Paragon Drive Backup Enterprise Server Edition

Best Practices for Oracle Database

Contents

1 Introduction

This paper addresses various aspects of an Oracle database data protection by using Paragon Drive Backup Enterprise. It describes the concepts, limitations and best practices for Paragon Drive Backup Enterprise to protect "no downtime" operational solutions based on Oracle database. All mentioned recommendations are generic and not specific for a certain Oracle database application.

1.1 About Drive Backup

Paragon Drive Backup Enterprise is a backup tool that implements the best of volume imaging techniques for reliable, fast and convenient data backup and restoration. The program includes end-user tools for building and automating recovery and replication procedures. It implements high-performance algorithms for intelligent data analysis and processing, provides an optimized manipulation for a large set of filesystems that covers all popular filesystems for Windows and Linux platforms and more features.

1.2 Backup Concepts

1.2.1 File-Level Backup and Volume Imaging

There are two concepts about backup subject. A *file-level backup* is oriented to store separate files. A *volume-level backup* or *imaging*, is oriented to store whole filesystem of a volume.

A *file-level backup* naturally provides an intuitive and flexible way to select objects to store. File-oriented backup tools allow to choose any combination of both local and network accessible files. A file-level data restoration allows to selectively restore only damaged files without affecting other ones. However, there are important file-related system objects which are not files and usually cannot be stored, restored and even accessed from a file level.

A *volume imaging* can store files and any associated metadata including distribution information, security data, quotas, extended attributes, named streams, multiple hard and symbolic links and so on. Imaging tools generally provide higher backup performance because they do not involve filesystem drivers to the process. In addition, they can backup offline filesystems including ones not being supported by a host operating system. A data restoration generally does not require a host operating system to run, so that imaging technique is a perfect choice for system cloning and disaster recovery tools. Disadvantages of volume imaging are that it cannot be applied to remote resources and a general ineffectiveness of backup and restore of selective files within the volume-imaging framework.

1.2.2 Data Consistency

The fundamental requirement to backup is saving of *data consistency*. This means that if applications are stopped, and data are restored, and applications are restarted, they will run smoothly with restored data.

A *data consistency* is conditionally divided into a *physical* and *logical* consistency. A *physical consistency* means storing information about involved files and file-related attributes. A *logical* consistency means an application-level correctness of stored data, which in turn means an accurate or at least auto-repairable state of business data. An automatic correction of minor inconsistency of business data is usually provided by the *transactions mechanism*, which is a basis of modern technologies of information processing.

1.2.3 Offline and Online Backup

As regards to the data consistency concept, offline data are in consistent state (with the only condition that an application or a system was shut down correctly). Offline data archiving is referred to as *offline backup*. Its advantages are ensured data consistency, increased backup speed (due to absence of concurrent data access), resource saving and low impact to a system performance.

The disadvantage is that offline backup is not applicable for "24x7 availability" systems. For continuously working systems *online backup* methods should be applied.

1.2.4 Snapshots

The great challenge to online backup is to provide a coherent state of all open files and databases involved in a backup, under the condition that applications may continue writing to a disk.

The "volume snapshot" concept has met that challenge. A *snapshot* is a point-in-time copy of a volume involved to a backup process. It must be quickly created at a period when applications do not write to disk. Once snapshot has been created, a backup utility copies data from it while applications continue working with an original volume. Modern snapshot technologies reduce *backup window* down to few seconds.

There are hardware and software based snapshot solutions. *Hardware snapshots* such as the *Split-Mirror* technique instantly provide an independent copy of a whole data set, they make no impact to the system performance and can participate in off-host backup solutions. Disadvantages are that hardware snapshots are naturally hardware-dependent, double requirements to storage resources and some time is required in order to re-synchronize a hardware snapshot after it has been used, with the actual disk data.

Instead, *software snapshots* are hardware-independent and resource-saving solutions. The disadvantage is that they cannot make a copy of a whole data set instantly, so that they run and make additional computing load to a host system until a backup routine is ended and a snapshot is destroyed.

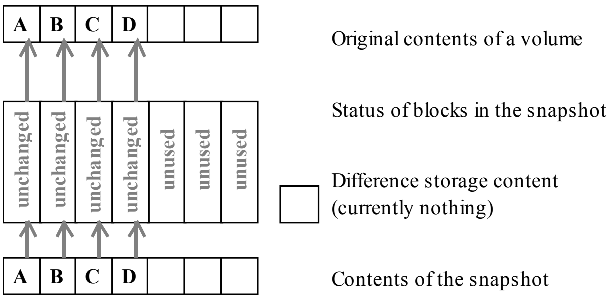
Paragon Drive Backup provides the original online backup technique (named *Paragon HotBackup*) and supports any snapshot technologies that can participate in the MS VSS framework.

1.2.5 Copy-On-Write

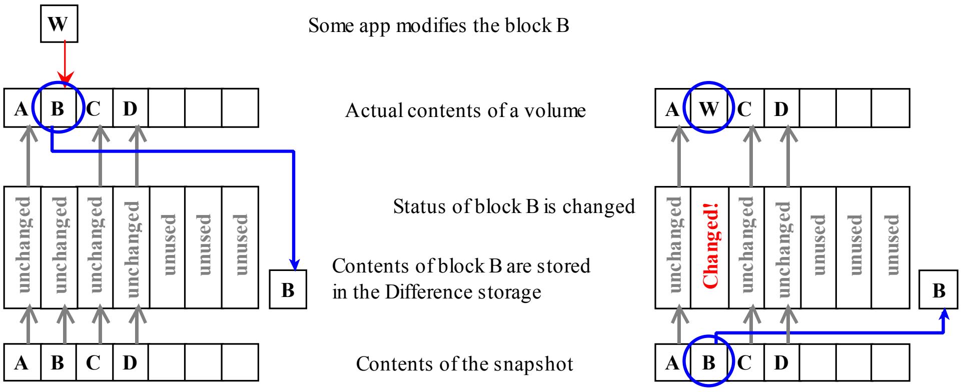
The *Copy-On-Write* (COW) method predominates among software snapshot techniques. It is based on the idea of preserving original contents of modified blocks before modifications are written to a disk, in a special *difference storage*. COW maintains an original volume in the actual state and stores its original content at the moment of snapshot creation in the difference storage.

A system-level agent, which is responsible for snapshot maintaining, must keep track of all changes being made on a volume. On first attempt to overwrite a block, the snapshot agent copies original block's content to the difference storage and then allows to modify that block. When a backup utility acquires some block from the snapshot, the agent determines its status and takes a "changed" block from the difference storage while "unchanged" block is taken from the original volume. The following illustrates COW basics:

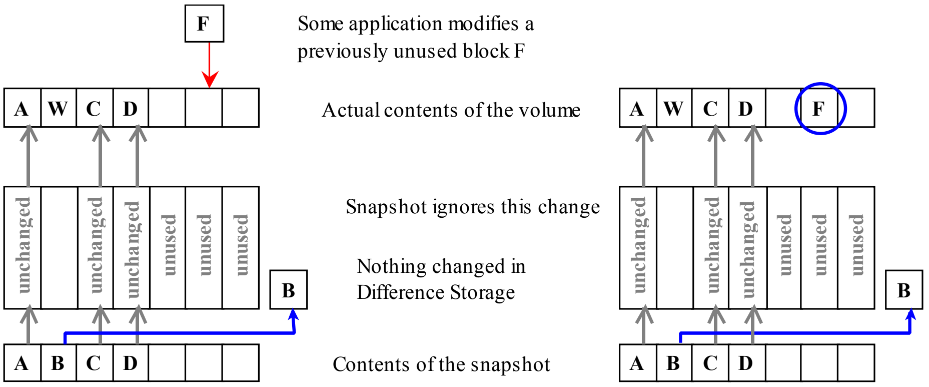
1. A volume was partially in use. Blocks A, B, C, D contained data. A COW based snapshot was created for this volume. The snapshot provider watches only the used blocks of a volume. Initially they are marked as "unchanged".



2. Some application tried to modify the block B. Before modifications are made, the snapshot provider copies contents of the block B to the difference storage. The block is now marked as "changed". Then the block B is updated. The snapshot will redirect all queries to the block B to data stored in the difference storage. Queries to blocks A, C and D will be directed to appropriate blocks of the volume.


W
Some app modifies the block B
A
B
C
D
Actual contents of a volume
A
W
C
D
Status of block B is changed
unchanged
unchanged
unchanged
unchanged
unused
unused
unused
Changed!
B
Contents of block B are stored
in the Difference storage
Contents of the snapshot
A
B
C
D

3. Some application tried to modify the block F, which were not originally in use. This block was not included in the snapshot. The snapshot provider does not take care fo this change. Nothing is changed in the snapshot data.



1.2.6 Data Synchronization

Another problem for online backup functionality is that applications may temporarily hold open files in logically inconsistent state. The snapshot techniques do not solve that problem as it concerns solely to business application operation. The true reason of the problem is that applications are unaware about a backup routine running and do not synchronize their data.

Microsoft has made a great attempt to solve the problem. The snapshot backup framework referred to as Volume Shadow Copy Service (VSS) has been built in latest versions of Windows (exactly to Windows XP, Windows Server 2003 and Windows Longhorn). VSS includes mechanisms for notifying applications about a backup, interchanging of related information between VSS participants and synchronizing execution of software components involved to the process.

VSS has several significant limitations:

- only VSS compliant applications can benefit from VSS framework.
- VSS is a local solution that works within a single host.
  Remote applications and distributed data systems aren't controlled by VSS.
- VSS currently works to its full capacity on Windows 2003 only.

1.2.7 Write Inactivity Paradigm

There are other (partial) solutions for synchronization problem that are applicable for non VSS compliant software. There is a popular solution based on the paradigm of *Write Inactivity Period* (WIP) that was introduced by St.Bernard Software company in the middle of 1990-th.

It is supposed that business applications for intensive information processing use transactions mechanisms. A *transaction* collects I/O writes into a compact group in order to reduce a chance of incomplete transaction committing if a failure of any type occurred. Data are in consistent state between transactions, so that a period when application(s) do not write to a disk is the best moment for a snapshot capture. This period is referred to as WIP – *Write Inactivity Period*.

2 Technology Overview

2.1 Paragon HotBackup Description

Paragon HotBackup is an online backup technology for Win'NT+ family operating systems. It was developed in 2001 and integrated to all company's backup solutions in 2002-2003. Currently it supports all versions of Windows NT4, 2000, XP and 2003 (including x86, IA64 and AMD64 versions).

2.1.1 Hotbackup Concepts

HotBackup is not a snapshot technology. However its concepts appear to be similar to ones used in software snapshot technologies. In particular, a sort of WIP observation and COW scheme are implemented.

During an online backup, Drive Backup uses the kernel mode driver HOTCORE.SYS in order to monitor and control write activity of applications and an operating system. The driver intercepts disk I/O requests and implements the most time-critical part of COW scheme while the Drive Backup utility includes disk data analysis and archiving functions.

The HOTCORE driver is installed during the standard Drive Backup setup procedure, and it is the reason why the system restart is required in order to complete the setup procedure. HOTCORE does nothing until it is activated by the Drive Backup. In the idle mode the driver bypasses any calls, makes no impact to the disk subsystem performance and only takes few kilobytes of system memory.

2.1.2 How it Works

HOTCORE driver is activated only in case the online backup is performed. In an offline backup mode, the driver is not involved to the process.

- Drive Backup activates the HOTCORE driver in the beginning of the "physical" backup.
- HOTCORE waits for a pause between I/O writes on the targeted volume.
- When the pause is observed, the driver takes a "snapshot".

Within the framework of HotBackup, a "snapshot" is a map of blocks to be protected by the COW scheme against loosing their original contents. The process requires the close cooperation between the driver and the utility. Finally, the "protection area" of the snapshot includes only used blocks with optional exception of *excluded files* (e.g. PAGEFILE.SYS and HIBERFIL.SYS). The embedded module for filesystem analysis allows to reduce this step down to few seconds or less.

- During the snapshot capture, the driver watches the volume against I/O writes. If a write occurred before "snapshot" is created, the step is repeated.
- Upon successful snapshot creation, the driver applies the COW scheme to the "protection area".
- The utility starts the backup process. Archived blocks are immediately excluded from the protection area, so that the area is shrinking during the backup.
- If a foreign application tried to modify a "protected" block, the driver preserves its original contents in a buffer. Initially, blocks are stored in a memory buffer. When it becomes near full, the utility moves blocks to temporary files (named like "X:\hb_nn.tmp", where X: is a drive defined in the Settings).
- Normally Drive Backup performs the fastest streamlined backup of used blocks. However, if temporary files grew very fast or became very large, the utility pauses the streamlined mode and begins emergency backup of buffered blocks.

In online backup the following time-related restrictions are used:

  - A snapshot must be created within 60 seconds.
  - Buffered data must be processed within 10 seconds.

These restrictions are hard-coded and cannot be changed. If any of these restrictions were not satisfied, the online backup session fails. If the backup operation was aborted by a user or the utility failed, the driver automatically switches to the idle mode in 10 seconds.

2.2 MS VSS Basics

Volume Shadow Copy Service (VSS) is an open system-level framework for snapshot backup solutions. It was developed by Microsoft in close cooperation with leading vendors of backup solutions. VSS is included in Windows XP and Windows Server 2003.

VSS provides mechanisms for notifying applications about a backup, interchanging of related information between VSS participants and synchronizing execution of software involved to the process. These mechanisms ensure consistent backup of online data for VSS compliant applications.

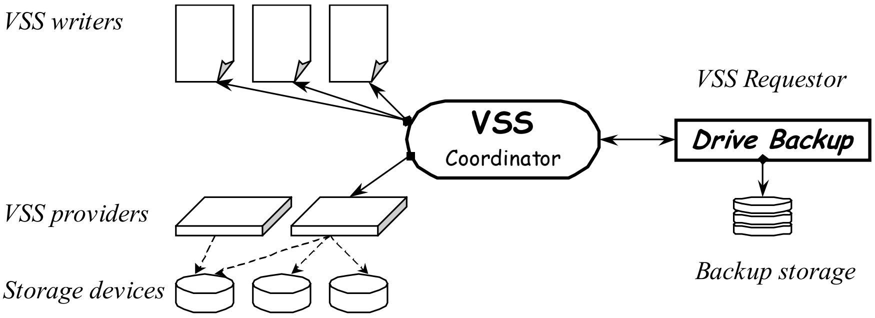
2.2.1 VSS Concepts

VSS is based on concepts of a snapshot and a volume shadow copy. Being invoked by a VSS aware backup utility, VSS creates snapshots for selected volumes and represents them as virtual read-only devices, which are referred to as *shadow copies of volumes*. Once shadow copies are created, a backup utility stores data from shadow copies while business applications continue writing to original volumes.

There are three kinds of software involved to the VSS framework:

- *Providers* (snapshot providers) – tools that create and maintain hardware or software snapshots.
- *Requestors* – utilities that acquire shadow copies, usually backup utilities.
- *Writers* – VSS compliant applications that hold open files on volumes, actual backup objectives.

Within the VSS framework, VSS writers are able to inform other VSS participants about files being in use, file groupping and restoration conditions. A group of files that constitute a whole entity in a business application and should be backed up together, is named a *writer's component*. For example, all files that constitute a database in Oracle are represented as a single writer's component. VSS writer can be an application itself or a special agent which provides VSS-to-application interaction.



VSS itself only coordinates activity of providers, writers and requestors. A standard Windows XP/2003 distribution includes the VSS coordinator, the universal software provider (VOLSNAP), several VSS writers for system components and the universal VSS writer for MS Desktop Engine (MSDEwriter). In the near future there is hope of an Oracle VSS writer.

2.2.2 How it Works

The comprehensive description of VSS can be found on appropriate Microsoft TechNet pages (e.g. see the descriptive topic "How Volume Shadow Copy Service Works",

http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/TechRef/2b0d2457-b7d8-42c3-b6c9-59c145b7765f.mspx). Below is only a brief description of MS VSS operation:

1. A VSS requestor (backup utility) invokes VSS to initialize the backup routine.
2. VSS acquires information from all VSS writers. As a result the *Writers Metadata Document* (WMD) is created. It contains distinctive names of applications and components, restoration parameters, list of files and other information.
3. A requestor receives WMD and makes a decision what to backup. It creates the *Backup Components Document* (BCD), which defines volumes, writers and components involved to the process and sends it to VSS. As an option, preferred VSS providers can be selected.
4. VSS commands to all involved VSS writers to finish running transactions and then "freeze". VSS waits until all involved writers complete that step.
5. Then VSS commands to appropriate VSS providers to create snapshots for selected volumes.

6. After snapshots are created, VSS allows writers to "thaw" and to write to disk. In addition, VSS asks writers if write operations were indeed suspended during the snapshot creation. If it was not the case, the whole backup procedure fails, snapshots are removed and the VSS requestor is notified.
7. Upon successful completion of above steps (within predefined time intervals), VSS creates *shadow copies* from snapshots and provides VSS requestor with appropriate references.
8. VSS requestor performs copying information from shadow copies….
9. Upon the process completion, the VSS requestor informs VSS about the backup completion status.
10. VSS informs all involved VSS writers about the backup completion status. VSS writers may use this notification to perform some specific actions (for example, Exchange truncates log files).

A shadow copy can be deleted immediately after the backup completes, or it can persist in the system. In the last case it can be mounted as an ordinary volume (this feature is available in Windows Server 2003).

2.2.3 MS VSS Limitations

MS VSS has several significant limitations:

- Only VSS compliant applications can benefit from VSS framework.
- VSS is a local solution working within a single host. Remote applications aren't controlled by VSS.
- VSS currently works to its full capacity on Windows 2003 only.

2.3 How Drive Backup Integrates with MS VSS

Storage management frameworks obviously provide benefits and can simplify storage management activity. Now Paragon Drive Backup Enterprise is adapted to participate in VSS framework operation as a requestor.

2.3.1 Enabling Backup via VSS

To perform backup operations via VSS services, select the "MS VSS technology" item in the "Hot processing technology" pull-down list on the "Hot processing options" tab in the program's Settings. Note that MS VSS service is available only in Windows XP, Windows Server 2003 and Windows Longhorn. In other operating systems, only Hotbackup technology is available for online backup.

Drive Backup provides a simplified VSS management, so that not all VSS options are controllable by a user. The program internally chooses only most reliable VSS operation modes.

2.3.2 How it Works

A VSS aware backup is performed in the following manner:

- A user chooses volume(s) to be backed up in the program's interface.
- When the "physical" backup operation begins, the program receives the compound WMD document from VSS (see #2 in the VSS description topic).
- Drive Backup determines VSS writers having components located on chosen volumes. These VSS writers are included to the BCD document (see #3). In other words, all VSS writers containing files on targeted volumes should participate in VSS operation at the snapshot creation (i.e. should be "frozen" and "thawed" by VSS).
- Drive Backup commands VSS to use the default order of VSS providers invocation: a hardware provider first (if available), a third-party software provider next (if available), the last is Microsoft's universal system provider VOLSNAP (always available).
- The program performs the volume-level backup of the created shadow copy set.
- After the operation completes, the shadow copy set is deleted.

Currently the program does not support restoration via VSS writers. In fact, VSS based restoration is generally just a file copying. Applications should be stopped, or appropriate components should be put in offline mode manually in order to be restored.

2.4 Choosing between Online and Offline Backup

The reasons to prefer offline backup are:

- Online backup via Hotbackup or VSS option is slower than offline backup.
- VSS initialization is long and generally unstable under high IO traffic on a targeted volume.

  - A resulting image produced by Hotbackup is slightly larger than one produced in offline mode because of non-sequential image structure, see Hotbackup description.
  - Neither of online backup options totally eliminates problems that are naturally inherent for online backup technologies in general.

VSS provide data consistency for VSS compliant applications only. Hotbackup does not guarantee 100% data consistency in any case, but only a very high probability of that. In fact, it provides perfect results for any applications that use transactions (e.g. Oracle).

Drive Backup provides some flexibility of choosing between offline or online backup mode. An user can choose between three modes: (a) "always offline backup", (b) "switch to online if a volume was in use" and (c) "always online backup". The differences between these modes are the following:

- (a) always offline mode:
  - The program does not backup volumes being in use. It suggests to reboot in order to complete the operation in the "Startup Bluescreen" mode.
  - If a volume wasn't in use, the program switches to exclusive use of the volume. No applications can access any files on the volume until the backup procedure is finished.
- (b) switch to online if a volume was in use:
  - If a volume was in use, the program performs the online backup. Other applications are allowed to access the volume during the operation.
  - If a volume wasn't in use, the program performs the offline backup. As it was described above, no applications can access any files on the volume until the backup procedure is finished.
- (c) "always online backup":
  - The program unconditionally performs the online backup. Other applications are allowed to access the volume during the operation.

3 Protecting Oracle Database

Oracle database is a general-purpose relational database server, which can scale from hosting simple databases to mission critical business applications. Oracle is arguably the most powerful and feature rich database on the market. The more your business depends on Oracle database, the more important it is to protect it.

This version of Paragon Drive Backup Enterprise does not have any Oracle specific functionality. For this reason, some considerations should be taken into account in order to provide a reliable Oracle databases backup.

The best practices presented in this guide are general principles, not guidelines for specific environments. This chapter discusses general requirements for both OLTP and DSS types of applications.

3.1 Database Files in Oracle

Designing and implementing an Oracle database is fairly intuitive, but it is important for database performance, maintenance and future growth. Understanding the relationship between database files, volumes and storage devices is essential for optimal use of DBE. This section discusses various layouts of an Oracle databases in relation to Drive Backup usage.

3.1.1 Database Physical Layout

The purposes of these best practices are to discuss backing up an Oracle database placed in a set of files.

It is a standard type of database layout. Database files can be located on any local volumes and can also be spread over multiple volumes. Drive Backup can be used for backing up of file-based databases with some weak constraints.

An Oracle database consists of *datafiles, control files, redo logs,* and *archive redo logs* when in archive mode, along with the *parameter file*, and the *password file* among a few other optional files.

The datafiles are used for long-term storing of data. It is used to store data associated with schema objects. For example, the datafiles hold the tables, indexes, views, and procedures also along with temporary data. Datafiles are the primary subject to backup but not the only one.

The *redo logs* are used for transaction managing and temporal holding most recent data changes. Depending on the database log-mode, redo logs can be necessary or unnecessary to be backed up. The *archive redo logs* keep history of redo logs evolution; it is used in advanced routines of database recovery (e.g. *point-in-time recovery* and *log shipping*). The *archive redo logs* are maintained only in case the database operates in the archive log-mode; if exist, it should be backed up.

The *control files* of a database store the status of the physical structure of the database; it is absolutely crucial to database operation. The control file contains a number of important information that Oracle needs to operate the database. The following pieces of information are held in a *control file*:

- the name of the database
- the file names of all datafiles that the database consists of
- the timestamp of when the database was created
- the checkpoint (all database changes prior to that checkpoint are saved in the datafiles)
- and information for *RMAN*.

When a database is mounted, its control file is used to find the datafiles and redo log files for that database. Because the control file is so important, it is imperative to back up the control file whenever a structural change was made in the database. For safety purpose, Oracle allows multiple copies of the control file placed in different locations; only one copy can be included to the backup set, however at the restoration all copies must be restored to their original locations and filenames.

The *parameter file* stores the initialization parameters of Oracle; it is read when Oracle instance is started. The parameter file is named like "INITxxx.ora" where the "xxx" substring coinside with the database name. In Oracle 9i and later versions there is one more file named like "SPFILEyyy.ora". The initialization parameter values that are currently in effect can be listed through `v$parameter` view:

```
SELECT name, value FROM v$parameter;
```

And finally, the *password file* ("PWDyyy.ora") stores the password information for database administrator (DBA). If the DBA wants to start up an Oracle instance there must be a way for Oracle to authenticate this DBA. That is if one is allowed to do so. Obviously, ones password can not be stored in the database, because Oracle can not access the database if the instance has not been started up. Therefore, the authentication of the DBA must happen outside of the database. There are two distinct mechanisms to authenticate the DBA: using the password file or through the operating system. The initialization parameter named `remote_login_passwordfile` specifies if a password file is used to authenticate the DBA or not. If it set either to `SHARED` or `EXCLUSIVE` value, a password file will be used. This value can be inspected from the output of the following query:

```
SELECT name, value
FROM v$parameter
WHERE name='remote_login_passwordfile';
```

In a file-based database all parts can be distributed over multiple volumes; this database layout is usually recommended for better performance. Within the framework of Oracle, the database files usually have the following default extensions:

File type	Extension	Filename examples
Datafiles	.DBF	D:\SampleDB\SYSTEM01.DBF
Control files	.CTL	D:\SampleDB\CONTROL01.CTL, E:\CONTROL02.CTL
Redo logs	.LOG	D:\SampleDB\Logs\REDO01.LOG
Initialization file	.ORA	INITorcl.ORA, plus SPFILEorcl.ORA (in v.9i+)
Password file	.ORA	PWDorcl.ORA

For example, if you installed your database using the default values you would more than likely have a datafile named "SYSTEM01.DBF", a control file named "CONTROL01.CTL", a redo log named "REDO01.LOG", a parameter file named "INITorcl.ORA" and "SPFILEorcl.ORA" depending on your Oracle version, and finally a password file named "PWDorcl.ORA". This of course is not the full list of files that would be installed but it gives a general idea of the main components.

3.1.2 Determining Database File Names

Some of mentioned filenames are automatically generated by Oracle while others can be customized by a user. For this reason, it is important to determine correct names and locations of database files that will be included to the backup set. Oracle provides a wide set of administration dictionaries (known as v$-views) that provide comprehensive information about various aspects of Oracle operation and database status. The v$-views can be queried by DBA from the SQLplus utility in order to retrieve information needed for making a correct backup of Oracle databases.

1. Use the `v$datafile` view to determine the list of the datafiles:

   ```
   SELECT name
   FROM v$datafile;
   ```

2. Use the `v$datafile` in conjunction with the `v$tablespace` view to determine the datafiles and the tablespace they belong to:

   ```
   SELECT t.name Tablespace, f.name Datafile
   FROM v$tablespace t, v$datafile f
   WHERE t.ts# = f.ts#;
   ```

3. Use the `v$controlfile` view to determine the list of the control files:

   ```
   SELECT name
   FROM v$controlfile;
   ```

4. Depending on the chosen backup-restoration scheme, you may have to ensure that the database is in *archive mode* and that the archive logs are saved. Use the following command in the SQLplus console to obtain the database log mode status:

   ```
   ARCHIVE LOG LIST;
   ```

5. Alternatively, use the `v$database` and `v$archive_dest` views to perform same as the previous command but by using PL/SQL statements only:

   ```
   SELECT name, log_mode FROM v$database;
   SELECT destination FROM v$archive_dest WHERE status='VALID';
   ```

6. Optionally, use the `v$archived_log` view to determine filenames of archived log files:

   ```
   SELECT name
   FROM v$archived_log
   WHERE archived='YES';
   ```

7. And of course the parameter file and password file also need to be backed up. The default location for the parameter and password file is `$ORACLE_HOME/dbs/` on UNIX and `%ORACLE_HOME%\database\` on Windows.

3.1.3 Tablespaces

A database is divided into one or more logical storage units called tablespaces. Tablespaces are divided into logical units of storage called segments, which are further divided into extents. Extents are a collection of contiguous blocks. Tablespaces are usually used to organize data into separate sections and also for performance reasons. For example, user data will be place in one tablespace, while index data will be placed in another.

There are several tablespaces that keep structure in an Oracle database. While user-defined tablespaces are important for business, the system tablespace is critical for Oracle operation. Below outlines the standard tablespaces in an Oracle database.

System	Every Oracle database contains a tablespace named SYSTEM, which Oracle creates automatically when the database is created. The SYSTEM tablespace is always online when the database is open.
Undo	Undo tablespaces are used solely for storing undo information. You cannot create any other segment types (for example, tables or indexes) in undo tablespaces. Each database contains zero or more undo tablespaces. In *automatic undo management* mode, each Oracle instance is assigned one (and only one) undo tablespace. Undo data is managed within an undo tablespace using *undo segments* that are automatically created and maintained by Oracle. When the first DML operation is run within a transaction, the transaction is bound (assigned) to an undo segment (and therefore to a transaction table) in the current undo tablespace. In rare circumstances, if the instance does not have a designated undo tablespace, the transaction binds to the system undo segment.
Temporary	When the SYSTEM tablespace is locally managed, you must define a default temporary tablespace when creating a database. A locally managed SYSTEM tablespace cannot be used for default temporary storage. If SYSTEM is dictionary managed and if you do not define a default temporary tablespace when creating the database, then SYSTEM is still used for default temporary storage. However, you will receive a warning in ALERT.LOG saying that a default temporary tablespace is recommended and will be necessary in future releases.

3.1.4 Datafiles

A tablespace in an Oracle database consists of one or more physical datafiles. A datafile can be associated with only one tablespace and only one database.

Oracle creates a datafile for a tablespace by allocating the specified amount of disk space plus the overhead required for the file header. When a datafile is created, the operating system under which Oracle runs is responsible for clearing old information and authorizations from a file before allocating it to Oracle. If the file is large, this process can take a significant amount of time. The first tablespace in any database is always the SYSTEM tablespace, so Oracle automatically allocates the first datafiles of any database for the SYSTEM tablespace during database creation.

3.1.5 Performance

There are multiple factors that affect database performance. The most significant of them are available memory and disk subsystem throughput. Available memory affects on read operations throughput, which is important for Decision Support Systems (DSS) performance. Disk subsystem throughput affects performance of both read and write operations. High write performance is essential for online transactions processing (OLTP) applications.

Most effective ways to increase disks throughput are using faster disks and using RAIDs. Striped RAID sets provide highest read/write performance which is found nearly a multiple of single disk performance.

RAID 0	Striped Disk Array
RAID 0+1	Mirroring of striped segments (RAID 1 over RAID 0)
RAID 3	Striped Disk Array with Isolated Parity
RAID 5	Striped Disk Array with Distributed Parity
RAID 10	Striping of mirrored segments (RAID 0 over RAID 1)

It is a good practice to place your production databases on a high-performance RAID set.

3.1.6 Reliability

While database backup provides a disaster recovery option, there are ways to reduce the chance of a failure. A hardware based method is to use fault-tolerant RAID configurations (for example, RAID-10 or RAID-5).

Additionally, a systems reliability can be increased by isolating database files away from an operating system, and from other intensively used file resources. It is a good practice to place your production database files on an isolated non-system volume, or even to distribute the database files among multiple non-system volumes.

3.1.7 Backup

First, to realize the concept of backup in an Oracle database environment, the concept of SCN (*System Change Number*) must be understood. The SCN is an increasing number associated with each *commit* operation. SCN serves in the role of a kind of clock within database, it indicates the current location of database components on the clock. SCN exists in control files, datafiles headers, online redo log files and archive log files.

The database open operation checks SCN values for all files, and confirms if the database is in a consistent state or an inconsistent state. When all files indicate the same SCN value, the database is in a consistent state. If the inconsistent state is detected, any appropriate recovery operation will be performed. The recovery operation to make the inconsistent database to be in a consistent state again requires the archive log files. Until the database is recovered to be consistent, the database can not be opened. Only when the database is shutdown normally, can it be expected that all files have the same SCN value and that it is in a consistent state.

For online backup of a tablespace, Oracle supports hot backup mode (`BEGIN BACKUP`), in which Oracle marks the mode and freezes the *checkpoint SCN* in the datafile header. Read/write operations to the datafile in hot backup mode are still active, and the user transactions are performed normally. With the freeze of the checkpoint SCN, the logging data unit in the redo log entry switches to logging full images of changed database blocks to the redo logs. Instead of recording how it changed a particular block (the *change vector*), it will log the entire image of the block after the change.

User-defined and system tablespaces along with control files, parameter files, password files, and archive log files (when in *archive mode*) must be regularly backed up in order to protect business applications from disaster.

The only exclusion from this rule is the temporary tablespaces (after Oracle 7.x) and the redo logs. They can be re-created upon successful distaster recovery. Temporary tablespaces are just that, temporary data and can be replaced. Redo logs are open files without any possibility to put in a "backup mode", that is the reason for archiving (copying) these online redo logs. Backing up online redo log files is not advised but could theoretically be backed up when performing cold backup with the database running in `NOARCHIVELOG` mode. However, this is also not needed as described in the following. Although it may seem that one should back up online redo logs along with the datafiles, control file, parameter files, and password file, this technique is dangerous. You should not back up online redo logs for the following reasons:

- The best method for protecting the online logs against media failure is by multiplexing them, that is, having multiple log members in each group, on different disks and disk controllers.
- If your database is in `ARCHIVELOG` mode, then the archiver is already archiving the filled redo logs.
- If your database is in `NOARCHIVELOG` mode, then the only type of backups that you should perform are closed, consistent, whole database backups. The files in this type of backup are all consistent and do not need recovery, so the online logs are not needed.
- You may accidentally restore backups of online redo logs while not intending to, thereby corrupting the database.

A number of situations are possible in which restoring the online logs cause significant problems to the database. The following sections describe scenarios that illustrate how restoring backed up online logs severely compromises recovery.

Unintentionally Restoring Online Redo Logs: Scenario

When a crisis occurs, it is easy to make a simple mistake. When restoring the whole database, you can accidentally restore the online redo logs, thus overwriting the current online logs with the older, useless backups. This action forces you to perform incomplete recovery instead of the intended complete recovery, thereby losing the ability to recover valuable data contained in the overwritten redo logs.

3.2 Backing up Oracle Databases with DBE

Some concerns should be taken into account in order to use Paragon Drive Backup Enterprise for successful backing up of an Oracle database. The following circumstances should be considered:

- Is a database spread over multiple volumes?

  - Should a database be backed up online, or is it acceptable to temporarily shutdown the database?
  - Which operating system is running on the host?

The governing factors that limit Drive Backup adaptability for Oracle backup are the following:

- This version of Drive Backup does backup network drives.
- This version of Drive Backup does not include Oracle Database specific agents. For this reason, databases are backed up like ordinary files. The WIP detection mechanism is used for database's data synchronization.
- Databases are always backed up in a mode that is effectively equal to the *full backup* (in terms of database backup types).

This version of Drive Backup is unaware of database logical structure. Drive Backup's "full" and "differential" modes refer to volume backup modes at block level, not to database backup modes. At restoration, a whole database will be rolled back to the pre-backup state including datafiles and transaction log. This behavior is effectively equal to the full offline database backup-&-restore.

3.2.1 Recommended Database Layouts

Oracle is sensitive to read performance and sensitive to write performance on the redo log files and on the archive log files. By using RAID fault tolerant volumes, many complications and expense can be avoided in the event of a disk failure. Thus, the following recommendations are given:

1. OS Volume - The OS should be installed on a RAID 1 disk volume. It is important that you do not need to restore/rebuild the OS in the event of a disk failure. This can be very time consuming and expensive. In addition, the Oracle binary files can be placed on this volume.
2. Redo Log Files - The Redo Log files should be placed on a RAID 1 or RAID 10 volume. The I/Os to the Redo Log files are 100% sequential and 100% writes, thus RAID 5 is inappropriate.
3. Datafiles - The Datafiles should be RAID 10 if the I/Os are 90% reads or less. If the I/O pattern is 90% or greater reads, then RAID 5 is OK. Again, your budget may help determine this.
4. Archive Log Files - The Archive Log files can either be RAID 10 or RAID 5. Archiving might take longer if it is RAID 5.

There are system-dependent recommendations:

5. In Windows Server 2003, set up Drive Backup to use MS VSS for online backup of databases distributed over multiple volumes. Choose the "Microsoft Volume Shadow Copy" item in the "Hot processing technology" pull-down list in the program settings. VSS gives benefits for backing up an Oracle database and enhances the chance of successful backup of distributed databases. It must also be noted that at this time Oracle is not VSS compliant but this VSS technology does provide extra benefits.
6. For now Oracle is not VSS compliant, so there is not any noticeable difference between Hotbackup and VSS options when backing up Oracle that is located only on one volume. However, when backing up Oracle that is spanned across multiple volumes VSS has the ability to synchronize the snapshot across all volumes, whereas Hotbackup does not. VSS is available in Windows XP/2003 only.
7. It must be noted that in Windows 2000 and NT 4.0 environments if an Oracle database is spanned across multiple volumes Drive Backup will be unable to perform a usable backup. If the database is located only on one volume, Drive Backup can back this up to various states, these various states will be discussed in more detail below under backing up an Oracle database. For these operating systems, only Hotbackup option is available. Hotbackup performs an asynchronous backup of multiple volumes, so that a distributed database may be archived in logically inconsistent state.
8. Distributed databases can be successfully backed up in offline mode. Shutdown the database before backing up volumes that contain a distributed database. This will provide in any case successful results.

A preferred system configuration could be like this:

- There is a separate storage (hard disk or another RAID set) dedicated for placing the system partition, and or for storing backup images.
- There is separate storage (hard disk or another RAID set) dedicated for placing the *redo logs*.
- There is a high-performance RAID set that is dedicated for placing production tablespace datafiles. For example, a RAID 10 provides both high reliability and performance levels.
- There is separate storage (hard disk or another RAID set) dedicated for placing the *archive logs*.
- *Control files* can be duplicated across all storage for added protection.

- Backup images are stored on a non-system volume, probably on a volume containing no critical data**.**

3.2.2 Unsupported Database Layouts

This version of Drive Backup has some limitations in relation to backing up an Oracle database:

1. A database that is entirely located on a mapped network drive cannot be backed up by using Drive Backup, because the program does not backup network drives.
2. A database that is partially located on a mapped network drive cannot be backed up by using Drive Backup, because not all database files can be stored.

Generally, a database partially located on a mapped network drive may be an obstacle for accurate data restoration for the whole volume. As Drive Backup is unable to backup completely such databases, it is unable to restore them correctly. To avoid possible problems caused by incomplete database restoration, do not apply a volume-level restoration of volumes containing distributed databases. Instead, a file-level restoration should be applied for such volumes. Use the *"Image Explorer"* utility for the file-level restoration from Drive Backup's backup images.

3. In Windows 2000 and NT 4.0 environments, distributed online database files cannot be successfully backed up in online mode (because of Hotbackup option synchronization limitations). Such databases can be accurately backed up in offline mode. Shutdown the databases before backing up volumes containing distributed database files.

3.3 Restoring Oracle Databases with DBE

This version of Drive Backup does not apply online restoration for an entire Oracle database (control files, datafiles, parameter files, etc.). Offline restoration is the only available method. A database migration-at-restoration is not supported as well.

A database can be restored in two ways. First, any database can be restored as a part of the volume restoration procedure. Second, database files can be manually be extracted from a volume's image and then manually placed to their original location.

The difference between these methods is the following:

- At *volume-level restoration*, all contents of a targeted volume are rolled back (point in time recovery). For example, if there were three datafiles on a volume, all of them are rolled back to the pre-backup state during the volume-level restoration. File system metadata are also restored and put into consistent state.
- The volume-level restoration is very fast and is able to recover volumes from scratch.
- At *file-level restoration*, one can extract from an image only required files. It provides selective data retrieval. Other volume's contents are not affected. However it is rather slow and requires a targeted volume to be healthy. File-level restoration cannot be used for recovering damaged volumes.

To restore a single datafile at file-level restoration with the database still open, one should take the tablespace containing the damaged datafile offline. After the datafile is restored, one should recover the tablespace by executing the following query:

```
RECOVER AUTOMATIC TABLESPACE <tablespace_name>;
```

The information to monitor the recovery process can be found in the alert log file "alert_xxxx.log". Once the datafile is recovered one can simply turn the tablespace back online by executing the following query:

```
alter tablespace <tablespace_name> online;
```

3.4 Choosing between Hotbackup and VSS Online Backup Options.

Drive Backup provides two options for snapshot-based online backup in Windows XP and Windows 2003. In Windows 2000 and Windows NT4.0, only Hotbackup option is available.

Underlying technologies (Hotbackup and MS VSS) are different by their concepts and features. As regards to backing up an Oracle database, these options exhibit different levels of operation stability and resulting data consistency. Following considerations can be taken into account when choosing between options:

- It must be noted that for right now Oracle is not VSS compliant but as reported from Oracle will be in the near future. The difference between online backup options is generally unimportant when backing up a single volume but a synchronized snapshot from multiple volumes is supported by VSS only.
- VSS option allows to create coherent backup of multiple volumes. With Hotbackup option, Drive Backup always performs an asynchronous backup of multiple volumes. This feature can be determinative for backing up databases distributed over multiple volumes (regardless of Oracle version) in conjunction with `alter tablespace begin backup` and `alter tablespace end backup` commands.
- Hotbackup requires less memory and disk resources to operate. For example, latest updates of VSS (for Windows XP and 2003) need at least 300MB of disk space per every created shadow copy (at the moment of snapshot initialization), while Hotbackup will consume disk space only under noticeable disk I/O traffic. In particular, it will take 300MB only under high disk load.
- Both VSS and Hotbackup may fail to initialize or fail to maintain a created snapshot under conditions of high I/O traffic on a volume being backed up. VSS requires large amount of free space to maintain a snapshot, while Hotbackup mostly needs high backup performance compared to SQL server's load. The backup performance depends on (a) throughput of archived volumes and backup storage and (b) compression throughput, which in turn depends mostly on CPU and memory speed.
- Hotbackup initialization is more stable in comparison to VSS. As concerns to an Oracle database, Hotbackup practically always initializes within predefined time intervals, while VSS may fail under high load of the Oracle database.
- Only latest versions of VSS are stable enough. It is highly recommended to install the latest service pack available and also check for latest hotfixes for Windows 2003 (or XP) related to VSS.

The tests have revealed a general instability of VSS initialization under a "stressed load" of the Oracle database (e.g. under lengthy server's load greater than 1-1.5 thousand modified pages per second). To solve such a problem, either use the Hotbackup option or suspend the database for a short time during the shadow copy process. This will be demonstrated next in more detail.

3.5 Examples of Using Online Backup Options

This section demonstrates how to use Paragon Drive Backup for online backup of an Oracle database for various levels of protection and data restoration in case of a database corruption. The use of Oracle's commands `ALTER TABLESPACE BEGIN BACKUP` and `ALTER TABLESPACE END BACKUP` are highly recommended in conjunction with both Paragon HotBackup and MS VSS technologies. This will be outlined in this section in detail for the results of when Oracle commands are not used and also for when they are used. The methods of retrieval of related database properties and their correlation to backup parameters were illustrated in the section "Determining Database File Names"**.**

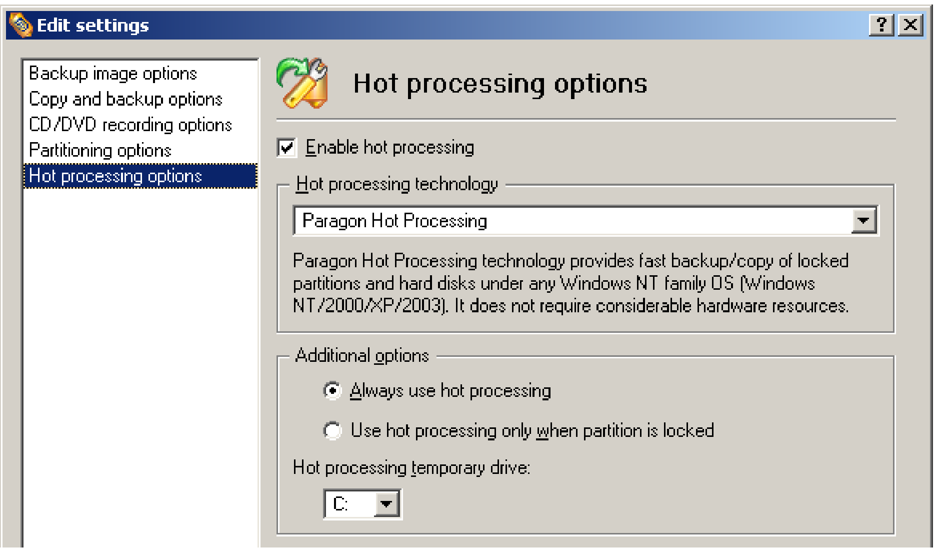
3.5.1 Example 1: Simplified Routines of a Database Backup and Restore

This example demonstrates how to backup and restore a complete database located on a single volume without the use of Oracle backup commands.

Conditions: the production database and its required files are placed on the dedicated volume D:. All database files (the *.dbf, *.ctl, *.log, init.ora, and Oracle binary files) are located on this volume.

Purpose: backup and restore the Oracle database instance from a hard drive crash.

1. Inspect database properties to ensure that all database files are located on same volume. See the section "Determining Database File Names" describes how to retrieve this information by using Oracle's v$-views, e.g. in conjunction with SQL*Plus utility.
2. Run Drive Backup, open Settings dialog (menu: "File"→"Settings…"). Go to the "Hot processing options" page, enable online backup and choose an online backup technology to implement: Paragon Hotbackup or Microsoft Volume Shadow Copy:



Edit settings

- Backup image options
- Copy and backup options
- CD/DVD recording options
- Partitioning options
- Hot processing options

Hot processing options

☑ Enable hot processing

Hot processing technology: Paragon Hot Processing

Paragon Hot Processing technology provides fast backup/copy of locked partitions and hard disks under any Windows NT family OS (Windows NT/2000/XP/2003). It does not require considerable hardware resources.

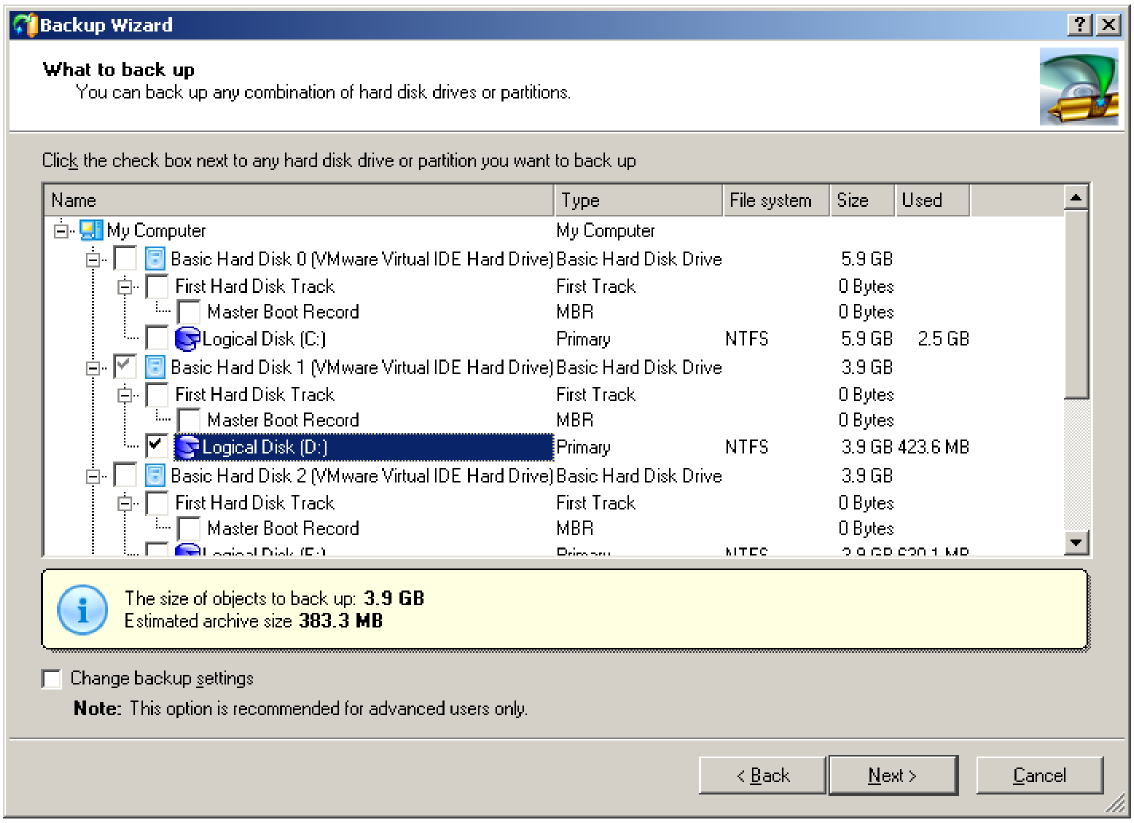
Additional options

● Always use hot processing

○ Use hot processing only when partition is locked

Hot processing temporary drive: C:

3. Run the Backup Wizard in order to set up the backup operation (no matter should it be a scheduled task or "run once" job). On the "What to back up" page, select the volume D: for backup:



Backup Wizard

What to back up

You can back up any combination of hard disk drives or partitions.

Click the check box next to any hard disk drive or partition you want to back up

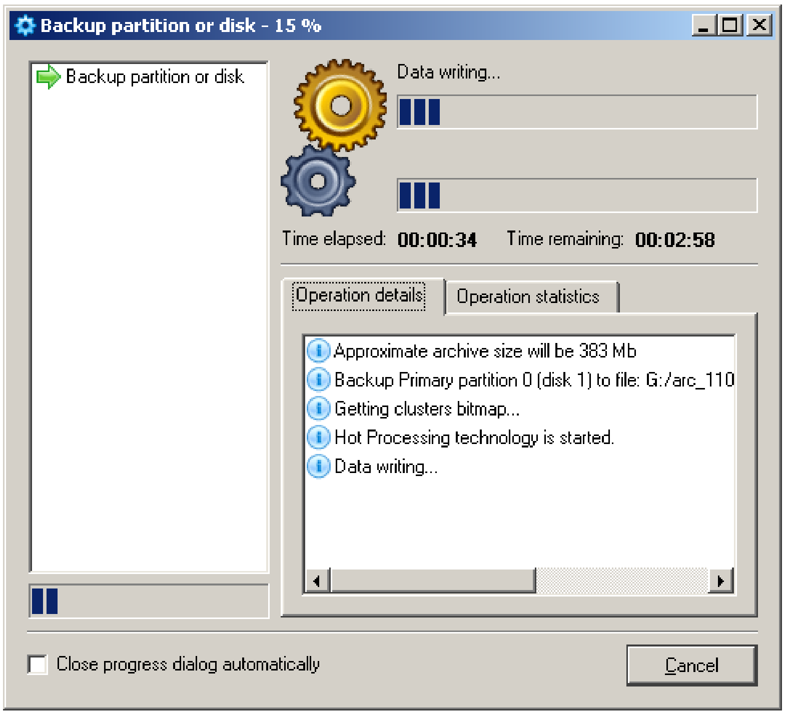
Name	Type	File system	Size	Used
My Computer	My Computer			
□ Basic Hard Disk 0 (VMware Virtual IDE Hard Drive)	Basic Hard Disk Drive		5.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
□ Logical Disk (C:)	Primary	NTFS	5.9 GB	2.5 GB
☑ Basic Hard Disk 1 (VMware Virtual IDE Hard Drive)	Basic Hard Disk Drive		3.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
☑ Logical Disk (D:)	Primary	NTFS	3.9 GB	423.6 MB
□ Basic Hard Disk 2 (VMware Virtual IDE Hard Drive)	Basic Hard Disk Drive		3.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
□ Logical Disk (E:)	Primary	NTFS	3.9 GB	630.1 MB

The size of objects to back up: **3.9 GB**
Estimated archive size **383.3 MB**

□ Change backup settings

Note: This option is recommended for advanced users only.

4. The program will make a backup image of the volume D: to the location that was selected



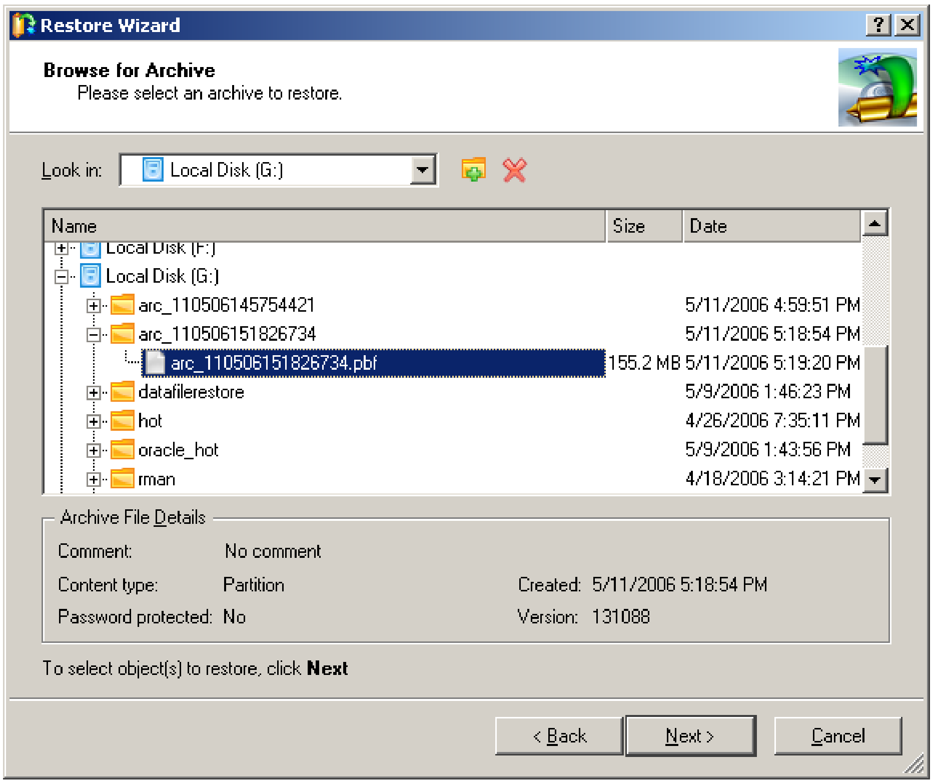
Data writing...

Time elapsed: 00:00:34 Time remaining: 00:02:58

Approximate archive size will be 383 Mb
Backup Primary partition 0 (disk 1) to file: G:/arc_110
Getting clusters bitmap...
Hot Processing technology is started.
Data writing...

Now suppose the D: drive has become corrupted and the complete drive must be replaced. All database features are unavailable. Now you need to restore the databases availability.

5. Run Drive Backup and restore the entire volume D: from the appropriate image. To do this, run the Restore Wizard and select appropriate image file (**.PBF**):



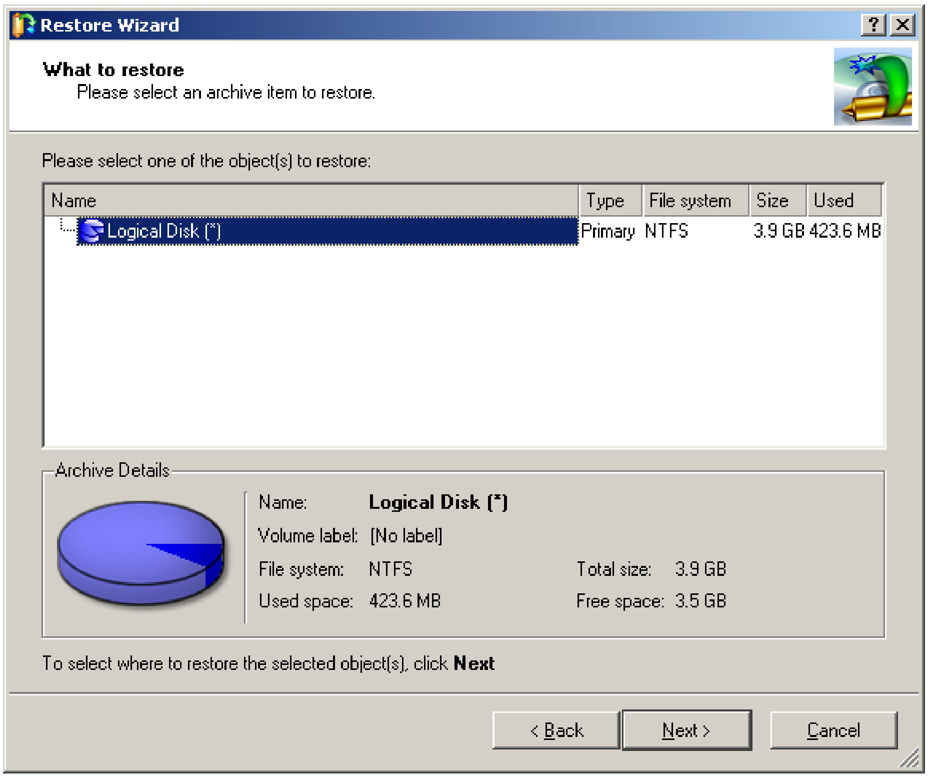
Name	Size	Date
Local Disk (F:)		
Local Disk (G:)		
arc_110506145754421		5/11/2006 4:59:51 PM
arc_110506151826734		5/11/2006 5:18:54 PM
arc_110506151826734.pbf	155.2 MB	5/11/2006 5:19:20 PM
datafilerestore		5/9/2006 1:46:23 PM
hot		4/26/2006 7:35:11 PM
oracle_hot		5/9/2006 1:43:56 PM
rman		4/18/2006 3:14:21 PM

Archive File Details

Comment: No comment
Content type: Partition Created: 5/11/2006 5:18:54 PM
Password protected: No Version: 131088

To select object(s) to restore, click **Next**

6. Select a volume stored in the image:



What to restore

Please select an archive item to restore.

Please select one of the object(s) to restore:

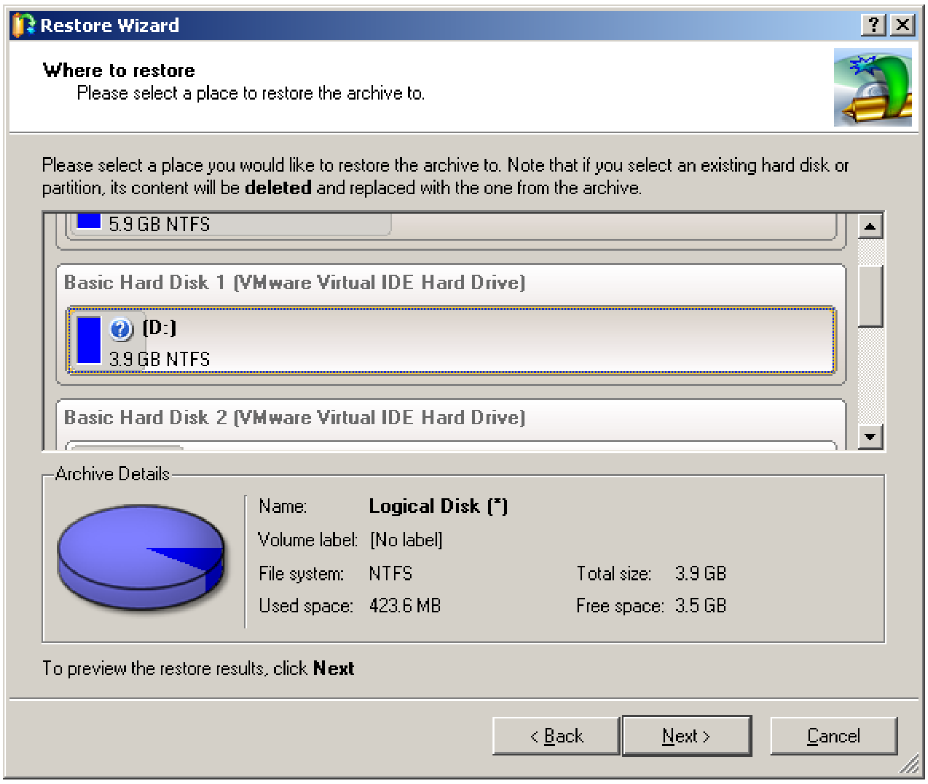
Name	Type	File system	Size	Used
Logical Disk (*)	Primary	NTFS	3.9 GB	423.6 MB

Archive Details

Name: **Logical Disk (*)**
Volume label: [No label]
File system: NTFS Total size: 3.9 GB
Used space: 423.6 MB Free space: 3.5 GB

To select where to restore the selected object(s), click **Next**

7. Select a volume on a disk where data should be restored:


Restore Wizard
5.9 GB NTFS
Basic Hard Disk 1 (VMware Virtual IDE Hard Drive)
(D:)
3.9 GB NTFS
Basic Hard Disk 2 (VMware Virtual IDE Hard Drive)
< Back
Next >
Cancel

Where to restore

Please select a place to restore the archive to.

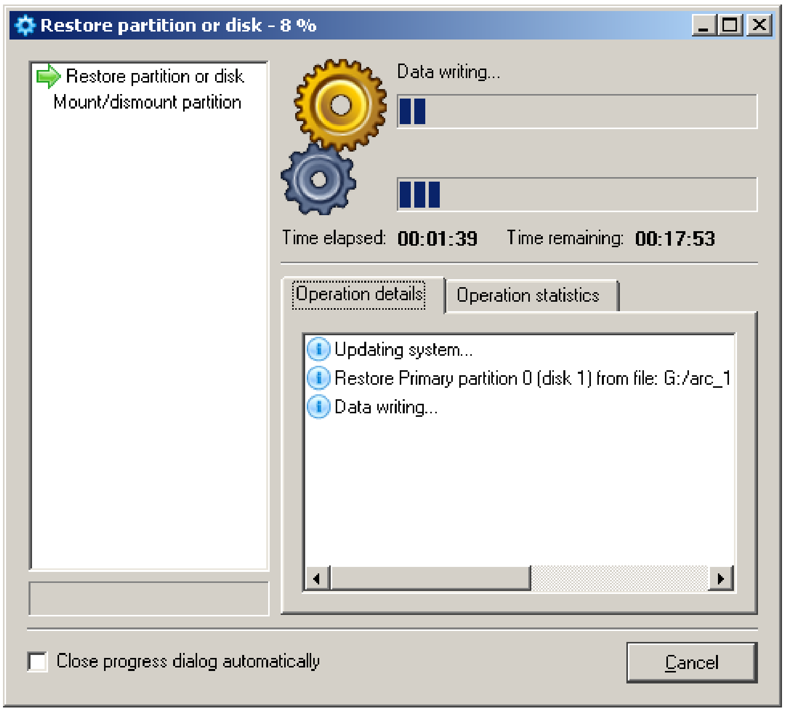
Please select a place you would like to restore the archive to. Note that if you select an existing hard disk or partition, its content will be **deleted** and replaced with the one from the archive.

Archive Details

Name: **Logical Disk (*)**
Volume label: [No label]
File system: NTFS Total size: 3.9 GB
Used space: 423.6 MB Free space: 3.5 GB

To preview the restore results, click **Next**

8. Drive Backup will restore the data for the volume after selecting apply:



Once the data from volume D: is restored the Oracle database can now be started.

9. Run SQL*Plus utility

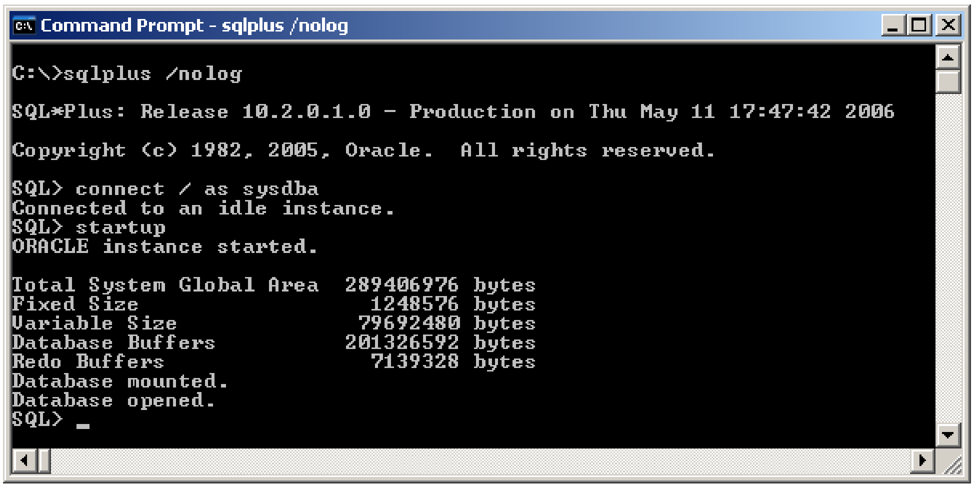
```
sqlplus /nolog
```

10. Connect to an idle Oracle instance:

```
CONNECT / AS SYSDBA
```

11. Run the `STARTUP` command:

```
STARTUP
```



```
C:\>sqlplus /nolog

SQL*Plus: Release 10.2.0.1.0 - Production on Thu May 11 17:47:42 2006

Copyright (c) 1982, 2005, Oracle.  All rights reserved.

SQL> connect / as sysdba
Connected to an idle instance.
SQL> startup
ORACLE instance started.

Total System Global Area  289406976 bytes
Fixed Size                  1248576 bytes
Variable Size              79692480 bytes
Database Buffers          201326592 bytes
Redo Buffers                7139328 bytes
Database mounted.
Database opened.
SQL> _
```

It must be noted that because the Oracle's commands:

```
ALTER TABLESPACE BEGIN BACKUP
ALTER TABLESPACE END BACKUP
```

were not used in this example, the restored database becomes in the *crash consistency state*, which is effectively equal to shutting off power or hardware resetting during a running instance. For this reason, the `startup` command will perform the automatic *crash recovery* without user intervention. The results of this operation can be viewed in the database's alert file. The default name and location for this file is:

```
<drive>:\oracle\admin\<dbname>\bdump\alert_<dbname>.log
```

For example, for the "testDB" database this file will be named as follows:

```
F:\oracle\admin\testDB\bdump\alert_testDB.log
```

Possible results could look like the following:

```
Beginning crash recovery of 1 threads
Thu May 11 13:56:01 2006
Started first pass scan
Thu May 11 13:56:03 2006
Completed first pass scan
46483 redo blocks read, 2055 data blocks need recovery
Thu May 11 13:56:04 2006
Started recovery at
Thread 1: logseq 14, block 186425, scn 0.0
Recovery of Online Redo Log: Thread 1 Group 4 Seq 14 Reading mem 0
Mem# 0 errs 0: E:\ORACLE\ORADATA\ORCL\REDO01A.LOG
Mem# 1 errs 0: E:\ORACLE\ORADATA\ORCL\REDO01B.LOG
Recovery of Online Redo Log: Thread 1 Group 5 Seq 15 Reading mem 0
Mem# 0 errs 0: E:\ORACLE\ORADATA\ORCL\REDO02A.LOG
Mem# 1 errs 0: E:\ORACLE\ORADATA\ORCL\REDO02B.LOG
Thu May 11 13:56:11 2006
Ended recovery at
Thread 1: logseq 15, block 28111, scn 0.764909
2055 data blocks read, 1998 data blocks written, 46483 redo blocks read
Crash recovery completed successfully
```

3.5.2 Example 2: Advanced Routine of a Database Backup and Restore

This example illustrates how to perform an advanced routine of database backup and restore located on multiple volumes running Windows XP/2003 with the use of Microsoft VSS and Oracle's backup commands. The demonstrated restore procedure allows to use advantages of archived log mode of Oracle databases.

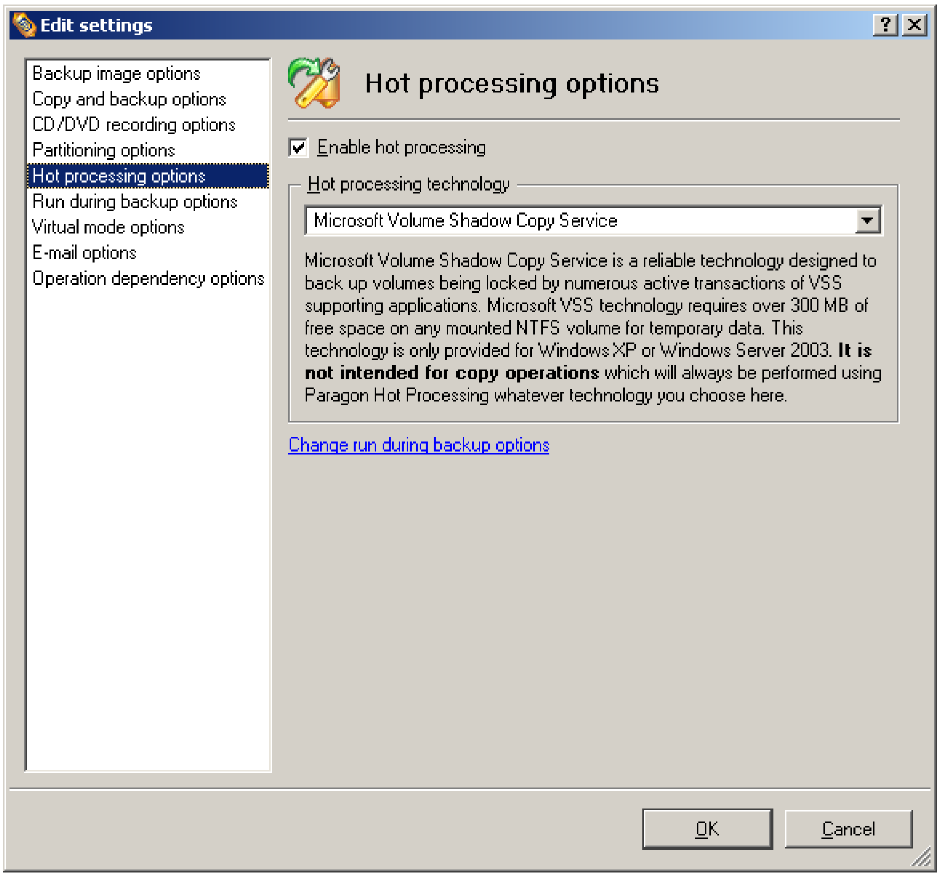
This is an advanced example of what one can do to take extra measures when backing up Oracle online in *archive mode* to ensure a proper *point-in-time recovery* in the event of a disaster. This will go into detail on using commands placed in the Drive Backup's "Run during backup options" to put the tablespaces in the backup mode, and also to copy the control file, and the archive logs to an external location before and after taking a synchronized snapshot from multiple volumes. Although there are many ways to backup and restore Oracle with Drive Backup Enterprise Server, this is one way for taking extra precautions.

The following will now demonstrate how to use Paragon Drive Backup in conjunction with Oracle backup procedures. This will give advanced ideas in protecting the system. The Paragon image along with the Oracle files being backed up will constitute the full backup set.

Conditions: the production database and its required files are spanned across multiple volumes D:, E:, F:. The archive logs and the first copy of the control file are located on drive D:, the redo logs and the second copy of the control file are located on drive E:, and the datafiles, the third copy of the control file, and Oracle binary files are located on drive F:

Purpose: backup and restore the Oracle database instance in the event of a disaster.

1. Inspect database properties to ensure that all database files are located on same volume. See the section "Determining Database File Names" describes how to retrieve this information by using Oracle's v$-views, e.g. in conjunction with SQL*Plus utility.
2. Run Drive Backup, open Settings dialog (menu: "File"→"Settings…"). Go to the "Hot processing options" page, enable online backup and choose the online backup technology "Microsoft Volume Shadow Copy". Choosing the "Microsoft Volume Shadow Copy" hot processing option allows for a synchronized snapshot off all volumes, whereas Paragon Hot Backup will not:



☑ Enable hot processing

Hot processing technology: Microsoft Volume Shadow Copy Service

Microsoft Volume Shadow Copy Service is a reliable technology designed to back up volumes being locked by numerous active transactions of VSS supporting applications. Microsoft VSS technology requires over 300 MB of free space on any mounted NTFS volume for temporary data. This technology is only provided for Windows XP or Windows Server 2003. **It is not intended for copy operations** which will always be performed using Paragon Hot Processing whatever technology you choose here.

Change run during backup options

3. Run Drive Backup, open Settings dialog (menu: "File"→"Settings…") and select the "Run during backup options" page.
4. Set up the "Execute before taking snapshot" parameter on the "Run during backup options" page. This parameter is purposed to run a command that will prepare business application(s) for backup. As concerns to Oracle operation, this command should run a PL/SQL script that will switch tablespaces of selected databases to the backup mode (`ALTER TABLESPACE BEGIN BACKUP`) and optionally place the databases in the `SUSPEND` mode. Below is an example of such script:

    (in SQL script)

    ```
    ALTER TABLESPACE system BEGIN BACKUP;
    ALTER TABLESPACE users BEGIN BACKUP;
       . . . . .
    ALTER SYSTEM SUSPEND;
    ```

5. Set up the "Execute after taking snapshot" parameter on the "Run during backup options" page. This parameter is purposed to run a command that will return business application(s) back to a normal operation mode. As concerns to Oracle operation, this command should run a PL/SQL script that will resume the database operation and take all tablespaces out of the backup mode (`ALTER TABLESPACE END BACKUP`). Additionally, this command should copy the control file and archive logs to an external backup destination. Below is an example of such script:

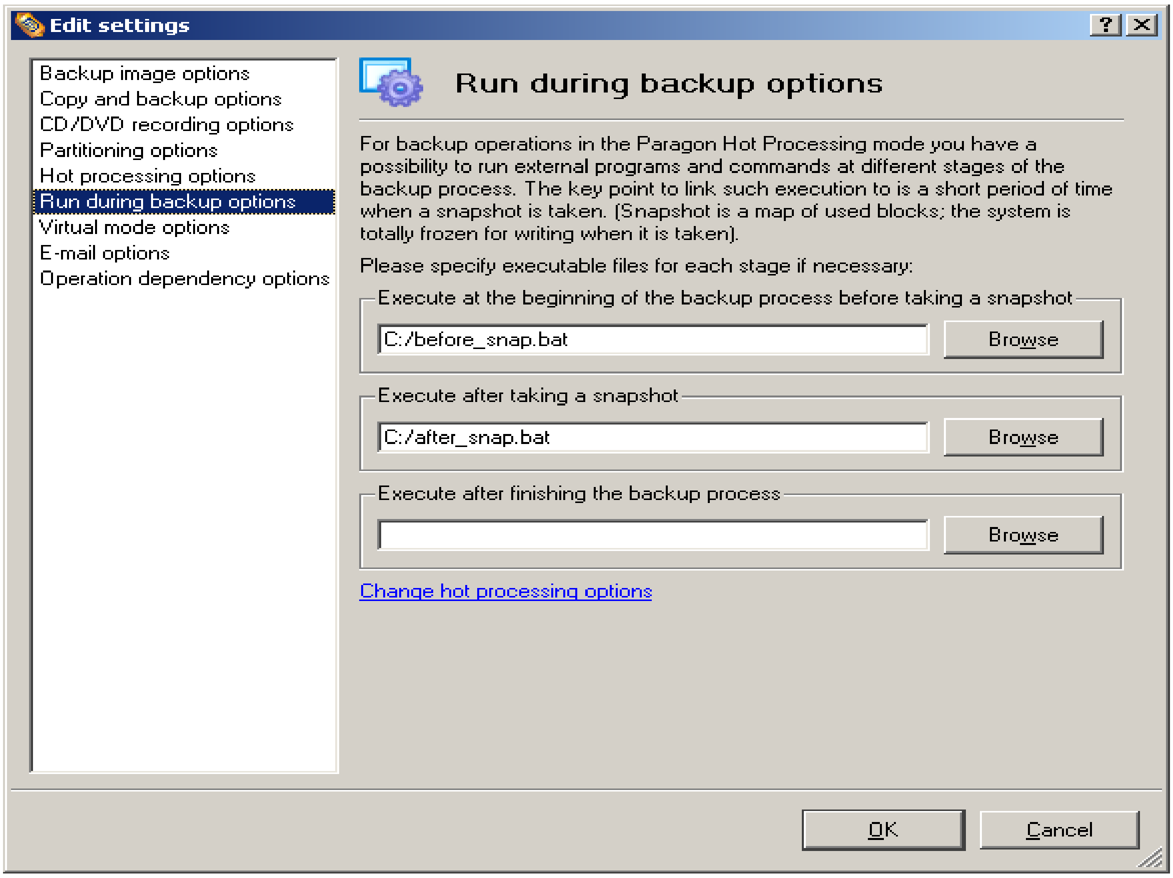
    (in BAT file)

    ```
    sqlplus /nolog <SQL-script>
    copy d:\oracle\oradata\<dbname>\archives\*.* g:\backup_destination
    ```

    (in SQL script)

    ```
    CONNECT <user>/<password>@<dbname> AS SYSDBA
    ALTER SYSTEM RESUME;
    ALTER TABLESPACE system END BACKUP;
    ALTER TABLESPACE users END BACKUP;
       . . . . .
    ```

```
ALTER SYSTEM BACKUP CONTROLFILE TO TRACE;
ALTER SYSTEM BACKUP CONTROLFILE TO g:\backup_destination
ALTER SYSTEM ARCHIVE LOG CURRENT;
```



Run during backup options

For backup operations in the Paragon Hot Processing mode you have a possibility to run external programs and commands at different stages of the backup process. The key point to link such execution to is a short period of time when a snapshot is taken. (Snapshot is a map of used blocks; the system is totally frozen for writing when it is taken).

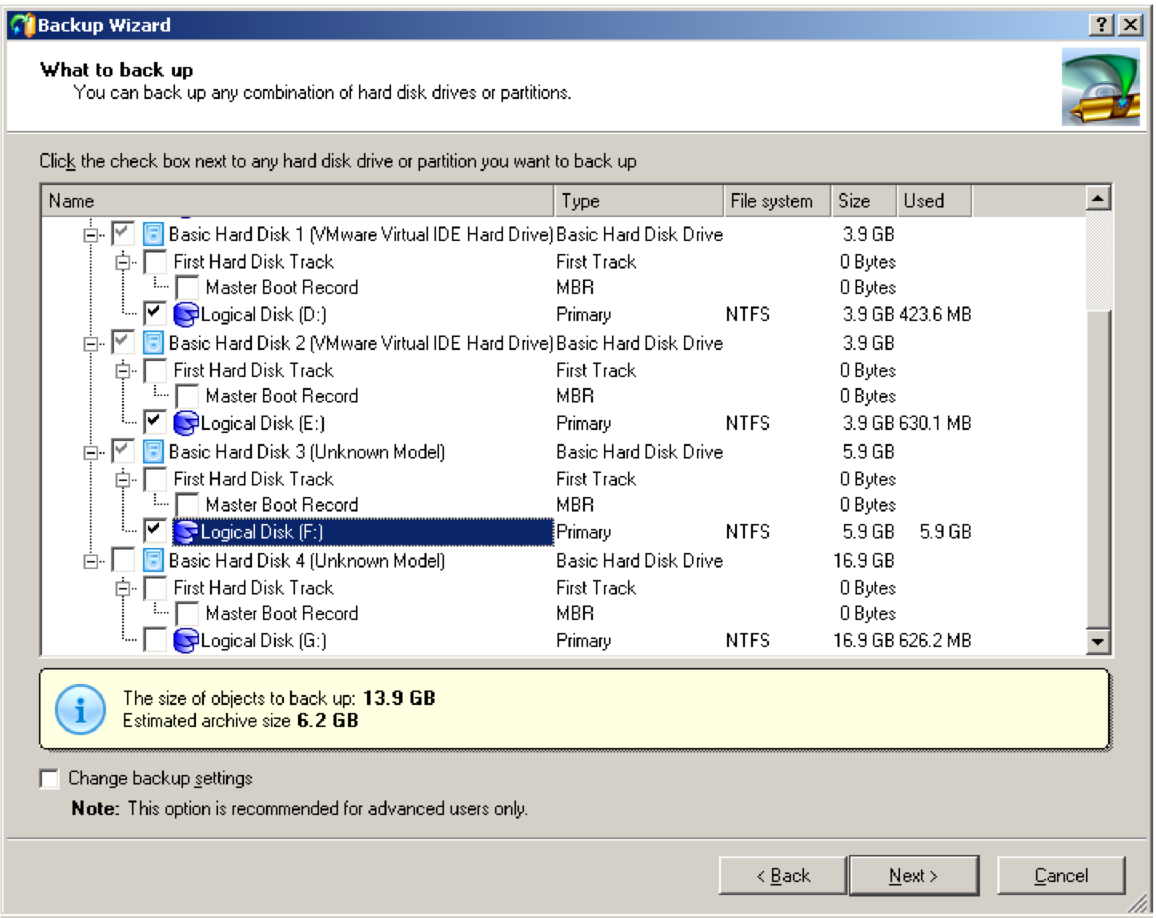
Please specify executable files for each stage if necessary:

Execute at the beginning of the backup process before taking a snapshot: C:/before_snap.bat

Execute after taking a snapshot: C:/after_snap.bat

Execute after finishing the backup process:

6. Run the Backup Wizard in order to set up the backup operation (no matter should it be a scheduled task or "run once" job). On the "What to back up" page, select the volume D:, E:, and F: for backup:



What to back up

You can back up any combination of hard disk drives or partitions.

Click the check box next to any hard disk drive or partition you want to back up

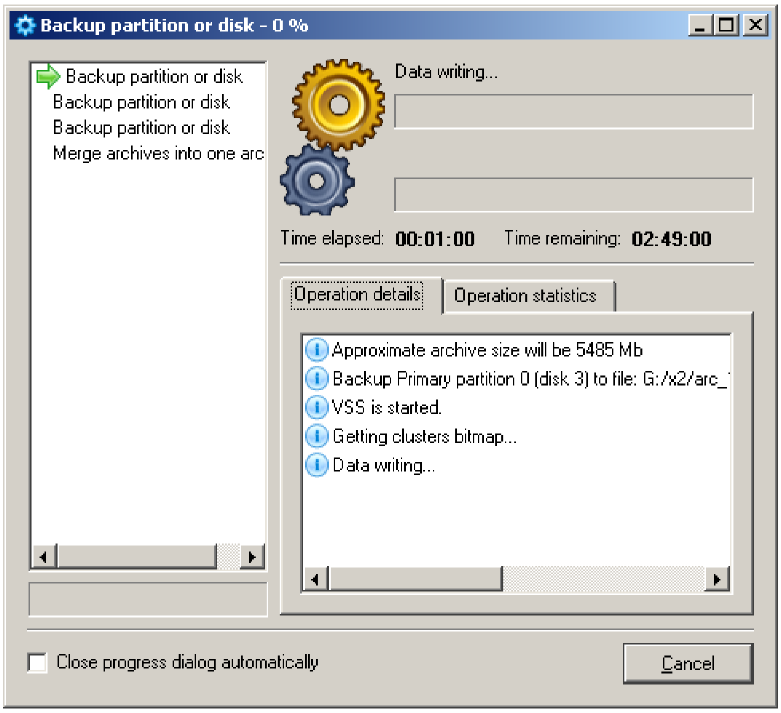
Name	Type	File system	Size	Used
☑ Basic Hard Disk 1 (VMware Virtual IDE Hard Drive)	Basic Hard Disk Drive		3.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
☑ Logical Disk (D:)	Primary	NTFS	3.9 GB	423.6 MB
☑ Basic Hard Disk 2 (VMware Virtual IDE Hard Drive)	Basic Hard Disk Drive		3.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
☑ Logical Disk (E:)	Primary	NTFS	3.9 GB	630.1 MB
☑ Basic Hard Disk 3 (Unknown Model)	Basic Hard Disk Drive		5.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
☑ Logical Disk (F:)	Primary	NTFS	5.9 GB	5.9 GB
□ Basic Hard Disk 4 (Unknown Model)	Basic Hard Disk Drive		16.9 GB	
□ First Hard Disk Track	First Track		0 Bytes	
□ Master Boot Record	MBR		0 Bytes	
□ Logical Disk (G:)	Primary	NTFS	16.9 GB	626.2 MB

The size of objects to back up: **13.9 GB**
Estimated archive size **6.2 GB**

□ Change backup settings

Note: This option is recommended for advanced users only.

7. The program will make a backup image of the volume D:, E:, and F: to the location that was selected:



Backup partition or disk
Backup partition or disk
Backup partition or disk
Merge archives into one arc

Data writing...

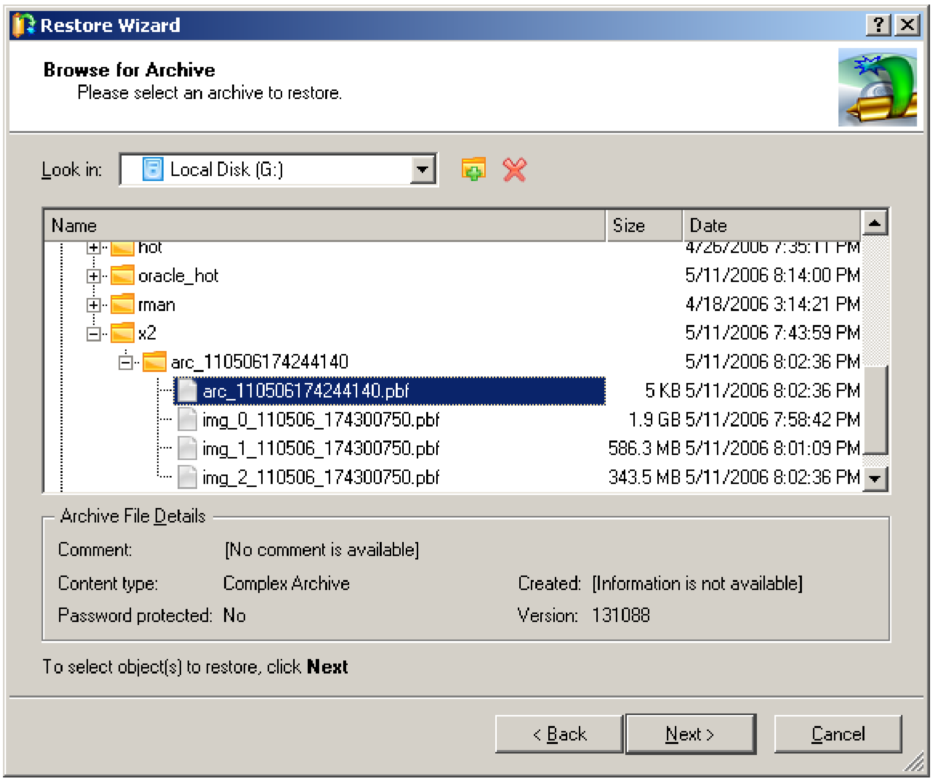
Time elapsed: **00:01:00** Time remaining: **02:49:00**

Approximate archive size will be 5485 Mb
Backup Primary partition 0 (disk 3) to file: G:/x2/arc_
VSS is started.
Getting clusters bitmap...
Data writing...

□ Close progress dialog automatically

8. Now suppose the computer has become unavailable in the event of a disaster and the system must be replaced entirely. After reinstalling or restoring the system drive and replacing all hard drives that were used for the Oracle install (D:, E:, and F:), a full restore and recovery can take place of the Oracle database instance.

9. Run Drive Backup and restore the entire volume D:, E:, F: from the appropriate image. To do this, run the Restore Wizard and select appropriate image file (**.PBF**). This step will have to be run for each volume separately.



Browse for Archive

Please select an archive to restore.

Look in: Local Disk (G:)

Name	Size	Date
hot		4/26/2006 7:35:11 PM
oracle_hot		5/11/2006 8:14:00 PM
rman		4/18/2006 3:14:21 PM
x2		5/11/2006 7:43:59 PM
arc_110506174244140		5/11/2006 8:02:36 PM
arc_110506174244140.pbf	5 KB	5/11/2006 8:02:36 PM
img_0_110506_174300750.pbf	1.9 GB	5/11/2006 7:58:42 PM
img_1_110506_174300750.pbf	586.3 MB	5/11/2006 8:01:09 PM
img_2_110506_174300750.pbf	343.5 MB	5/11/2006 8:02:36 PM

Archive File Details

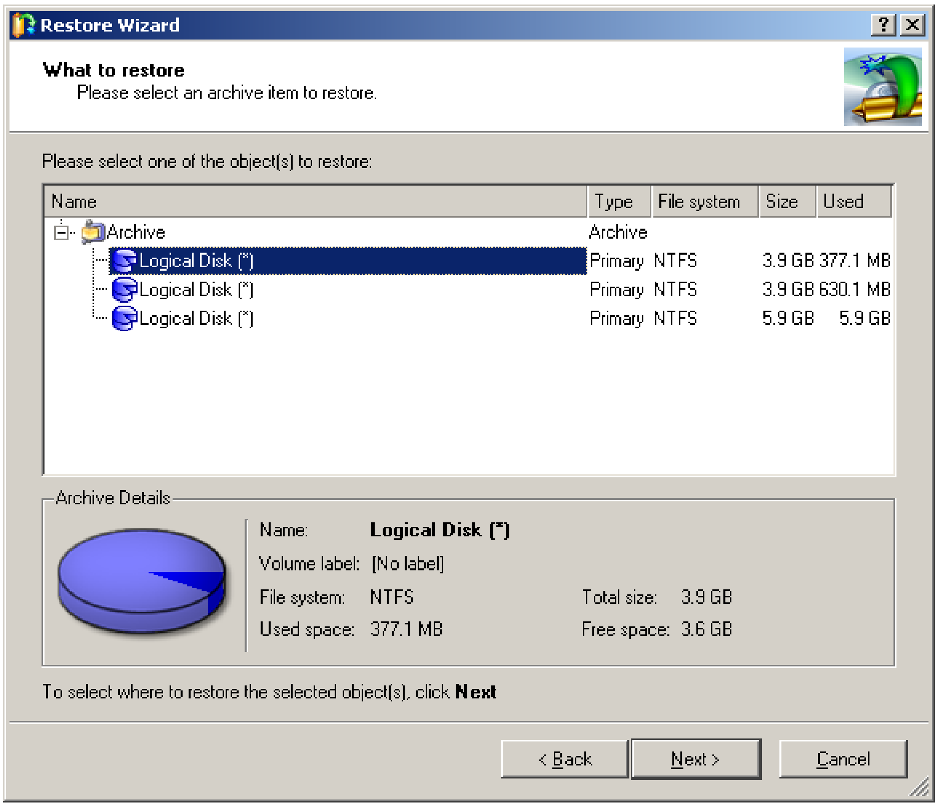
Comment: [No comment is available]

Content type: Complex Archive Created: [Information is not available]

Password protected: No Version: 131088

To select object(s) to restore, click **Next**

10. Select the first volume stored in the image:



What to restore

Please select an archive item to restore.

Please select one of the object(s) to restore:

Name	Type	File system	Size	Used
Archive	Archive			
Logical Disk (*)	Primary	NTFS	3.9 GB	377.1 MB
Logical Disk (*)	Primary	NTFS	3.9 GB	630.1 MB
Logical Disk (*)	Primary	NTFS	5.9 GB	5.9 GB

Archive Details

Name: **Logical Disk (*)**

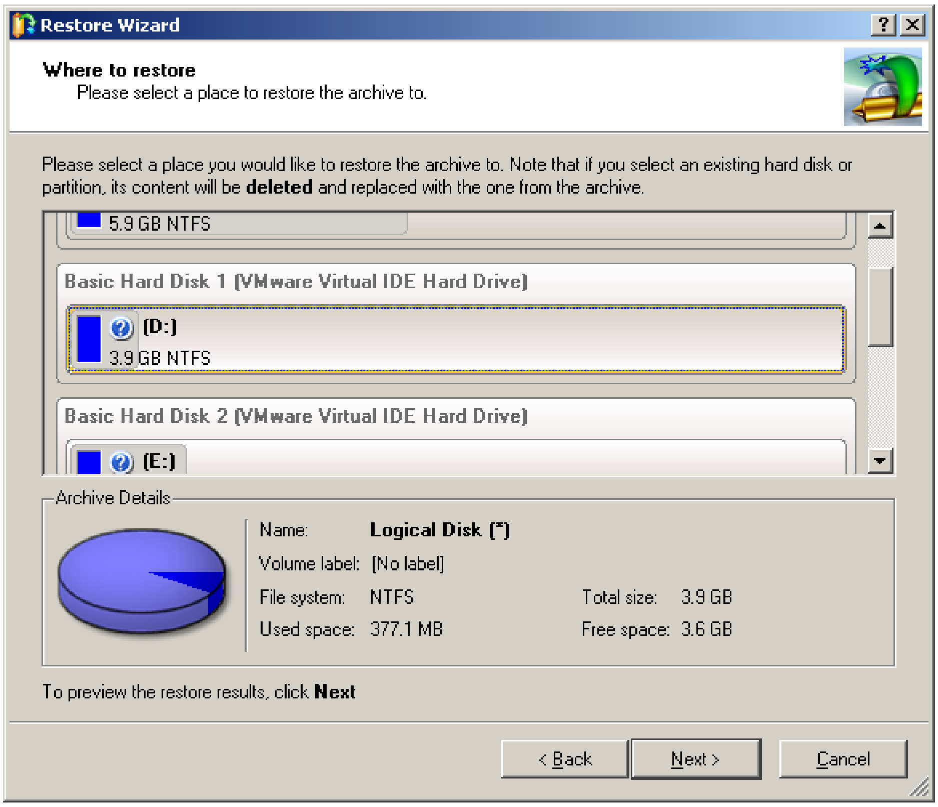
Volume label: [No label]

File system: NTFS Total size: 3.9 GB

Used space: 377.1 MB Free space: 3.6 GB

To select where to restore the selected object(s), click **Next**

11. Select a volume on a disk where data should be restored, in this case D:



Where to restore

Please select a place to restore the archive to.

Please select a place you would like to restore the archive to. Note that if you select an existing hard disk or partition, its content will be **deleted** and replaced with the one from the archive.

5.9 GB NTFS

Basic Hard Disk 1 (VMware Virtual IDE Hard Drive)

(D:)
3.9 GB NTFS

Basic Hard Disk 2 (VMware Virtual IDE Hard Drive)

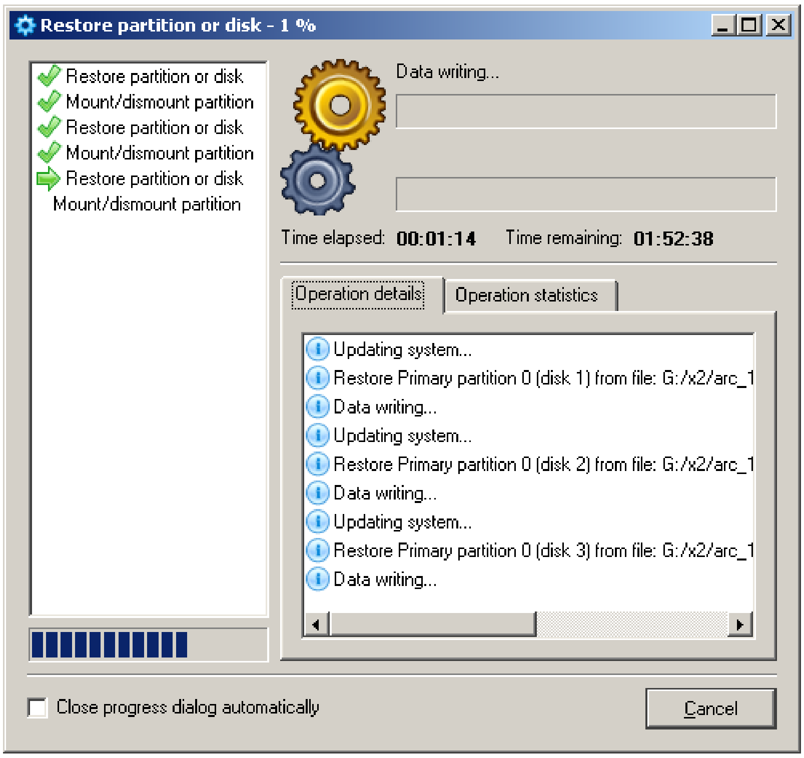
(E:)

Archive Details

Name: **Logical Disk (*)**
Volume label: [No label]
File system: NTFS
Total size: 3.9 GB
Used space: 377.1 MB
Free space: 3.6 GB

To preview the restore results, click **Next**

12. Repeat steps 7, 8, and 9 for the next two volumes (E: and F:)
13. Drive Backup will restore the data for all three volumes after selecting apply:



✓ Restore partition or disk
✓ Mount/dismount partition
✓ Restore partition or disk
✓ Mount/dismount partition
➔ Restore partition or disk
Mount/dismount partition

Data writing...

Time elapsed: **00:01:14** Time remaining: **01:52:38**

```
Updating system...
Restore Primary partition 0 (disk 1) from file: G:/x2/arc_1
Data writing...
Updating system...
Restore Primary partition 0 (disk 2) from file: G:/x2/arc_1
Data writing...
Updating system...
Restore Primary partition 0 (disk 3) from file: G:/x2/arc_1
Data writing...
```

☐ Close progress dialog automatically

14. Once the data from volumes D:, E:, and F: are restored, it will now be time to copy back over the archive logs and the backup control file. These files should be on the external backup destination that was selected from the "Execute after snapshot is taken" command that was used from the "Run during backup options" from step 3. Once these files are located they should now be copied over to there original locations as follows:

- Copy over the archive logs to the destination that was found before the backup from section "Determining Database File Names". It is okay to overwrite the archive logs that were restored from the Paragon Backup image.
- Copy the backup control file from the external backup destination to its original destination or destinations. This information was also found from the section "Determining Database File Names". Keep in mind that if there were originally multiple copies of the control file, then the backup control file will also have to be copied over all these original places and named accordingly. The control file or control files that were restored from the Paragon Backup image will be replaced from this external backup control file, using the same name(s) as the control file(s) that are from the Paragon Backup image.

For example:

```
copy g:\backup_destination\Controlback.ctl D:\ORACLE\CONTROL01.CTL
copy g:\backup_destination\Controlback.ctl E:\ORACLE\CONTROL02.CTL
copy g:\backup_destination\Controlback.ctl F:\ORACLE\CONTROL03.CTL
```

(Controlback.ctl will be renamed accordingly)

Once the Paragon Backup image and the external backup files are restored, it will now be time to perform an Oracle point-in-time recovery.

15. An database administrator (SYSDBA) should start an Oracle instance (`STARTUP MOUNT`) and invoke the database recovery (RECOVER DATABASE...):

(in SQL script)

```
CONNECT / AS SYSDBA
RECOVER DATABASE UNTIL CANCEL USING BACKUP CONTROLFILE;
```

16. The `RECOVER DATABASE` command will use the restored control file in recovery routine. At the final stage, Oracle will interactively locate and replay undamaged archived logs to the database thus recovering data to a moment in time very close to the disaster moment. The process finishes if archived log files are ended or if the `CANCEL` command is entered.

Example:

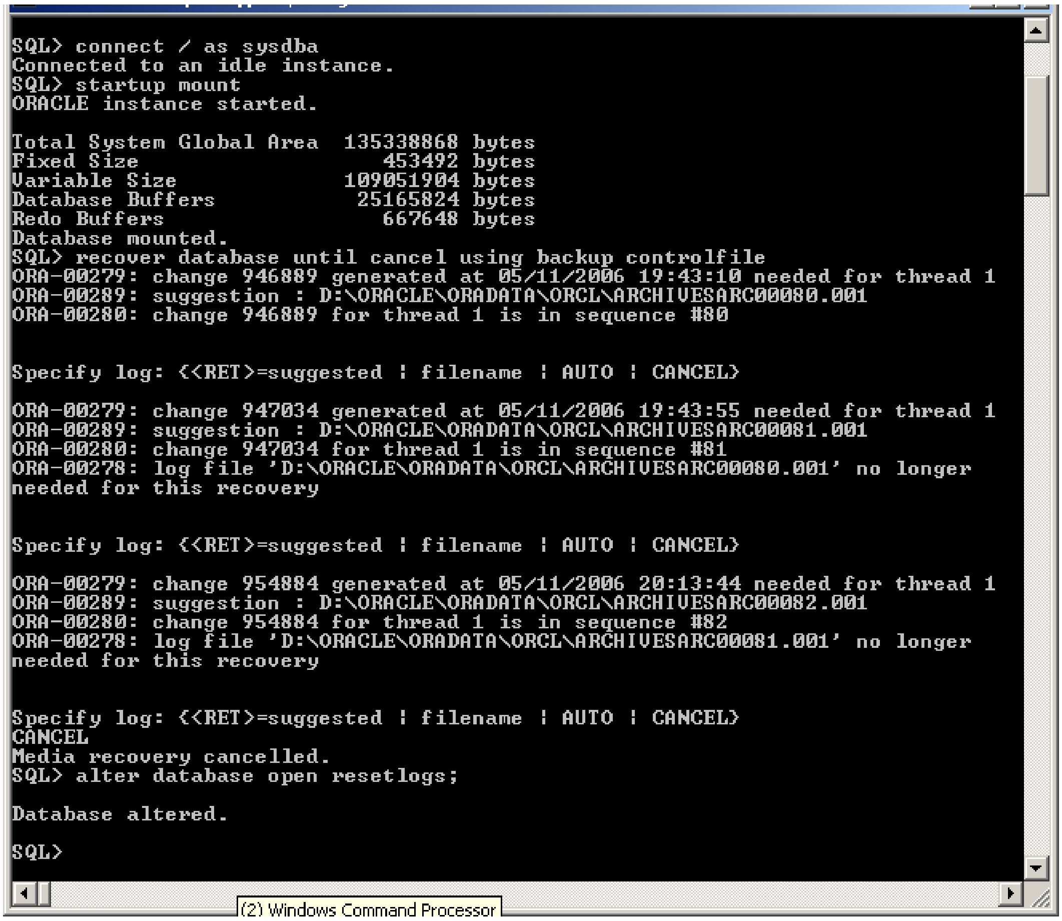
If there were three *archive logs* copied over named in sequence as follows:

```
ARCHIVESARC00079.001
ARCHIVESARC00080.001
ARCHIVESARC00081.001
```

then after Oracle asks for, and applies the ARCHIVESARC00081.001 log then enter `CANCEL`. Just to note, one could also issue the following command:

```
RECOVER AUTOMATIC DATABASE UNTIL CANCEL USING BACKUP CONTROLFILE;
```

and when Oracle asks for the archive log ARCHIVESARC00082.001 then enter `CANCEL`. Once the archive logs are applied, the database can be opened with the `ALTER DATABASE OPEN RESETLOGS` command, this will reset the online redo logs and the sequence of these logs start back at 01. This will now be demonstrated:



```
SQL> connect / as sysdba
Connected to an idle instance.
SQL> startup mount
ORACLE instance started.

Total System Global Area  135338868 bytes
Fixed Size                   453492 bytes
Variable Size             109051904 bytes
Database Buffers           25165824 bytes
Redo Buffers                 667648 bytes
Database mounted.
SQL> recover database until cancel using backup controlfile
ORA-00279: change 946889 generated at 05/11/2006 19:43:10 needed for thread 1
ORA-00289: suggestion : D:\ORACLE\ORADATA\ORCL\ARCHIVESARC00080.001
ORA-00280: change 946889 for thread 1 is in sequence #80


Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

ORA-00279: change 947034 generated at 05/11/2006 19:43:55 needed for thread 1
ORA-00289: suggestion : D:\ORACLE\ORADATA\ORCL\ARCHIVESARC00081.001
ORA-00280: change 947034 for thread 1 is in sequence #81
ORA-00278: log file 'D:\ORACLE\ORADATA\ORCL\ARCHIVESARC00080.001' no longer
needed for this recovery


Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

ORA-00279: change 954884 generated at 05/11/2006 20:13:44 needed for thread 1
ORA-00289: suggestion : D:\ORACLE\ORADATA\ORCL\ARCHIVESARC00082.001
ORA-00280: change 954884 for thread 1 is in sequence #82
ORA-00278: log file 'D:\ORACLE\ORADATA\ORCL\ARCHIVESARC00081.001' no longer
needed for this recovery


Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
CANCEL
Media recovery cancelled.
SQL> alter database open resetlogs;

Database altered.

SQL>
```

The database is now open. The results of the recovery can be found by viewing the database's alert file (alert_xxxx.log). The default name and location for this file is:

`<drive:>\oracle\admin\<dbname>\bdump\alert_<dbname>.log`

Note: The above example was for a point-in-time recovery in case of a disaster. It should be mentioned that if just a datafile is restored from the Paragon Backup image using the *Volume Explorer* and all other Oracle files are still intact and not needing restoration than just the datafile can be restored by:

- taking the tablespace offline with the `ALTER TABLESPACE <tablespace> OFFLINE` command
- restoring the corrupted datafile from the *Volume Explorer*,
- then issuing an Oracle `RECOVER AUTOMATIC TABLESPACE <tablespace>` command
- putting the tablespace back online with the `ALTER TABLESPACE <tablespace> ONLINE` command.

A detailed description of restoring single files using the Paragon Volume Explorer can be found in the Drive Backup 8 Enterprise Server Edition Help.

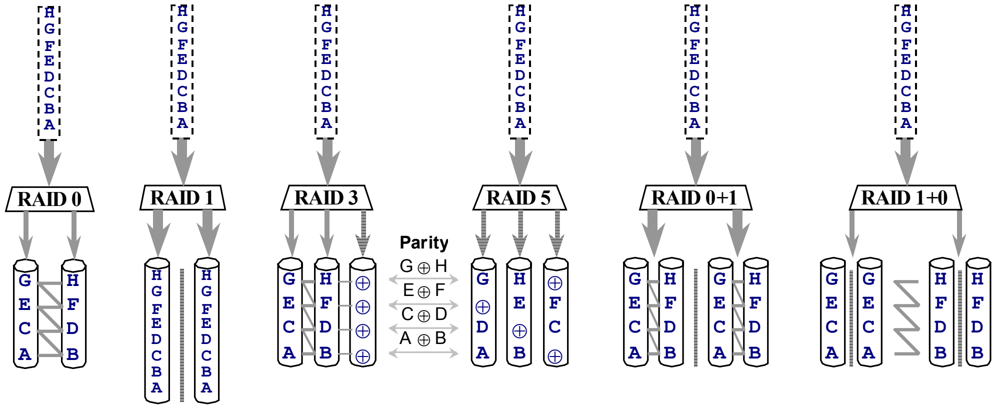
4 Appendix

4.1 RAID Levels

Use of RAIDs is the most effective way to increase throughput and provide a hardware-level fault tolerance of a disk subsystem. Striped RAID sets provide highest read/write performance that is found nearly a multiple of a single disk performance.

RAID level	Brief description	Amount of disks	Size factor	Performance factor	Fault tolerance
0	Striped Disk Array	N ≥ 2	× N	× N	No
1	Mirrored Disk Array	M ≥ 2	× 1	× 1	Yes
0+1	Mirroring of striped segments (RAID 1 over RAID 0)	M•N ≥ 4 M ≥ 2, N ≥ 2	× N	× N	Yes
3	Striped Disk Array	N+1 ≥ 3	× N	× N (for reads)	Yes

	with Isolated Parity			× 1 (for writes)	
5	Striped Disk Array with Distributed Parity	N+1 ≥ 3	× N	× N (for reads) × 1 (for writes)	Yes
10	Striping of mirrored segments (RAID 0 over RAID 1)	N•M ≥ 4 N ≥ 2, M ≥ 2	× N	× N	Yes



A more detailed characteristic of various RAID levels can be found in the Internet.

4.2 Script Examples for Run During Backup Options

The following gives an example of how to place the tablespaces in backup mode and also to copy the archive logs and control file to an external backup destination. These examples refer to the section 3.5.2 Example 2: Advanced Routine of a Database Backup and Restore

Note: These batch files are connecting to the database using the empty instance, which is issued by the following command:

```
sqlplus / as sysdba
```

It only works if the user is a member of the *ORA_DBA group* and Oracle is operating in the *operating system authentication mode*. For *Oracle authentication mode* the user will need to use a valid sysdba account. In this case, the command should have the following format:

```
sqlplus sys/<password> as sysdba
```

1. Build a batch file that will call a PL/SQL script to be ran at the beginning of the backup process before taking a snapshot. Let's name that file "before_snap.bat".

Contents of the "before_snap.bat" file:

```
sqlplus " / as sysdba" @before_snap.sql
del c:\before_snap2.sql
```

(file dynamically built from the before_snap.sql)

2. Build a PL/SQL script that will dynamically find the tablespaces to put in backup mode and then suspend the database. This is the "before_snap.sql" file that is being called from the above batch file. This file will then dynamically build a file called "c:\before_snap2.sql" with the necessary commands, which is then deleted at the end of the routine.

Contents of "before_snap.sql":

```
set serveroutput on
set trimspool on
set line 500
set head off
set feed off
set verify off

spool c:\before_snap2.sql

declare
   logmode varchar2(30);
begin

   select log_mode
   into   logmode
   from   sys.v_$database;

-- Check if the database is in archive mode.
if logmode <> 'ARCHIVELOG' then
   raise_application_error(-20000, 'ERROR: database must be in archivelog mode!');
return;
end if;

-- all tablespaces begin backup
for c1 in (select tablespace_name ts from sys.dba_tablespaces where contents !='TEMPORARY')
loop
   dbms_output.put_line('alter tablespace '||c1.ts||' begin backup;');
end loop;
dbms_output.put_line('alter system suspend;');
end;
/

spool off
set head on
set feed on
set serveroutput off

@c:\before_snap2.sql
exit
```

3. Build a batch file that will then call a PL/SQL script to be ran at the after taking a snapshot.

Contents of the "after_snap.bat" file:

```
sqlplus " / as sysdba" @after_snap.sql "d:\arc_ctl_backup_dest"
del c:\after_snap2.sql
```

(file dynamically built from the after_snap.sql)

Note: the external backup destination directory "d:\arc_ctl_backup_dest" should be changed to an appropriate external destination for backing up the archive logs and control file.

4. Build a PL/SQL script that will resume the database and then dynamically find the tablespaces to turn off backup mode, along with copying the control file and archive logs to the destination that was edited from the "after_snap.bat" file. The "after_snap.sql" file is being called from the above batch file. This file will build dynamically a file called "c:\after_snap2.sql", which is then deleted at the end of the routine.

Contents of the "after_snap.sql" file:

```
    alter system resume;
    set serveroutput on
    set trimspool on
    set line 500
    set head off
    set feed off
    set verify off

    spool c:\after_snap2.sql

 declare
   archivelog_source varchar2(150);
   dbname  varchar2(30);
begin

     select name
     into   dbname
     from   sys.v_$database;

     select value
     into   archivelog_source
     from   v$parameter where name = 'log_archive_dest';

for c1 in (select tablespace_name ts from sys.dba_tablespaces where contents !='TEMPORARY')
    loop
       dbms_output.put_line('alter tablespace '||c1.ts||' end backup;');
     end loop;

   dbms_output.put_line('alter system archive log current;');
   dbms_output.put_line('alter database backup controlfile to trace;');
   dbms_output.put_line('alter database backup controlfile to
                             ''&1\'||dbname||'.'||sysdate||''' reuse;');
   dbms_output.put_line('host copy '||archivelog_source||'\*.* &1');
end;
/
spool off
set head on
set feed on
set serveroutput off

@c:\after_snap2.sql &1
exit
```

After completing this operation there should now be a control file and the archive logs located in the chosen external backup destination that was edited in the "after_snap.bat" file. In the case of this example the destination would be "d:\arc_ctl_backup_dest". These files along with the Paragon backup image will constitute the full backup of the database.