then
echo "hot database differential incremental backup requested"

elif [ "$NB_ORA_CINC" = "1" ]
then
echo "hot database cumulative incremental backup requested"

else # default
echo "hot database full backup requested"
fi

# ------------------------------------------------------------------------
# Specify the Recovery Manager command file for the database proxy backup.
# ------------------------------------------------------------------------

CMDFILE=${NB_ORA_SCRIPTS}/hot_database_backup_proxy.rcv

# ------------------------------------------------------------------------
# Call Recovery Manager to initiate the backup. This example does not use a
# Recovery Catalog. If you chose to use one, remove the option, nocatalog,
# from the rman command line below and add a 'connect rcvcat' statement to
# the corresponding command file ($CMDFILE). An alternative to putting the
# connect statements in the command file would be to add them to the rman
# command line.

# Use the send command on the command line to send NBU class, schedule name
# and the number of the backup streams to be used in the database proxy backup.
# ------------------------------------------------------------------------

echo
echo "executing ${ORACLE_HOME}/bin/$RMAN nocatalog send
\"'NB_ORA_CLASS=$NB_ORA_CLASS, NB_ORA_PC_SCHED=$NB_ORA_PC_SCHED,
NB_ORA_PC_FILESYSTEMS=$NB_ORA_PC_FILESYSTEMS,
NB_ORA_PC_STREAMS=$NB_ORA_PC_STREAMS'\" cmdfile \"$CMDFILE\""
echo

su - $ORACLE_USER -c "
${ORACLE_HOME}/bin/$RMAN nocatalog send \"'NB_ORA_CLASS=$NB_ORA_CLASS,
NB_ORA_PC_SCHED=$NB_ORA_PC_SCHED,
NB_ORA_PC_FILESYSTEMS=$NB_ORA_PC_FILESYSTEMS,
NB_ORA_PC_STREAMS=$NB_ORA_PC_STREAMS'\" cmdfile "$CMDFILE"
"
RETURN_STATUS=$?

echo
echo "`date` ----------------End of Script------------------"
echo
echo "exit $RETURN_STATUS"
exit $RETURN_STATUS

} >> $OUTF
```

RMAN Script

For non-proxy operations, Oracle8 Recovery Manager (RMAN) supports the following different types of backups:

- Full Backup
- Level 0 Backup (Level 0 Incremental)
- Level n Backup (Differential Incremental Backup)
- Level n Backup (Cumulative Incremental Backup)

When generating a datafile backup set, you can make either an incremental backup or a full backup. An incremental backup is a backup of one or more datafiles that contain only those blocks that have been modified since a previous backup. A full backup is a non-incremental backup of one or more datafiles that contain all blocks of the datafiles.

For proxy backups, RMAN supports only full and level 0 backups. NetBackup Advanced BLI Agent performs full and incremental backups using the NetBackup BLI Backup interface.

Example 1, `cold_database_backup_level1_cumulative.rcv`

```
# $Header: cold_database_backup_level1_cumulative.rcv,v 1.4 99/05/07 17:01:52
  fma Stab $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ------------------------------------------------------------------------
  #  cold_database_backup_level1_cumulative.rcv
  # ------------------------------------------------------------------------
  #
  #  Backs up only the blocks which have been modified since the last
  #  level 0 backup was performed. (note, the controlfile is always backed
  #  up in it's entirety i.e. control file backups are never compressed).
  #
  #  Typically, a level 1 backup would be done at least once in between
  #  level 0 backups.
  #
  #  The scenario assumes:
  #     o you want to perform a level 1 cumulative incremental backup
  #     o you are backing your database up to two tape drives
  #     o you want each backup set to include a maximum of 5 files
  #     o you wish to include offline datafiles, and read-only tablespaces,
  #       in the backup
  #     o you want the backup to terminate if any files are not
  #       accessible
```

```
#     o you want to open the database after the backup completes
#     o you are not using a Recovery Catalog
#     o you are using a TNS alias name for the target database
#
#  Note that the format string is constructed to guarantee uniqueness and
#  to enhance NetBackup for Oracle backup and restore performance.
#
#  The connect statements could be part of the rman command line in the
#  Oracle script file that calls this script. They were put here so that
#  the passwords could be protected by setting file permissions to only
#  allow the Oracle dba access.
#
#  Add the following connect statement if using a Recovery Catalog.
#      connect rcvcat '<user>/<passwd>@<TNS alias>'
#
#  Modify the target connect statement according to the following syntax:
#      connect target '<user>/<passwd>@<TNS alias>'
#
#  Script run by:
#      rman nocatalog cmdfile cold_database_backup_level1_cumulative.rcv
#
# ------------------------------------------------------------------------

connect target 'internal/oracle@productionDB'

run {
# Cold database level 1 cumulative incremental backup
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
backup
  incremental level 1
  cumulative
  tag cold_db_bk_level1_cumulative
  filesperset 5
  # recommended format
  format 'bk_%s_%p_%t'
    (database);

# now that the backup is complete, open the db.
sql 'alter database open';
}
```

Example 2, hot_database_backup_level0.rcv

```
# $Header: hot_database_backup_level0.rcv,v 1.4 99/04/23 17:11:39 fma Exp $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  hot_database_backup_level0.rcv
  # ---------------------------------------------------------------------------
  #
  #  Backs up the whole database.  This backup is part of the incremental
  #  strategy (this means it can have incremental backups of levels > 0
  #  applied to it).
  #
  #  We do not need to explicitly request the control file to be included
  #  in this backup, as it is automatically included each time file 1 of
  #  the system tablespace is backed up (the inference: as it is a whole
  #  database backup, file 1 of the system tablespace will be backed up,
  #  hence the controlfile will also be included automatically).
  #
  #  Typically, a level 0 backup would be done at least once a week.
  #
  #  The scenario assumes:
  #     o you want to perform a level 0 backup
  #     o you are backing your database up to two tape drives
  #     o you want each backup set to include a maximum of 5 files
  #     o you wish to include offline datafiles, and read-only tablespaces,
  #       in the backup
  #     o you want the backup to continue if any files are inaccessible.
  #     o you are not using a Recovery Catalog
  #     o you want to archive the current log and then back up all the
  #       archive logs, putting a maximum of 20 logs in a backup set and
  #       deleting them once the backup is complete.
  #     o you are using a TNS alias name for the target database
  #
  #  Note that the format string is constructed to guarantee uniqueness and
  #  to enhance NetBackup for Oracle backup and restore performance.
  #
  #  The connect statements could be part of the rman command line in the
  #  Oracle script file that calls this script. They were put here so that
  #  the passwords could be protected by setting file permissions to only
  #  allow the Oracle dba access.
  #
  #  Add the following connect statement if using a Recovery Catalog.
  #      connect rcvcat '<user>/<passwd>@<TNS alias>'
  #
  #  Modify the target connect statement according to the following syntax:
  #      connect target '<user>/<passwd>@<TNS alias>'
  #
```

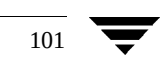
```
#  Script run by:
#      rman nocatalog cmdfile hot_database_backup_level0.rcv
#
# ------------------------------------------------------------------------

connect target 'internal/oracle@productionDB'

run {
# Hot database level 0 whole backup
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
backup
  incremental level 0
  skip inaccessible
  tag hot_db_bk_level0
  filesperset 5
  # recommended format
  format 'bk_%s_%p_%t'
    (database);
  sql 'alter system archive log current';
  # backup all archive logs
  backup
   filesperset 20
   format 'al_%s_%p_%t'
   (archivelog all
    delete input);
}
```

Example 3, `hot_database_backup_level1_differential.rcv`

```
# $Header: hot_database_backup_level1_differential.rcv,v 1.4 99/04/26 14:58:08
fma Stab $
#
#bcpyrght
#***************************************************************************
#* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
#***************************************************************************
#ecpyrght
#
# ------------------------------------------------------------------------
#  hot_database_backup_level1_differential.rcv
# ------------------------------------------------------------------------
#
#  Backs up only the blocks which have been modified since the last
#  level 0 backup was performed. (note, the controlfile is always backed
#  up in it's entirety i.e. control file backups are never compressed).
#
#  Typically, a level 1 backup would be done at least once in between
#  level 0 backups.
#
#  The scenario assumes:
#     o you want to perform a differential level 1 backup
#     o you are backing your database up to two tape drives
```

```
#     o you want each backup set to include a maximum of 5 files
#     o you already have valid backups of read-only tablespaces, and
#       offline datafiles so we skip backing up these files
#     o you want the backup to continue if any files are inaccessible.
#     o you are not using a Recovery Catalog
#     o you want to archive the current log and then back up all the
#       archive logs, putting a maximum of 20 logs in a backup set and
#       deleting them once the backup is complete.
#     o you are using a TNS alias name for the target database
#
#  Note that the format string is constructed to guarantee uniqueness and
#  to enhance NetBackup for Oracle backup and restore performance.
#
#  The connect statements could be part of the rman command line in the
#  Oracle script file that calls this script. They were put here so that
#  the passwords could be protected by setting file permissions to only
#  allow the Oracle dba access.
#
#  Add the following connect statement if using a Recovery Catalog.
#      connect rcvcat '<user>/<passwd>@<TNS alias>'
#
#  Modify the target connect statement according to the following syntax:
#      connect target '<user>/<passwd>@<TNS alias>'
#
#  Script run by:
#      rman nocatalog cmdfile hot_database_backup_level1_differential.rcv
#
# ------------------------------------------------------------------------

connect target 'internal/oracle@productionDB'

run {
# Hot database level 1 differential incremental backup
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
backup
  incremental level 1
  skip offline
  skip readonly
  skip inaccessible
  tag hot_db_bk_level1_cum
  filesperset 5
  # recommended format
  format 'bk_%s_%p_%t'
    (database);
  sql 'alter system archive log current';
  # backup all archive logs
  backup
   filesperset 20
   format 'al_%s_%p_%t'
   (archivelog all
    delete input);
}
```

Example 4, `hot_tablespace_backup_level1_differential.rcv`

```
# $Header: hot_tablespace_backup_level1_differential.rcv,v 1.2 99/04/20 14:01:41
  fma Stab $
  #
  #bcpyrght
  #***************************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #***************************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------------
  #  hot_tablespace_backup_level1_differential.rcv
  # ---------------------------------------------------------------------------
  #
  #  This script backs up the system tablespace to tape and archives the
  #  current log. This backup is part of the incremental strategy, it
  #  backs up only the blocks which have been modified since the last
  #  level 0 backup was performed. (note, the controlfile is always backed
  #  up in it's entirety i.e. control file backups are never compressed).
  #
  #  Typically, a level 1 backup would be done at least once in between
  #  level 0 backups.
  #
  #  Oracle recommends that the current log be archived immediately after
  #  performing any open (i.e. hot) backup to ensure that the backup is
  #  immediately usable, even if the online log is lost.
  #
  #  We do not need to explicitly request the control file to be included
  #  in this backup, as it is automatically included each time file 1 of
  #  the system tablespace is backed up.
  #
  #  The scenario assumes:
  #     o you want to perform a level 1 differential incremental backup
  #     o you are backing your tablespace up to one tape drive
  #     o you wish to include offline datafiles
  #     o you want the backup to continue if any files are inaccessible.
  #     o you are using a Recovery Catalog
  #     o you want to archive the current log
  #     o you are using TNS alias names for target and catalog databases
  #
  #  Note that the format string is constructed to guarantee uniqueness and
  #  to enhance NetBackup for Oracle backup and restore performance.
  #
  #  The connect statements could be part of the rman command line in the
  #  Oracle script file that calls this script. They were put here so that
  #  the passwords could be protected by setting file permissions to only
  #  allow the Oracle dba access.
  #
  #  Modify the rcvcat connect statement according to the following syntax:
  #      connect rcvcat '<user>/<passwd>@<TNS alias>'
  #
```

```
#  Modify the target connect statement according to the following syntax:
#      connect target '<user>/<passwd>@<TNS alias>'
#
#  Script run by:
#      rman cmdfile hot_tablespace_backup_level1_differential.rcv
#
# ------------------------------------------------------------------------

connect target 'internal/oracle@productionDB'
connect rcvcat 'rman/rman@rcvcatDB'

run {
allocate channel t1 type 'SBT_TAPE';
backup
  incremental level 1
  skip inaccessible
  tag hot_tbs_bk_level1_dif
  # recommended format
  format 'df_%s_%p_%t'
    (tablespace system);
sql 'alter system archive log current';
}
```

Example 5, `database_restore.rcv`

```
# $Header: database_restore.rcv,v 1.2 99/04/20 14:01:03 fma Stab $
#
#bcpyrght
#***************************************************************************
#* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
#***************************************************************************
#ecpyrght
#
# ------------------------------------------------------------------------
#  database_restore.rcv
# ------------------------------------------------------------------------
#
#  The script below restores all datafiles, and recovers them completely.
#  All data files are restored to their current locations, from their most
#  recent backups.  It does not restore the control file. If you wish to
#  omit restoring one or more tablespaces, use the skip tablespace clause
#  on the restore command.
#
#  Note recovery manager automatically:
#    o determines whether the controlfile is a backup, and will perform the
#      correct type of recovery.
#    o restores any archived redo logs that are currently not on disk,
#      which are required for recovery.
#
#  If there are incremental backups, it will apply these first, then
#  apply any redo required to fully recover.
#
#  The connect statements could be part of the rman command line in the
```

```
#  Oracle script file that calls this script. They were put here so that
#  the passwords could be protected by setting file permissions to only
#  allow the Oracle dba access.
#
#  Add the following connect statement if using a Recovery Catalog.
#      connect rcvcat '<user>/<passwd>@<TNS alias>'
#
#  Modify the target connect statement according to the following syntax:
#      connect target '<user>/<passwd>@<TNS alias>'
#
#  Script run by:
#      rman nocatalog cmdfile database_restore.rcv
#
# ------------------------------------------------------------------------

connect target 'internal/oracle@productionDB'

run {
# Restore all datafiles, and recovers them completely.
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
  restore
    (database);
  recover
    database;
}
```

Example 6, `pit_database_restore.rcv`

```
# $Header: pit_database_restore.rcv,v 1.2 99/04/20 14:01:46 fma Stab $
#
#bcpyrght
#***************************************************************************
#* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
#***************************************************************************
#ecpyrght
#
# ------------------------------------------------------------------------
#  pit_database_restore.rcv
# ------------------------------------------------------------------------
#  The following scenario assumes:
#    o you wish to do an incomplete recovery due to an
#      application error which was made at a specific time
#    o the database is in the mount state
#    o there are 2 tape drives
#    o the NLS_LANG and NLS_DATE_FORMAT environment variables are set
#    o you are using a recovery catalog
#
#  Note: It is highly advisable to backup the database immediately after
#        opening the database resetlogs.
#
#  The following script restores and recovers the database to the time
#  immediately before the user error occurred.
```

```
#
#    o restores the database files (to the original locations)
#    o recovers the datafiles by either using a combination
#      of incremental backups and redo, or just redo.
#    o Recovery Manager will complete the recovery when it
#      reaches the time specified.
#    o Opens the database resetlogs.
#    o Oracle recommends you backup your database after the
#      resetlogs (this is not shown in the example).
#
#
#  Ensure you set your NLS_LANG and NLS_DATE_FORMAT environment variables.
#  You can set these to whatever you wish - the example below keeps the date
#  format to the standard date format used for recovery.
#
#  Ensure that the target database is in the mount state. To mount the
#  database you can start up Server Manager, and:
#
#  SVRMGR> connect internal
#  Connected.
#  SVRMGR> startup mount restrict
#
#  Script run by:
#      rman target <userid>/<passwd>@<tns alias> \
#           rcvcat <userid>/<passwd>@<tns alias> \
#           cmdfile pit_database_restore.rcv
#  Or if not using a recovery catalog:
#      rman target <userid>/<passwd>@<tns alias> nocatalog \
#           cmdfile pit_database_restore.rcv
# ------------------------------------------------------------------------

run {

#  The 'set until time' command is in effect for all commands executed
#  between the { and } braces.  This means both the restore and recover
#  will both be relative to that point-in-time.
#  Note that Recovery Manager is able to query the Recovery Catalog,
#  determine what the structure of the database was at that time, and
#  restore it.


set until time '1997-05-09:14:30:00';

allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';

restore
  (database);

#  There is no need to manually catalog any archivelogs before the recovery,
#  as Recovery Manager does an implicit catalog resync from the current
#  control file.

    recover
```

```
      database;

sql 'alter database open resetlogs';
}
```

Example 7, `cold_duplex_database_backup_full.rcv`

```
# $Header: cold_duplex_database_backup_full.rcv,v 1.1 99/04/23 17:09:03 fma Exp $
#
#bcpyrght
#***************************************************************************
#* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
#***************************************************************************
#ecpyrght
#
# ---------------------------------------------------------------------------
#  cold_duplex_database_backup_full.rcv
# ---------------------------------------------------------------------------
#
#  NOTE: THIS SCRIPT CONTAINS ORACLE 8.1.5 and later COMMANDS THAT
#        WILL NOT WORK WITH ORACLE 8.0.X
#
#  This script performs a full database backup, creating two identical
#  copies. A full backup is a non-incremental backup that contain all blocks
#  of the datafiles. Since it is a non-incremental backup it can not have
#  incremental backups applied to it.
#
#  We do not need to explicitly request the control file to be included
#  in this backup, as it is automatically included each time file 1 of
#  the system tablespace is backed up (the inference: as it is a full
#  database backup, file 1 of the system tablespace will be backed up,
#  hence the controlfile will also be included automatically).
#
#  The scenario assumes:
#     o you are using Oracle RMAN version 8.1.5 and later.
#     o you want to perform a non-incremental consistent (cold) full backup
#     o you want to use shutdown and startup commands to guarantee
#       that the target database is in a consistent state
#     o you want to use the set duplex command to create two (2)
#       copies of the backup set
#     o you have enabled (set to TRUE) the BACKUP_TAPE_IO_SLAVES
#       initialization parameter in the init.ora file.
#     o you are backing up your database to two tape drives (one for
#       each copy, as identified by a backup policy schedule)
#     o you want each backup set to include a maximum of 5 files
#     o you wish to include offline datafiles, and read-only tablespaces,
#       in the backup
#     o you want the backup to terminate if any files are not
#       accessible
#     o you have already set the ORACLE_SID environment variable to the
#       target database for use by the startup command (not necessary if
#       the pfile= option is used with startup)
#     o you want to open the database after the backup completes
```

```
#
#  Note that the format string is constructed to guarantee uniqueness and
#  to enhance NetBackup for Oracle backup and restore performance.
#
#  Script run by:
#      rman target <userid>/<passwd>@<tns alias> \
#           rcvcat <userid>/<passwd>@<tns alias> \
#           cmdfile cold_duplex_database_backup.rcv
#
#  or if not using a recovery catalog:
#
#      rman target <userid>/<passwd>@<tns alias> nocatalog \
#           cmdfile cold_duplex_database_backup.rcv
#
# The shutdown-startup logic of this script can be commented out if you know
# that the database will always be mounted and in a consistent state before
# this script is executed.
# ------------------------------------------------------------------------

# Shutdown the target database in immediate priority. This lets current
# calls to the database complete, but prevents further logons or calls.

shutdown immediate;

# Startup the database in case it had crashed or was not shutdown cleanly
# prior to starting this script. This will perform a crash recover if
# it is needed. Use the DBA option because we are going to shutdown
# again right away and we don't want to let users in during the short
# interval. The default init<SID>.ora file will be used by the startup
# command (requires the environment variable, ORACLE_SID, to be set for
# the target database). If ORACLE_SID is not properly set, or the init file
# was moved or renamed, use the 'pfile=' option on the startup command.

startup dba open;
shutdown immediate;

# Now we know that the database is cleanly closed and is ready for a
# cold backup. RMAN requires that the database be started and mounted
# to perform a backup.

startup mount;

run {
# two identical copies of a cold full database backup
set duplex=ON;
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
# two separate backup policy schedules specify distinct tape drives
# change to your CLASS and BACKUP POLICY SCHEDULE
send 'CPF1_BK_CLASS=ColdDbFullBk_duplex, CPF1_BK_POLICY=user_cp1';
send 'CPF2_BK_CLASS=ColdDbFullBk_duplex, CPF2_BK_POLICY=user_cp2';
backup
  full
  tag cold_db_bk_full
```

```
    filesperset 5
    # Recommended format for uniqueness and performance.
    format 'bk_%U_%t'
      (database);

  # now that the backup is complete, open the db.
  alter database open;
  }
```

As shown in Example 7, cold_duplex_database_backup_full.rcv, Oracle8i RMAN provides an API that allows you to make up to four backup sets simultaneously, each an exact duplicate of the others. Using NetBackup for Oracle on UNIX, you can back up each copy to a different tape to protect against disaster, media damage, or human error. You must use the `set duplex` and the `send` commands to take advantage of this feature.

The `set duplex` command specifies the number of copies of each backup piece that the channels should create. The `set duplex` command affects all channels allocated after issuing the command and is in effect until explicitly disabled or changed during the session. You cannot issue the `set duplex` command after allocating a channel.

The command syntax is:

```
set duplex = {ON | OFF | 1 | 2 | 3 | 4}
```

By default, duplex is `OFF` (a single backup set is produced). If you specify `ON`, two identical backup sets are produced.

Each output file must have a unique name. You should use the `%U` format specifier to satisfy this restriction. `%U` is equivalent to `%u_%p_%c` and it guarantees the uniqueness of the backup set name in all circumstances.

Note that you must enable the BACKUP_TAPE_IO_SLAVES initialization parameter to perform duplexed backups. RMAN will configure as many media as needed for the number of backup copies you request. For more information on BACKUP_TAPE_IO_SLAVES, see the Oracle8 Reference. For improved NetBackup for Oracle on UNIX performance, also use the `%t` format specifier at the end of the format string. For example: `%U_%t`.

Use the `send` command to specify the class and / or schedule to use with each backup. Because NetBackup for Oracle on UNIX uses the class or schedule to determine what media to use, this information is required for each copy, or an error will occur.

The command syntax is:

```
send 'keyword=value [, keyword=value,...]';
```

The keywords to specify class are: CPF1_BK_CLASS, CPF2_BK_CLASS, CPF3_BK_CLASS, and CPF4_BK_CLASS, which specify the backup class for duplexed file 1 through duplexed file 4.

The keywords used to specify schedule are: CPF1_BK_POLICY, CPF2_BK_POLICY, CPF3_BK_POLICY, and CPF4_BK_POLICY, which specify the Backup Policy schedule for duplexed file 1 through duplexed file 4.

Example 8, `hot_database_backup_proxy.rcv`

In this example, RMAN will allocate two channels.

One channel will be used to proxy backup all database datafiles. The number of backup jobs started simultaneously will be determined by two variables specified in the calling script `hot_database_backup_proxy.sh`. These variables are NB_ORA_PC_FILESYSTEMS and NB_ORA_PC_STREAMS.

Another channel will be used to perform a non-proxy backup of the control file.

After that, both channels will be used to perform the non-proxy backups of archived logs with 20 files per backup set.

Automatic backup schedules execute the NetBackup for Oracle script `hot_database_backup_proxy.sh`, defined in the file list, and perform proxy BLI Backup.

Backup Policy schedule performs a non-proxy backup of control file and archived logs.

```
# $Header: hot_database_backup_proxy.rcv,v 1.1 99/07/20 14:21:02 exb Exp $
  #
  #bcpyrght
  #*********************************************************************
  #* Copyright 1993 - 1999 VERITAS Software Corporation, All Rights Reserved *
  #*********************************************************************
  #ecpyrght
  #
  # ---------------------------------------------------------------------
  #  hot_database_backup_proxy.rcv
  # ---------------------------------------------------------------------
  #
  #  Backs up the whole database using proxy copy feature.
  #
  #  For proxy backups the number of parallel jobs is conttrolled by the
  #  NB_ORA_STREAMS parameter specified in the calling shell script.
  #
  #  You do not need to explicitly request the control file to be included
  #  in this backup, as it is automatically included each time file 1 of
  #  the system tablespace is backed up (the inference: as it is a whole
  #  database backup, file 1 of the system tablespace will be backed up,
  #  hence the controlfile will also be included automatically).
  #
  #  Note that in 8.1.5 controlfile and archivelogs are not backed up
  #  using proxy copy method. Rman will initiate non-proxy copy sessions
  #  to backup the controlfile and archivelogs.
```

```
#
#  The scenario assumes:
#     o you want to backup datafiles using proxy copy method.
#     o you wish to include offline datafiles, and read-only
#       tablespaces, in the backup
#     o you want the backup to continue if any files are inaccessible.
#     o you are not using a Recovery Catalog
#     o you want to archive the current log, back up all the
#       archive logs using two channels and putting a maximum of 20 logs in a
#       backup set, and deleting them once the backup is complete.
#     o you are using a TNS alias name for the target database
#
#  Note that the format string is constructed to guarantee uniqueness and
#  to enhance NetBackup for Oracle backup and restore performance.
#
#  The connect statements could be part of the rman command line in the
#  Oracle script file that calls this script. They were put here so that
#  the passwords could be protected by setting file permissions to only
#  allow the Oracle dba access.
#
#  Add the following connect statement if using a Recovery Catalog.
#      connect rcvcat '<user>/<passwd>@<TNS alias>'
#
#  Modify the target connect statement according to the following syntax:
#      connect target '<user>/<passwd>@<TNS alias>'
#
#  Script run by:
#      rman nocatalog cmdfile hot_database_backup_proxy.rcv
#
# ---------------------------------------------------------------------

connect target 'internal/oracle@productionDB'

run {
allocate channel t1 type 'SBT_TAPE';
allocate channel t2 type 'SBT_TAPE';
# database proxy backup
backup
  proxy
  skip inaccessible
  tag hot_db_bk_proxy
  # recommended format
  format 'bk_%s_%p_%t'
    (database);
  sql 'alter system archive log current';
  # archive logs non-proxy backup
  backup
   filesperset 20
   format 'al_%s_%p_%t'
   (archivelog all
    delete input);
release channel t1;
release channel t2;
}
```

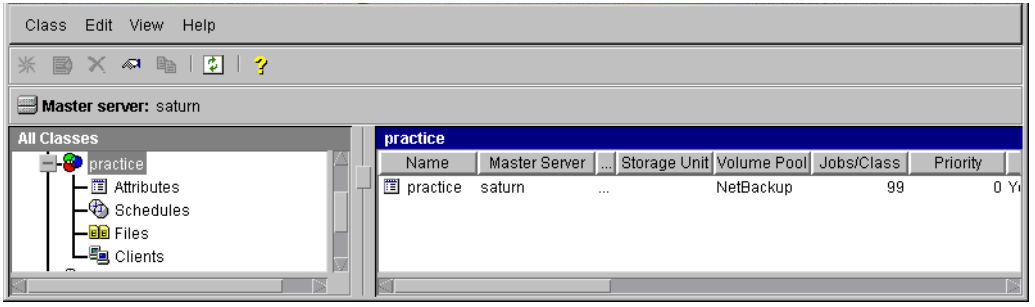
Test NetBackup for Oracle on UNIX Configuration Settings

After you have configured the master server for NetBackup for Oracle on UNIX, you should test the configuration settings. For a description of status codes, refer to the *NetBackup Troubleshooting Guide - Windows NT Server* if you are using a Windows NT/2000 server or the *NetBackup Troubleshooting Guide - UNIX* if you are using a UNIX server.

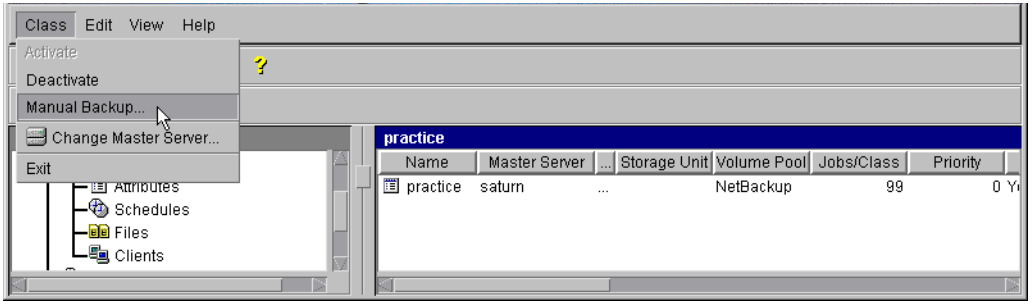
NetBackup Administration - Java Interface

Use this procedure to test a class configuration on the NetBackup Administration - Java Interface for HP or Solaris operating systems.

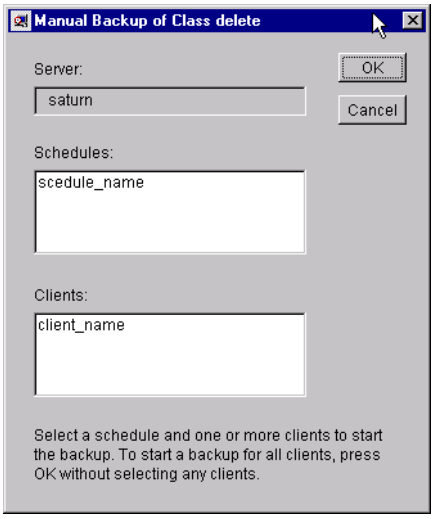
1. Log onto the server as root.
2. Start the NetBackup administrator interface.
3. Click the Backup Policy Management icon. The Backup Policy Management (Classes) - NetBackup dialog appears.
4. Select a class to back up.



5. On the Class menu, click Manual Backup.



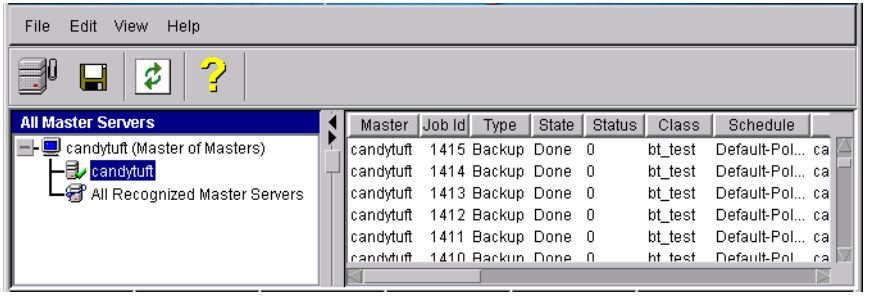
The Manual Backup dialog box appears.



The Schedule pane contains the name of a schedule configured for the class you are going to test. The Client pane contains the name of the client(s) listed in the class you are going to test.

6. Follow the instructions on the dialog box.

7. Click Activity Monitor on the NetBackup Administration interface to open the Activity Monitor dialog box.



If the test does not exit with a successful status, refer to the Troubleshooting chapter.

xbpadm Interface

Use this procedure to test a class configuration on a UNIX NetBackup master server.

1. Log onto the server as root.
2. Start the NetBackup xbpadm administrator interface.
   - If the DISPLAY variable is set, type:

     /usr/openv/netbackup/bin/goodies/xbpadm &
   - If the DISPLAY variable is not set, use the -d option:

/usr/openv/netbackup/bin/goodies/xbpadm -d (*your_machine_name*):0 &

   The NetBackup Administration dialog box will open.
3. Under Classes, select the Oracle class you configured.
4. Under Actions, select Manual Backup. The Manual Backup dialog box will appear.
   a. Select a schedule in the Schedules pane.
   b. Select a client in the Clients pane.
   c. Click OK. The Manual Backup dialog box will close.
5. Under File, select Job Monitor. The Job Monitor dialog box will appear. A status code will display in the Status column.

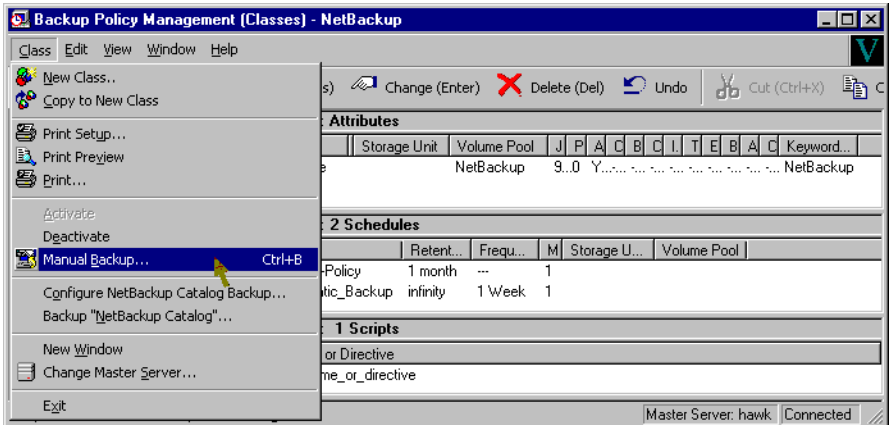
Note The jobs listed in the Job Monitor dialog box include one job for the overall database backup and multiple default-policy jobs which depended on the need of drives have been specified.

If the test does not exit with a successful status, refer to the Troubleshooting chapter.

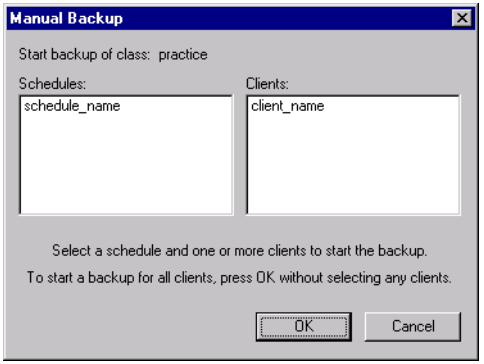
NetBackup Administration - Windows NT/2000 Interface

Use this procedure to test a class configuration from a Windows NT / 2000 server or from the NetBackup Administration Client host.

1. Log onto the server as Administrator.
2. Start the NetBackup administrator interface.
3. Click the Backup Policy Management icon. The Backup Policy Management (Classes) - NetBackup dialog appears.
4. Select a class to back up.
5. On the Class menu, click Manual Backup.



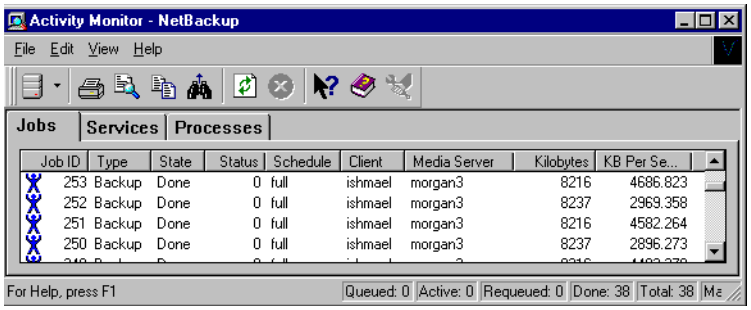
The Manual Backup dialog box appears.



The Schedule pane contains the name of a schedule configured for the class you are going to test. The Client pane contains the name of the client(s) listed in the class you are going to test.

6. Follow the instructions on the dialog box.

7. Click Activity Monitor on the NetBackup Administration interface to open the Activity Monitor dialog box.



If the test does not exit with a successful status, refer to the Troubleshooting chapter.

Using NetBackup for Oracle on UNIX 4

When installation and configuration is complete, you can use the NetBackup interfaces, the Oracle Enterprise Manager, or the command line interface to start Oracle backups and restores, and to maintain the recovery catalog.

This chapter contains the following sections:

- Maintaining a Recovery Catalog
- Querying the Recovery Catalog
- Performing a Backup
- Browsing Backups
- Performing a Restore
- Performing Other RMAN Actions

Maintaining a Recovery Catalog

Oracle Enterprise Manager (RMAN) provides recovery catalog maintenance commands to allow the following functions. Refer to your *Oracle Server Backup and Recovery Guide* for more details on recovery catalog maintenance.

Function	Description
Registering a database with the recovery catalog	Before using RMAN, you must register the target database in the recovery catalog. To do this, start and mount the target database but do not open it. At the RMAN prompt, issue a `register database` command.
Resetting the information in the recovery catalog	The `reset database` command directs RMAN to create a new database incarnation record in the recovery catalog.

Cross-checking the information in the recovery catalog (Oracle 8.1.x only)	Because the Media Manager can mark tapes as expired, the recovery catalog can contain outdated information about archived material. To ensure that data in the recovery catalog is in sync with data in the media management catalog, perform a cross-check. Use the cross-check feature to:
	◆ Determine whether a backup set is available or expired.
	◆ Delete any expired backup sets found in the recovery catalog.
	◆ Call the media manager about the status of a backup piece and then mark it as available or expired.
	The `crosscheck backupset` command will operate on available and expired backup pieces found in the recovery catalog. RMAN will update their status with the result (available or expired).
	To crosscheck a database, start RMAN and connect to the target and recovery catalog databases. At the `rman` command prompt, issue the following commands:
	`allocate channel for maintenance type 'sbt_tape';`
	`crosscheck backupset of database;`

Function	Description

	The `delete expired backupset` command will operate only on expired backup pieces found in the recovery catalog. RMAN will remove them from the recovery catalog.
	
	To delete expired backupsets of a database from the recovery catalog, start RMAN and connect to the target and recovery catalog databases. At the RMAN command prompt, issue the following commands:
	
	`allocate channel for maintenance type 'sbt_tape';`
	
	`delete expired backupset of database;`
	
	The crosscheck and delete backupset commands allow you to restrict the list of objects operated on to the specified device type (i.e., disk or tape), object type (i.e., archived logs or database files), and date range.
	
	The `change ... crosscheck` command will query the media manager to determine if a backup piece is available. If a backup piece is unavailable, RMAN will mark the backup piece as expired. If it was expired but is now available, RMAN will mark the backup piece as available. The command syntax is:
	
	`change backuppiece {`*primary-key-list*`|`*filename-list*`|`*tag*`} crosscheck;`
	
	`change backupset {`*primary-key-list*`} crosscheck;`

Resynchronizing the recovery catalog	RMAN compares the recovery catalog to either the current control file of the target database or a backup control file and updates it with information that is missing or changed. If you are running in ARCHIVELOG mode, resynchronize the recovery catalog regularly since the recovery catalog is not updated automatically when a log switch occurs or when a redo log is archived. You must also resynchronize the recovery catalog after making any change to the physical structure of the target database. As with log archive operations, the recovery catalog is not automatically updated when a physical schema change is made. The RMAN `backup`, `copy`, `restore`, and `switch` commands update the recovery catalog automatically when the target database control file is available and the recovery catalog database is available when any of these commands are executed. If the recovery catalog is unavailable when you issue `backup` or `copy` commands, you should resynchronize it manually.

Function	Description
	To resynchronized the recovery catalog, start RMAN and issue the `resync catalog` command.
Changing the availability of a backup set or file copy	You may periodically need to notify RMAN that the status of a backup set, backup piece, datafile copy, or archived redo log has changed. The RMAN `change` command enables you to make a variety of useful record changes.
	The `change ... uncatalog` command removes references to a backup piece, datafile copy, or archivelog from the recovery catalog. This command works only with a recovery catalog.
	The `change ... delete` command removes references to a backup piece, datafile copy, or archivelog from the control file and recovery catalog. It physically deletes the file. This command works with or without a recovery catalog.
	The `change ... crosscheck` command removes references to a backup piece, datafile copy, or archivelog from the control and recovery catalog when that file no longer exists. This command works with or without a recovery catalog.
	The `change ... unavailable` command marks a backup piece, datafile copy, or archivelog as unavailable. This command works only with a recovery catalog.
Cataloging O/S backups	Cataloging O/S backups will make RMAN aware of the existence of file copies that are created through means other than RMAN.
Recovering a lost or damaged recovery catalog	If the recovery catalog is lost or damaged and recovery of the recovery catalog database through normal Oracle recovery mechanisms is not possible, catalog maintenance commands can be used to partially re-create its contents.

Querying the Recovery Catalog

RMAN allows you to generate a number of reports relevant for backup and recovery using the `report` and `list` commands. The `list` command lists the contents of the recovery catalog, while the `report` command performs a more detailed analyses.

You can use the `report` command to answer questions, such as:

- Which files need a backup?
- Which files have not had a backup in a while?
- Which files are not recoverable due to unrecoverable operations?
- Which backup files can be deleted?
- What was the physical schema of the database at some previous point in time?

The `list` command queries the recovery catalog and produces a listing of its contents. You can list:

- Backup sets containing a backup of a specified list of datafiles.
- Copies of a specified list of datafiles.
- Backup sets containing a backup of any datafile that is a member of a specified list of tablespaces.
- Copies of any datafile that is a member of a specified list of tablespaces.
- All backup sets or copies of all datafiles in the database.
- Backup sets containing a backup of any archivelogs with a specified name and / or within a specified range.
- Copies of any archivelog with a specified name and / or within a specified range.
- Incarnations of a specified database or of all databases known to the recovery catalog.

Refer to your *Oracle Server Backup and Recovery Guide* for more details on querying the recovery catalog.

Performing a Backup

Backups can be performed by using:

- Automatic Backup of an Oracle Class
- Manual Backup of an Oracle Class
- User Directed Backup From the Client
- Executing the rman Command on the Client

Automatic Backup of an Oracle Class

The most convenient way to back up your database is to set up schedules for automatic backups. When the NetBackup scheduler invokes a schedule for an automatic backup, the NetBackup for Oracle scripts run:

- In the same order as they appear in the file list
- On all clients that have them (that is, matching pathnames)

The NetBackup for Oracle scripts will start the database backup by calling rman with the command file.

When the backup is started through NetBackup, NetBackup for Oracle on UNIX leaves script error checking for RMAN. The `rman` command generates an error if it considers a command invalid, but allows any of the commands it normally considers valid to proceed. This means that by specifying the wrong script file name you could start an unintended operation.

Manual Backup of an Oracle Class

The administrator can use the NetBackup server software to manually execute an automatic backup schedule for the Oracle class. See the *NetBackup System Administrator's Guide - UNIX* or the *NetBackup System Administrator's Guide - Windows NT/2000* for detailed instructions.

Refer to "Test NetBackup for Oracle on UNIX Configuration Settings" on page 112 for instructions on initiating a manual backup of an Oracle class.

User Directed Backup From the Client

This section describes the following procedures for performing user-directed backups:

- Using xbp to Perform Backups
- Executing the NetBackup for Oracle Script on the Client
- Executing the rman Command on the Client

Using xbp to Perform Backups

The following describes how to use `xbp` to backup your database. Refer to the *NetBackup User's Guide - UNIX* for detailed instructions on using `xbp` to backup the database.

1. Log in as the Oracle administrator or as root.

   If a different user account is used, change the `su -` command to the Oracle administrator.

2. Execute `xbp` on the client to which you want to backup a database.

   install_path`/netbackup/bin/xbp`

3. In the Directory to Search Box, type the path name of the location of the NetBackup for Oracle scripts. For example:

   install_path`/netbackup/ext/db_ext/oracle/scripts/`

4. From the File menu, click Browse File System for Backup Scripts. The xbp dialog box appears.

5. Select the backup script from the Files pane.

6. On the Backup menu, click Backup Database Using Selected Scripts. The xbp_confirm dialog box appears.

7. Click OK.

   A NetBackup process called `bphdb` starts the NetBackup for Oracle script on the client.

8. View the status of the script execution.

   a. On the Backup menu, click Report Progress Of Backup.... The xbp_progress dialog box will appear.

   b. Select the log file for your backup.

The Contents of Selected Log File pane displays only the status of the script execution. A status =0 message indicates that the script was successfully completed. Go to Step 9 for a detailed status report. For a status other than 0, refer to the Troubleshooting section of this manual.

9. View the log file for the NetBackup operation.

a. Change directories to the bphdb log directory.

```
cd /usr/openv/netbackup/logs/bphdb
```

b. Open the log file with the tail option.

```
tail -f log.mmddyy
```

A Backup completed SUCCESSFULLY message indicates a successfully completed NetBackup operation.

Executing the NetBackup for Oracle Script on the Client

If you know the pathname of the NetBackup for Oracle script that initiates the backup or restore, you can execute the NetBackup for Oracle script from the UNIX command line.

For example, to perform a database backup, at the UNIX command line you might enter:

```
/oracle8/scripts/cold_database_backup.sh
```

The UNIX shell starts the database backup by executing the Oracle script file. The Oracle script file contains commands to execute rman.

Executing the rman Command on the Client

As an oracle user you can also execute the rman command from the UNIX command line with the RMAN script file as a parameter.

The following describes how to set the master server to windows and the Oracle Class to obk before starting the backup.

Since we are connecting to a database using a TNS alias, we must use the RMAN send command or a parms operand to specify the environment variables. At the command prompt enter:

```
rman target ‘internal/oracle@ORCL’ rcvcat ‘rman/rman@RCAT’
send "‘NB_ORA_CLASS=obk,NB_ORA_SERV=windows’" cmdfile
‘/oracle8/scripts/cold_database_backup_level0.rcv’
```

Refer to “RMAN send Command” on page 79 and “RMAN parms Operand” on page 80 for details.

Browsing Backups

This section describes the following procedures for browsing backup images:

- Using xbp to Browse Backups
- Using the Recovery Catalog to Browse Backups
- Using bplist to Browse Backups

Using xbp to Browse Backups

The following describes how to use xbp to browse your database backups. Refer to the *NetBackup User's Guide - UNIX* for detailed instructions on using xbp to browse database backups.

1. Log in as the Oracle administrator or as root.

   If a different user account is used, change the su command to the Oracle administrator.

2. Execute xbp on the client.

   install_path/netbackup/bin/xbp

   The xbp dialog box appears.

3. From the File menu, click Configuration....

   The xbp_config dialog box appears.

   a. In the Class type of image to browse, specify the Oracle class.

   b. Click OK to close the dialog box.

   c. In the Directory To Search Box, enter / as the directory to search.

   d. Set the Range of Dates to include the date of your backup.

   e. Enter 9 in the Directory Depth Box.

   f. From the File menu, select Browse Backups (Restore). A list of backup images will appear in the dialog box.

Using the Recovery Catalog to Browse Backups

If you use a Recovery Catalog, you can use the RMAN report and list commands to browse Oracle backups.

See “Maintaining a Recovery Catalog” on page 118 for more details.

Using `bplist` to Browse Backups

You can use the `bplist` command to browse Oracle backups. The result is the list of backup file names. The following example uses `bplist` to search all Oracle backups for a client named jupiter.

```
install_path/netbackup/bplist -C jupiter -t 4 -R /
/exb_n2bm5bco_1_1392342936
/exb_mabm02ko_1_1392170136
/exb_lqbltds6_1_1392083334
/block2/home/ora8/oradata/cabbage/drsys01.dbf
/block12/oradata/block12tbs2.dat
/block1/oradata/.block1tbsdf1new.dat
/block1/oradata/block1tbsdf1new.dat
```

The `-t 4` on this command specifies the Oracle backups. The `-R` specifies the default number of directory levels to search, 999. Refer to `bplist(1M)` man page for more information on this command.

The first three lines of output are non-proxy backup image names. The last four lines of output are proxy backup image names.

Performing a Restore

Make sure a backup has been successfully completed before attempting a restore. If a backup history does not exist, an error will occur during the execution of a restore.

This section describes the following procedures for performing user-directed restores:

- Using xbp to Restore Backups
- Executing an Oracle Script From a Command Line
- `rman` Command
- Alternate Client Restores

Using xbp to Restore Backups

The following describes how to use `xbp` to restore your database. Refer to the *NetBackup User's Guide - UNIX* for detailed instructions on using `xbp` to restore database backups.

1. Log in as the Oracle administrator or as root.

   If a different user account is used, change the `su -` command to the Oracle administrator.

2. Execute `xbp` on the client to which you want to restore a database.

   install_path`/netbackup/bin/xbp`

Note You cannot restore a database to a remote machine.

3. In the Directory to Search Box, type in the path name of the location of the NetBackup for Oracle scripts. For example:

   install_path`/netbackup/ext/db_ext/oracle/scripts/`

4. From the File menu, click Browse File System for Restore Scripts. The xbp dialog box appears.

5. Select the restore script from the Files pane.

6. On the Restore menu, click Restore Database Using Selected Scripts. The xbp_confirm dialog box appears.

7. Click OK.

   A NetBackup process called `bphdb` starts the NetBackup for Oracle script on the client.

8. View the status of the script execution.

   a. On the Restore menu, click Report Progress Of Restore.... The xbp_progress dialog box will appear.

   b. Select the log file for your restore.

      The Contents of Selected Log File pane displays only the status of the script execution. A status `=0` message indicates that the script was successfully completed. Go to Step 9 for a detailed status report. For a status other than 0, refer to the Troubleshooting section of this manual.

9. View the log file for the NetBackup operation.

   a. Change directories to the `bphdb` log directory.

      ```
      cd /usr/openv/netbackup/logs/bphdb
      ```

   b. Open the log file with the tail option.

      ```
      tail -f log.mmddyy
      ```

A `Restore completed SUCCESSFULLY` message indicates a successfully completed NetBackup operation.

Executing an Oracle Script From a Command Line

If you know the pathname of the Oracle script that initiates the restore, you can start it from the UNIX command prompt. For example, to perform a database restore at the UNIX prompt you might enter:

```
/oracle8/scripts/database_restore.sh
```

The UNIX shell starts the database restore by executing the Oracle script file. The Oracle script file contains commands to execute `rman`.

`rman` Command

Execute the `rman` command from the UNIX command line on the client, by using the appropriate RMAN command script file as a parameter. This example assumes that you are logged on as an Oracle administrator.

```
rman target 'internal/oracle@ORCL' rcvcat 'rman/rman@RCAT'
  cmdfile '/oracle8/scripts/database_restore.rcv'
```

Alternate Client Restores

With NetBackup for Oracle on UNIX you have the option to restore a database to a client other than the one that originally performed the backup. The process of restoring data to another client is called alternate client restore.

The following describes how to restore backups to an alternate client using RMAN.

Server Configuration

Ensure that the NetBackup server is configured to allow alternate client restore. The administrator can remove restrictions for all clients by creating the following file on the Netbackup master sever:

install_path`/netbackup/db/altnames/No.Restrictions`

Or the administrator can restrict clients to restore only from certain other clients by creating and then adding client names to the file:

install_path`/netbackup/db/altnames/`*client_name*

Where *client_name* is the name of the client allowed to do the alternate client restore.

See the *NetBackup System Administrator's Guide - UNIX* for details.

Alternate Client Configuration

The same user name (UNIX account) that was used for the Oracle database backup must be used for the alternate client restore.

Perform the following on the alternate client host if you want to restore RMAN backups that are owned by another client.

1. Enable a network connection to the RMAN catalog database, which was used by the original client.
2. Set ORACLE_SID to the same value as the original database.
3. Set the environment variable, NB_ORA_CLIENT, to the original client.
4. Create an `init`*SID*`.ora` file for the previously set ORACLE_SID.
5. Grant write permission to the directory where the datafiles will be restored.
6. Start up the database in the nomount state.
7. Start `rman`, connecting to the catalog.

8. Connect to the target database without using a userid / password.

9. Run an RMAN restore script.

Example

In this example assume:

- Original client is camel
- Alternate client is giraffe
- Server is lion
- ORACLE_SID is test
- UNIX user is ora8 on both camel and giraffe.

1. Create the file, *install_path*`/netbackup/db/altnames/giraffe`, on server lion and edit it to contain the name camel.

2. Log in to giraffe as ora8.

3. Set `SERVER=lion` in *$ORACLE_HOME*`/bp.conf`.

4. Modify the network `tnsnames.ora` file to enable RMAN catalog connection.

5. Set the environment variables ORACLE_SID to test and NB_ORA_CLIENT to camel.

6. Make sure the directory exists and has appropriate access permissions.

   The datafiles will be restored to a directory path with the same name they had when backed up.

7. Create an `inittest.ora` file.

8. Start up the database.

```
$svrmgr1
SVRMGR> connect internal
Connected.
SVRMGR> startup nomount
ORACLE instance started.
...
SVRMGR> exit
$
```

9. Run rman.

```
$rman rcvcat rman/rman@rcat
Recovery Manager: Release 8.0.5.0.0 - Production
RMAN-06008: connected to recovery catalog database
RMAN> connect target
RMAN-06006: connected to target database: test (not mounted)
RMAN> @restore_database.rcv
```

Where restore_database.rcv is the RMAN script containing the restore commands. See the *Oracle Backup and Recovery Guide* for details.

Performing Other RMAN Actions

To execute script files for database operations other than backups or restores, it is recommended that you execute the rman command directly rather than using NetBackup. For the rman command script syntax and examples, see the *Oracle8 Server Backup and Recovery Guide*.

Troubleshooting 5

NetBackup, NetBackup for Oracle on UNIX, and the Oracle8 Recovery Manager all provide reports on database operations. These reports are useful for finding errors associated with those applications.

This chapter contains the following sections.

- NetBackup and NetBackup for Oracle on UNIX
- Troubleshooting Procedure
- Backup or Restore Errors

NetBackup and NetBackup for Oracle on UNIX

The following describes troubleshooting logs and reports generated by NetBackup products.

NetBackup Advanced BLI Agent Logs

The NetBackup server and client software allow you to set up detailed activity logs for troubleshooting problems that occur outside of either NetBackup for Oracle on UNIX or the Oracle8 Recovery Manager. See the *NetBackup Troubleshooting Guide - UNIX* or the *NetBackup Troubleshooting Guide - Windows NT/2000* for a complete description of activity logs. Also see the *install_path*/netbackup/logs/README.debug file.

Note These logs do not reveal errors that occur during the execution of the Oracle8 Recovery Manager, unless those errors also affect NetBackup for Oracle on UNIX. Your best sources for Oracle error information are the logs provided by the Oracle.

Enable the NetBackup Advanced BLI Agent logs by performing the following steps.

1. Create the following directories on the client:

   install_path/netbackup/logs/bphdb
   install_path/netbackup/logs/dbclient
   install_path/netbackup/logs/bpbkar
   install_path/netbackup/logs/tar

   ```
   % cd install_path/netbackup/logs
   % mkdir bphdb
   % mkdir dbclient
   % mkdir bpbkar
   % mkdir tar
   ```

2. Set the access permissions to 777 on these log directories.

   ```
   % chmod 777 bphdb
   % chmod 777 dbclient
   % chmod 777 bpbkar
   % chmod 777 tar
   ```

The following sections describe the logs created when you create the log directories. Use a text editor to view the contents of the logs.

bphdb Directory on the Client

The *install_path*/netbackup/logs/bphdb directory contains the following types of logs. These logs are a good starting place to determine what type of error occurred.

`obk_stdout.`*`mmddyy`*

Unless redirected elsewhere, NetBackup places NetBackup for Oracle script output in this file.

`obk_stderr.`*`mmddyy`*

Unless redirected elsewhere, NetBackup places NetBackup for Oracle script errors in this file.

`log.`*`mmddyy`*

`bphdb` is the NetBackup Database Backup binary. This log contains debugging information for the `bphdb` process. NetBackup for Oracle on UNIX uses this client process for NetBackup for Oracle script execution. It is invoked when an automatic backup schedule is executed.

`dbclient` Directory on the Client

The *install_path*`/netbackup/logs/dbclient` directory contains the following execution log.

`log.`*`mmddyy`*

This log contains debugging information and execution status for the Oracle processes linked to the library provided with NetBackup for Oracle on UNIX.

`bpbkar` Directory on the Client

The *install_path*`/netbackup/logs/bpbkar` directory contains the following execution log.

`log.`*`mmddyy`*

This log contains debugging information and execution status for the BLI Backup process, which performs the actual data transfer of Oracle datafiles for proxy backups.

`tar` Directory on the Client

The *install_path*`/netbackup/logs/tar` directory contains the following execution log.

`log.`*`mmddyy`*

This log contains debugging information and execution status for the NetBackup process, which performs the actual data transfer of Oracle datafiles for proxy restores.

NetBackup Server Reports

NetBackup provides other reports that are useful in isolating problems. One such report is All Logs Entries on the server. See the *NetBackup System Administrator's Guide* for a description of this and other reports.

Oracle8 Recovery Manager Utility Logs

The Oracle8 Recovery Manager uses a command language interpreter, and can be executed in interactive or batch mode. You can specify a log file on the command line to record significant RMAN actions. The syntax is:

```
msglog 'logfile name'
```

Troubleshooting Procedure

Note For brevity, this chapter refers to the `libobk` module as API. Many media manager vendors also call the `libobk` module DMO (Database Module).

The following products are properly installed and configured:

- NetBackup 3.4
- Oracle RDBMS 8i
- NetBackup Advanced BLI Agent 3.4

1. When verifying your installation, ensure that the following two NetBackup for Oracle on UNIX binaries exist:

   - `bphdb` resides on the client and is used by both the NetBackup scheduler and the graphical interface to start backups and restores. The main purpose of `bphdb` is to execute an Oracle script file that in turn calls `rman`. Only the RMAN script is required when `rman` is executed from the command line.
   - `libobk` is a shared library module containing functions callable by the Oracle8 Recovery Manager. This library is loaded when Oracle8 Recovery Manager is started. The actual name of this binary is dependent on the operating system.

     On Solaris it is `libobk.so.1`

     On HP-UX, it is `libobk.a` (64-bit)

     On HP-UX, it is `libobk.sl` (32-bit)

2. Check that both the NetBackup server and client software are working properly. That is, check that normal operating system files can be backed up and restored from the client. The NetBackup client must be running the same version of software as the NetBackup server.

   The following NetBackup logs are turned on in VERBOSE mode:

   - On the client: `dbclient`, `bphdb`, `bpbkar`, `tar`, and `bpcd`.
   - On the master server: `bprd`, `bpsched`, and `bpdbm`.
   - On the host with the storage unit: `bpbrm`, and `bptm`.

These logs may become very large, especially `bpsched` and `bpdbm`. Ensure that enough free disk space exists in the log directory disk partition.

Backup or Restore Errors

An Oracle8 Recovery Manager backup error can originate:

- On the NetBackup side

  An error can be from the API, the NetBackup server or client, or Media Manager.

- On the Oracle side

  The error can be from the Oracle8 Recovery Manager, or the target database.

VERITAS suggests that you use the following steps when troubleshooting a failed operation:

1. Check the logs to determine the source of the error.
2. Troubleshoot each stage of the backup or restore.

The following sections describe these steps in detail.

Check the Logs to Determine the Source of the Error

Determine whether the error is from NetBackup or the Oracle8 Recovery Manager.

If the error came from NetBackup, go to "Troubleshoot Each Stage of the Backup or Restore" on page 139.

Otherwise, have the Oracle database administrator or the NetBackup administrator look at the problem.

The majority of the problems setting up NetBackup and RMAN concern the relinking process not working, an error with the RMAN script that is being executed, issues with the NetBackup class being set up incorrectly, or problems allocating tape devices / media.

Note Ensure that server and user-directed regular file system backups work for the client prior to debugging the Oracle RMAN backups. Try backing up the Oracle binaries.

- Always start by running `rman` from the command line rather than having NetBackup scheduler execute a script. These troubleshooting steps assume you are running `rman` from the command line.
- If NetBackup and Oracle's `rman` are communicating, you should see a log in *install_path*`/netbackup/logs/dbclient`. If a log file exists and you are having problems, this usually means a configuration problem within the NetBackup class or media availability.

```
cd install_path/netbackup/logs/dbclient
ls
   log.123098
```

If you have a log file, then NetBackup and Oracle RMAN are linked correctly.

- If NetBackup and Oracle's RMAN are NOT communicating, you will NOT see a log in *install_path*/netbackup/logs/dbclient.

```
cd install_path/netbackup/logs/dbclient
ls
```

Logs will not be created in this directory if the permissions are not set for the Oracle user to write the directory. Full permissions is best, chmod 777.

Verify that the RMAN script works correctly. That is, verify that the RMAN script will back up to `allocate channel c1 type disk` instead of `allocate channel c1 type sbt_tape`.

If RMAN can back up to disk and not to tape, and no log file exists in `dbclient`, then the relink did not work correctly.

Troubleshoot Each Stage of the Backup or Restore

The following explains the sequence of events for an Oracle8 Recovery Manager initiated action and suggests solutions for problems that can occur at each point in the sequence.

1. `rman` starts.

A backup or restore can be started in any of the following ways:

- Command line from the system prompt such as:

  For Oracle8 Recovery Manager Backup or Restore:

  ```
  % rman target user/pwd[@TNS alias] \
      rcvcat user/pwd[@TNS alias]\
      cmdfile RMAN script file name
  ```

  where *RMAN script file name* is fully qualified.

- Manually from the administrator interface on the master server.
- Automatically by an automatic backup schedule.

If an error occurs at this point, check the Oracle8 Recovery Manager log.

2. Oracle8 Recovery Manager verifies its environment, then issues requests to the API.

Some information such as the NetBackup version, API versions, and trace filename, and NetBackup signal handlers are registered with the Oracle8 Recovery Manager.

An error at this point is usually due to a problem with client and server communication. Check the messages in the bprd, bpsched, and bpcd logs for clues. Also verify the bp.conf entries on the client.

3. Oracle8 Recovery Manager issues a backup or restore request.

The API gathers necessary parameters, and then sends the backup or restore request to the NetBackup server. The API waits until both the server and client are ready to transfer data before returning.

Oracle8 Recovery Manager requests a backup or restore by passing the following to the API:

- Filename it wants NetBackup to use
- Mode (backup or restore)
- Actual name of the Oracle data file
- Information about the target database

The API gathers information from the following:

- Environment
- User's bp.conf and *install_path*/netbackup/bp.conf files on the client
- Information file from bphdb to be used as parameters to the bprd process

The API then sends this information to the master server's bprd process.

To troubleshoot a problem in this part of the first sequence, examine the *install_path*/netbackup/logs/dbclient/log.*mmddyy*.

- If the bprd process failed, check the logs for bprd, bpbrm, and bpsched.

  A failure at this point is frequently due to bad NetBackup server or Oracle class configuration parameters:

  NetBackup can usually select the correct Oracle class and schedules but not always when there are several Oracle classes in its database. Try setting the SERVER and CLASS values in the bp.conf file on the client or by using environment variables. For example, the following C Shell setenv commands specify the Oracle class, schedule, and server for NetBackup to use:

  ```
  setenv NB_ORA_CLASS classname
  setenv NB_ORA_SCHED schedule
  setenv NB_ORA_SERV NetBackup server
  ```

4. Oracle8 Recovery Manager issues read or write requests to the API, which then transfers data to or from the NetBackup server.

A failure here is probably due to NetBackup media, network, or timeout errors.

5. Oracle8 Recovery Manager tells the API to close the session.

   The API waits for the server to complete its necessary actions (backup image verification and so on) and then exits.

   An error can originate from either NetBackup or Oracle8 Recovery Manager.

   - Oracle8 Recovery Manager will abort if they encounter an error while reading a data file during the backup (for example, if Oracle blocks are out of sequence). It also aborts if NetBackup sends a bad backup image during the restore.
   - NetBackup may return an error code to the API if for some reason it could not complete the backup successfully.

Index

A

Activity logs
  description 134
Administrator-directed backups
  definition 4
Alternate client restore 57, 62, 74, 129
API
  error 138, 141
  libobk module 137
Automatic backup
  class 122
  schedule
    bphdb log 135
    manual backup 122
Automatic-scheduled backups
  definition 4
  file list 62

B

Backup
  see also Automatic backups
  automatic
    using scripts 122
  manual of a class 112, 115
  user directed
    with xbp 123
Backup Policy schedule
  automatic backups 53, 60, 70
  configuration 53, 60, 69
BLI Backup
  defined 13
bp.conf
  troubleshooting 140
bpbrm log
  verbose mode 137
bpcd
  verbose mode 137
bpdbm log
  verbose mode 137
bphdb 137
  description 137
  information file used as parameters 140
bphdb log 134, 135
  verbose mode 137
bphdb process 123, 127
bplist
  example 126
bplist log
  verbose mode 137
bprd log
  verbose mode 137
bpsched
  verbose mode 137
bptm log
  verbose mode 137
Browse
  with xbp 125

C

Caution
  script usage 42
Class
  adding 46
  automatic backup schedule 54, 61
  Backup Policy schedule 53, 60, 69
  file list 62
Class configuration
  java interface 46
  Windows NT/2000 interface 64
Client
  class 46
  install 24
Commands
  allocate channel 10, 80
  backup 10, 119
  bplist 126
  change 120
  copy 119

crosscheck 119
crosscheck backupset 118
delete expired backupset 119
list 121
register database 118
report 121
reset database 118
restore 119
resync catalog 120
rman 80
execute scripts 131
performing restore 128
script syntax 131
send 79, 80, 109
send channel 80
send device type 80
set duplex 109
switch 119
Configuration
database class 58, 64
Maximum Jobs per Client 43, 45, 64
media manager 40
test 112, 114, 115

D

Default-Policy schedule
Backup Policy schedule 53, 60, 69
Directory
xbp
directory to search 123, 127

E

Environmental variable
user-directed backup 124
Error
information 134
script name 42
Error checking 122
Example
bplist 126
RMAN script 98
Execution log 135

F

Failed operation
troubleshooting 138
File list
configuration 62
Full backup 4
retention period 53, 55, 60, 61, 69, 71

I

Incremental backup
Automatic-scheduled backups 4
Install
NetBackup client software 24, 63
NetBackup server software 24

J

Java interface
class configuration 46
set maximum jobs per client global attributes 43
testing class configuration 112

L

libobk,shared library module 137
Logs
error 134
troubleshooting 134

M

Manual backups 112, 115, 122
Maximum Jobs per Client 64
Media Manager
configuration 40
definition 4

N

NetBackup definition 4

O

Oracle script
description 82
Oracle8 Recovery Manager
errors 138
example RMAN script 98
installation 34
rman script 82

P

parms operand 80
Path
xbp
to browse for backups 125
to search for backup scripts 123
to search for restore scripts 127

R

Reports
database operations 133
Restore
alternate client 57, 62, 74, 129
user directed

with xbp 127
rman change command 41
RMAN script
example 98

S

Schedule
automatic backup 122
Scripts
cold_database_backup.sh 83
cold_database_backup_level1_cumulative.rcv 98
cold_duplex_database_backup_full.rcv 107
cold_duplex_database_backup_full.sh 91
database_restore.rcv 104
hot_database_backup.sh 88
hot_database_backup_level0.rcv 100
hot_tablespace_backup_level1_differential.rcv 103
pit_database_restore.rcv 105
RMAN 6, 10
scheduler 122
Server install 24
Set Maximum jobs per client global attribute
java interface 43
Windows NT/2000 interface 45
xbpadm interface 44
Shared library module, libobk 137

T

Testing class configuration
java interface 112
Windows NT/2000 interface 115
xbpadm interface 114
Troubleshooting logs 134

U

User-directed backup, and restore
definition 4

V

VERBOSE mode 137
Verifying installation 137

W

Windows NT/2000 interface
class configuration 64
set maximum jobs per client global attributes 45
testing class configuration 115

X

xbp
backup procedure 123
browse procedure 125
restore procedure 127
xbpadm interface
set maximum jobs per client global attributes 44
testing class configuration 114