

END-TO-END STORAGE MANAGEMENT FOR ORACLE DATABASES

AVAILABILITY, MANAGEABILITY, AND PERFORMANCE FOR LARGE ORACLE DATABASES

Fred van den Bosch, VERITAS Software Corporation

ABSTRACT

Modern Storage Systems consist of many components, including File Systems, Volume Managers, Device Drivers, Backup/Recovery tools, and Hardware Subsystems that include complex software components to implement RAID and other advanced Storage Management functions. The components are usually provided by a variety of different vendors, and each component typically comes with its own management interface, that may or may not expose its functional behavior to the System Administrator. In addition, System Administrators can not readily obtain insight in the I/O access patterns of applications running on the systems under their control. These factors make the management of Storage Systems for optimal availability, cost and performance a sheer impossible task for ordinary (or even extra-ordinary) humans. This paper discusses products and technologies that make it possible to manage storage for Oracle databases for maximum availability and performance, while limiting the effort and knowledge required by System Administrators.

INTRODUCTION

THE SHIFTING LANDSCAPE – INFORMATION CENTRIC COMPUTING

Since the early days of computing, storage devices have been viewed as peripherals to computer systems, in the same manner as workstations were considered peripherals before the evolution of the Local Area Network and client-server computing.

Until recently, interconnection technology has restricted the distance over which storage devices can be attached, and the number of hosts and devices than can be connected. This has caused this “peripheral view” of storage to persist, despite the fact that companies now invest more in storage resources than in computing resources.

With the emergence of Fibre Channel and other new storage interconnection technologies, we are now at the beginning of an era, in which storage systems will become central elements of Information Systems. We refer to this as “Storage Centric” or “Information Centric” computing.

THE IMPORTANCE OF STORAGE MANAGEMENT

Storage Management is comprised of all activities required for configuring and maintaining the storage resources used for electronically storing information. The most important goals of Storage Management are:

- to protect data against loss that may result from hardware failures, human errors, or software errors
- to minimize loss of access to data, for example as a result of the corrective or preventive maintenance activities
- to optimize the speed with which data can be accessed

The ability to keep vast amounts of data accessible on a continuous basis, and with optimal performance, is increasingly critical to companies for meeting their business objectives. In conjunction with the shift towards Information Centric computing, this causes Storage Management to rapidly evolve from a little-known activity in the bowels of IS organizations, to a critical element of the IS strategy of companies around the world.

With much of the business critical data managed by Oracle, there is a need and opportunity for Storage Management solutions that are optimized for Oracle databases.

STORAGE MANAGEMENT OBJECTIVES

Before discussing Storage Management technologies and products, we will review the objectives IS organizations have in performing Storage Management tasks:

- Data availability

As stated earlier, the most important goal of Storage Management is protect data against loss that may result from hardware failures, human errors, or software errors, and to minimize loss of access to data. Both planned and unplanned unavailability need to be minimized, which translates in the requirement to minimize backup windows and recovery times, and to perform backup as well as other storage management tasks (such as re-configuration) on-line.

- Minimize the impact of data protection measures on application availability and performance.

In addition to avoiding that data and applications have to be taken off-line for performing Storage Management tasks, this objective requires that Application Servers should not have to perform I/O or compute intensive tasks for this.

- Maximize I/O performance and optimize the speed with which data can be accessed

This implies the need to optimize database I/O policies for a given application and storage configuration, as well as the need to configure storage in a manner that is optimal for the I/O access patterns generated by the application.

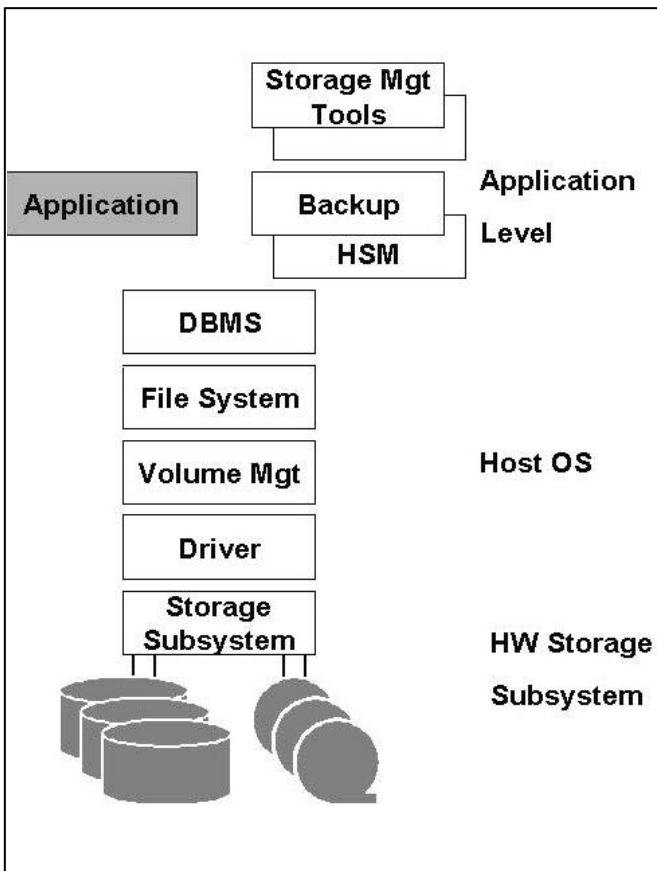
- Limit management costs

Some analysts estimate that the cost of managing storage is as much as 8 times higher than the costs of the storage devices them selves. Limiting Storage Management costs, despite the rapid increase in the volume of information that is stored electronically and the need to keep more of this information available on a continuous basis, is therefore an important objective.

THE STORAGE MANAGEMENT CHALLENGE

The Storage System of a (networked) Computing Environment is comprised of a variety of different types of components, as shown in figure 1.

- Hardware Subsystems, such as RAID controllers, RAID Subsystems, or File Servers



- Operating System Components, such as Device Drivers, Logical Volume Managers, and File Systems
- Applications level tools, such as Backup / Recovery Managers, Hierarchical Storage Managers, and Storage Resource Management tools.

The following factors make the management of Storage Configurations for optimal data availability, performance, and cost a highly challenging undertaking:

- Almost invariably, components are supplied by a number of different vendors.
- Each component typically comes with it's own management interface, and may or may not expose it's (frequently complex) functional behavior to the System Administrator.
- System Administrators can not readily obtain insight in the I/O access patterns of applications running on the systems under their control.
- The volume of information that is stored electronically is doubling each year
- The lack of integration between products from different vendors results in missed opportunities for creation of solutions that support IS organizations in meeting their Storage Management objectives.

Add to this that most enterprises deploy heterogeneous system

Figure 1: Overview of Storage Management Components

configurations, and it will be clear that performing Storage Management reliably and affordably has become a significant challenge for many IS organizations.

END-TO-END STORAGE MANAGEMENT

To support IS organizations in addressing these challenges, VERITAS has embarked on a number of initiatives. These include the development of new products and technologies as well as the cooperation with other vendors to integrate Storage Management products through common API's, management tools, and interfaces. The resulting "End-To-End Storage Management" solutions aim to increase data availability, improve performance, simplify and automate management tasks, and reduce management costs. An overview of initiatives that are important for Oracle database environments is given below:

- File System as a preferred platform for Oracle databases.

File Systems are an attractive storage solution for databases from a point of view of manageability. However, the UNIX file system has traditionally been considered too slow and unreliable to host mission critical databases. The VERITAS VxFS file system offers the manageability advantages of file systems, without compromising reliability or performance. VERITAS has worked with Oracle to perform extensive tests of VxFS with Oracle databases and ensure its correct behavior.

- New "continuous data availability" solutions for Oracle databases by integration of file system, backup, and Hardware Storage Subsystems.

- Reduce cost of administration by sharing information between Storage Management products.

To automate and simplify the manner in which products work together, API's are being defined through which products from different vendors can share relevant information. This makes it possible for these products to transparently optimize their behaviors, and simplify Storage Management tasks by hiding complexities from the System Administrator.

- Oracle specific, cross-product intelligent management tools to simplify and automate Storage Management tasks
- Product Suites for Oracle

Effective Storage Management requires knowledge of the characteristics of the application environments for which storage is to be configured and managed. For example, storage for an Oracle database will need to be configured and managed in a different manner than the storage for a multi-media server. Storage Management can be simplified significantly by building domain specific knowledge into products, and providing Storage Management Product Suites that consist of multiple such products, integrated together to offer complete Storage Management solutions for specific application domains.

The goal is to support these capabilities on a variety of configurations, regular UP and SMP systems, parallel cluster configurations, and configurations deploying intelligent Storage Subsystems.

Creating these "End-To-End Storage Management" solutions requires vendors of Hardware Storage Systems, Database Management Systems, and Storage Management software products to cooperate in the definition and implementation of API's between their products, and the validation and testing of products and solutions based on these API's. Oracle and VERITAS share the vision that we superior, easy-to-manage solutions can be provided by cooperating in this manner, and are working together in the definition and implementation of such solutions. The remainder of this paper gives a more detailed overview of the initiatives mentioned above.

VXFS FILE SYSTEM - A PLATFORM FOR ORACLE DATABASES

File Systems are an attractive storage solution for databases from a point of view of manageability. However, traditionally many databases on UNX have been implemented using the raw device interface, because file systems were considered too slow and unreliable to host mission critical databases. The main reasons for this are the long recovery times after system failures, use of buffered I/O, and the fact that only a single process can write to a file at any given time.

- The long recovery times result from the fact that the traditional UFS file system needs to perform a full file system check (a.k.a. "fsck") that traverses all of the file system meta data structures, to ensure their correctness after a system failure.
- Buffered I/O often speeds up repeated reads on the same data blocks. This is useful for smaller files that are accessed frequently. However, since databases maintain their own cache, buffering the data in the kernel actually increases CPU

overhead for data copying between Oracle cache and kernel cache. In addition, it reduces the memory space available for the database cache.

- Having only one writer have the exclusive right to update a file forces other read and write operations on the same file to wait and makes it impossible for the database to perform concurrent asynchronous write operations.

The VxFS File System uses an internal log to ensure that all of its meta data changes are atomic. As a result, it does not need to perform a full fsck, and recovers very quickly after system failures. To address the performance drawbacks traditionally associated with file systems, VxFS offers a “Quick I/O” interface. By providing unbuffered and asynchronous I/O, this interface offers characteristics similar to the raw device, and a comparable degree of read/write parallelism and CPU overhead.

As Computer Systems evolve towards support of 64-bit address spaces, some Operating Systems do not yet make it possible for Oracle to take advantage of system memory configurations larger than 4GB. To make it possible to achieve better performance on such large systems, VxFS also supports a “Cached Quick I/O” capability. In this mode read operations are served through the VxFS cache. This will often reduce the number of physical I/O operations and thus improves read performance. For write operations, Cached Quick I/O functions as the “standard” Quick I/O facility to guarantee data integrity. For on-line transaction processing on Solaris 2.5 and 2.6, Cached Quick I/O achieves better than raw performance in database throughput on large memory configurations. Cached Quick I/O also helps sequential table scans due to the read-ahead algorithm used in the VERITAS File System, resulting in reduced query response times.

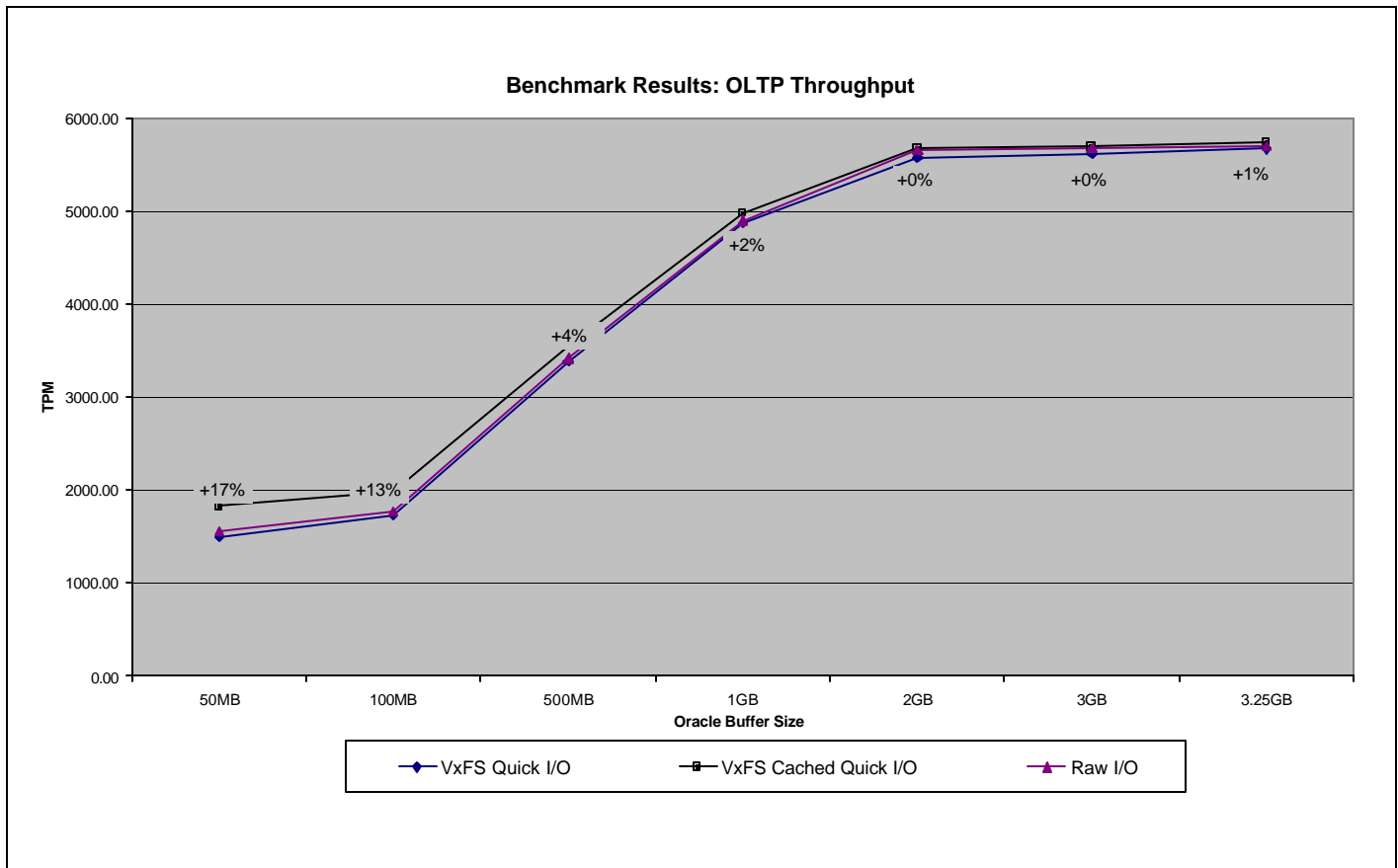


Figure 2: OLTP throughput using raw device, ufs, and vxfs

VERITAS has worked closely with Oracle to execute validation and performance tests of Oracle databases on VxFS. This has helped VxFS to evolve as a storage solution for Oracle databases that offers significant manageability advantages without compromising reliability, and with performance that is equivalent or –up to 15-20%– higher than when using the raw device interface.

Figure 2 gives an overview of OLTP transaction rates for Oracle 8.0.3 on Solaris 2.6, and VxFS 3.3 with Quick I/O and Cached Quick I/O, as compared to raw device. Tests were run on a Sun Microsystems Ultra Enterprise 10000 domain with eight processors, 6GB of memory, and 37 disk drives (4 per controller). A 50GB TCP-C database was used. All the VxFS file system tunable parameters, including the `mkfs` parameters, were default.

With, in addition to this, the Continuous Availability features that are discussed in the next section, VxFS is the storage solution for Oracle databases. Table 1, gives an overview of the advantages of VxFS over the use of Raw Partitions or UFS.

Issue	UFS (Unix File System)		Oracle Raw Partitions		VERITAS VxFS File System	
Data Integrity	<u>Short</u> window for data corruption.	-	<u>No</u> Window for data corruption	+	<u>No</u> window for data corruption.	+
Access efficiency	Requires <u>250,000</u> calls to kernel to read 2GB file.	-	<u>No</u> calls to kernel	+	Requires <u>1 call</u> to kernel to read 2GB file.	+
Caching	<u>Double Cache</u> , wastes memory and CPU resources	-	<u>No</u> Double Caching	+	Only caches Read Operations for >4GB memory configurations	+
Write Performance	<u>Single</u> CPU can write to file at a time.	0	<u>Multiple</u> CPU's can write to file at a time.	+	<u>Multiple</u> CPU's can write to a file at a time.	+
Backup	<u>Easy</u>	+	<u>Difficult</u> , harder to manage and restore	-	<u>Easy</u>	+
Manageability	<u>Easy</u> – Grow FS after unmounting	+	<u>Difficult</u> – must dump and restore.	-	<u>Easy</u> – Grow or Shrink FS while mounted.	+
Protection against user error	<u>Good</u>	+	<u>Poor</u> – can format partition actively used by Oracle.	-	<u>Good</u>	+

Table 1: Comparison of UFS, Raw Partitions, and VxFS as storage solution for Oracle

CONTINUOUS DATA AVAILABILITY

OBJECTIVES

As stated earlier, the most important goals of Storage Management are to maximize data availability, in the face of hardware failures, human errors and software errors, and to minimize the impact of data protection measures on application availability and performance. This requires:

- On-line operation
It should be possible to perform any storage management task without the need to take applications off-line, and without the need to have Application Servers perform I/O or compute intensive tasks.
- Highly efficient methods for creating copies of data that can be used as backup or for other (e.g. Decision Support) applications.
- The ability to very quickly (i.e. with minimal data movement) restore a consistent image of the application data in case of data loss or corruption.

The VERITAS Volume Manager and File System have long provided the possibility to perform all management operations on-line. This included the ability to grow or shrink logical volumes and file systems, to reorganize file systems, and to improve performance by moving data or changing storage layouts.

More recently, innovation at VERITAS and other Storage Solution providers has focussed on making significant improvements in the areas of backup & recovery and on-line Decision Support. The emphasis of these efforts is on the

development of new technologies as well as on integration between Storage Management products. Specific facilities that are available now, or will be available in the foreseeable future are:

- Online backup and Decision Support
- Incremental backup (also online)
- Off-host backup and Decision Support
- Storage Checkpoint / Rollback for fast recovery

SNAPSHOT TECHNIQUES

Several techniques have been developed that make it possible to provide these capabilities. They have in common that they make it possible to create a “snapshot” image of a file system or database. A snapshot is a consistent image of the data as it was at a specific point in time (the moment the snapshot was created). Snapshots can be accessed for read, and in some cases write access, while applications are running and modifying the original “live” data. This allows them to be used for on-line backup, Decision Support, and other applications. The most common snapshot techniques are:

MIRROR BREAK OFF

The simplest, but most expensive, snapshot technology uses mirroring (RAID-1). A snapshot image is created by “breaking off” a mirror. This requires an amount of disk space equal to the total size of the file system or database. This technique is supported by some advanced Hardware Storage Subsystems.

COPY ON WRITE

In this case a stable image is maintained by saving the pre-images of changed blocks in a separate storage area. The first time a write changes a particular data block, the old data is first read and copied to the snapshot area before the new data is written. A subsequent read request for that block in the Snapshot will be satisfied by reading the data from the snapshot data area, rather than from the “live” file system or database. Subsequent writes to the block on the live database do not result in additional copies to the checkpoint, since the old data only needs to be saved once. Read requests for blocks in the snapshot database that have not changed, are satisfied from the “live” database. The advantage of copy-on-write snapshots is that they minimize the number of disk accesses and the storage space required for maintaining the snapshot, and work for database files as well as flat files. The challenge is to implement them with minimal performance degradation.

Copy-on-write snapshots can be implemented in Hardware Storage Subsystems, file systems or driver level products. The VxFS File System provides a copy-on-write snapshot (referred to as “Storage Checkpoint”) that can also be combined with incremental backup solutions (see below). Through the use of coalesced write operations, logging and other optimization techniques, the performance overhead associated with maintaining Storage Checkpoints is very small. Multiple Storage Checkpoints can exist concurrently, representing images of the file system or database at different points in time. Creating a Storage Checkpoint is a fast operation, which is typically completed in a few seconds. The file system and database must be in a consistent state while the Checkpoint is created. In the case of Oracle, this can be accomplished by using the “Archive Log Mode”. In this mode, a consistent image of the database is available of which a snapshot can be taken without any database down time.

We will now explain the use of the Snapshot technology for improving data availability.

ONLINE BACKUP AND DECISION SUPPORT

The techniques described above, make it possible to create a consistent, instant-in-time view of a database, and can therefore be used to perform a backup, while the application is running and updating the “live” database. As described above, creating a consistent snapshot of an Oracle database can be accomplished without Database down time by using Oracle’s Archive Log Mode.. Standard backup utilities, Decision Support, and other applications can now process the snapshot. In this manner it is possible to perform a cold database backup while the database is actually online (“hot“)!

To support on-line backup of Oracle databases, VERITAS NetBackup has been integrated with the snapshot facilities offered by EMC in its Symmetrix Storage Subsystem, and by the VxFS file system. It transparently handles the operations required to establish and discard the snapshots. It also supports the Oracle RMAN on-line backup capabilities, giving users the opportunity to select the most appropriate solution for their application environment.

INCREMENTAL BACKUP

Although online backup eliminates the time a database has to be taken offline for backup, it still requires all of the data in the database or table space(s) that are backed up to be copied. This will often have a significant impact on application performance during the time the backup takes place. To minimize this impact, it is desirable to perform an “incremental backup”, copying only the data that has changed since the previous backup was made. This will often reduce the amount of data to be copied by one to two orders of magnitude. When incremental backup is done on-line, the impact of backup on application availability and performance is dramatically reduced.

Below we will discuss the different types of incremental backup solutions that are available or will be available in the foreseeable future, and their suitability for various application environments.

- File level

File systems maintain a timestamp for each file, which indicates when the file was last changed. This makes file level incremental backup relatively simple to implement, and most backup products, including NetBackup, support it.

File level incremental backup works well for backing up “traditional” file server environments. In these environments, the average file size is small and updates to a file are accomplished by truncating the file and re-writing it in its entirety. In database environments, the average file size is large, and the unit of change is a fixed size block rather than an entire file. This makes this technique not suitable for those environments.

- Byte level

Statistical “differencing” algorithms make it possible to determine small changes in files, such as the insertion of a small number of characters. This technique, possibly in combination with compression, can reduce the amount of data to be backed up significantly. However, it requires the reading and processing of all changed files, and does not scale well in large (or even medium) database environments.

- Block level

Block level incremental backup is targeted at the typical database environment, where data is written at the block level. There are fundamentally two approaches for block level incremental backup:

- Differencing engine

Like the byte level differencing described above, this technique also reads all data. However, it can use a simple method to determine which blocks have changed. Since databases use a small number of large files, it requires almost the entire database to be read. As a result it does not scale for larger databases, and does not result in a significant reduction in backup times. The incremental backup facility offered by Oracle’s Recovery Manager (RMAN), uses this technique.

- Run-time differencing

This technique keeps track of which blocks have changed at run-time, as write operations are performed. At backup time it is only necessary to read the changed blocks.

The VxFS File System implements this technique, incurring a very small runtime overhead independent of the size of the database. This solution results in a reduction in backup time that is proportional to the percentage of data blocks that has changed, and is scalable to very large databases. The block size can be set to match the block size used by Oracle.

Table 2 shows the results of a comparative test between Oracle RMAN incremental backup and VxFS Block Level Incremental backup, that was performed by Jeffrey Carter at Boeing. It clearly shows the significant reduction in backup times that can be achieved with block-level run-time differencing solutions as provided by VxFS. By using disk as the backup medium for the usually small amount of data, backup times can be further reduced. Also important to note is the significant reduction in Restore time that is achieved.

The tests were performed on a SUN UE3000, using Oracle 8.0.4 and VxFS 3.3. The total database file size allocated was 5.7 GB, with 3.4 GB of actual data. An 8 table schema and TPC-D data was used for the test. The approximate table sizes in records are as follows:

Customer 300,000
 Lineitem 12,000,000
 Nation 25
 Orders 3,000,000
 Parts 400,000
 Partsupp 1,600,000
 Region 5
 Supplier 20,000

The test consists of performing one full and two incremental backups. Prior to the first incremental backup, 12,000 records in the CUSTOMER table are updated. Prior to the second incremental backup, 80,000 records are updated in the PARTS table. Elapsed times for performing the backups are logged. A restore and recovery is executed on both databases after the second incremental backup. Both disk and tape backups were performed for this comparison. Veritas NetBackup is used for writing to tape, and only RMAN for writing to disk.

Table 2: Backup and Restore Elapsed Times (MIN:SEC)

	Full	Incremental_1	Incremental_2	Restore
VxFS BLI (Tape)	138:24	5:01	5:02	16:11
RMAN (Tape)	103:33	15:10	15:24	34:09
VxFS BLI (Disk)	31:01	1:42	1:59	1:04
RMAN (Disk)	25:00	12:17	11:59	4:48

Figure 3 shows the results of a series of tests executed by VERITAS that illustrate how VxFS Block Level Incremental Backup times correlate with the percentage of data blocks that has changed.

- Track level

Track level incremental backup is similar to block level incremental backup in the sense that changes are tracked at runtime. However, the unit of data is larger than in case of block level incremental backup, which means that more data has to be backed up. Some Intelligent Storage Subsystems are expected to implement this technique.

VERITAS NetBackup will support a variety of incremental backup solutions for Oracle databases, transparently handling the complexities of performing the database operations and snapshot management required for this. At this time, it supports the Oracle RMAN incremental backup facility as well as a VxFS Storage Checkpoint based incremental backup solution. The company is working with Hardware Storage Subsystem vendors to support solutions offered by those vendors, as these come available.

Although incremental backup can significantly reduce backup time and overhead, a recent full image of a database is still required to be able to restore a database in the shortest possible time in case of a major disaster. By making it possible to integrate one or more incremental backups with an existing “full” image of a file system, it is possible to always have a recent “full” image of the data available, without ever making a full backup! Creation of such a “synthetic” full backup can be done on a secondary server, without impacting application performance. VERITAS will offer a “synthetic” full backup capability in a future release of NetBackup.

OFF-HOST BACKUP AND DECISION SUPPORT

On-line incremental backup results in dramatic reductions in the impact of backup on application availability and performance. However, the Application Server still needs to copy the data from disk to the backup medium or network

connection. Off-host backup techniques make it possible for another system to perform the backup. The Application Server only needs to be involved in quiescing the file system or database before the backup can start.

Off-host backup can be accomplished in a number of manners, as summarized below:

- In a shared-storage configuration, through third-mirror break-off on the Application Server, and subsequent backup of the third mirror by the Backup Server. As mentioned earlier, this requires an amount of disk space equal to the total size of the file system or database to be backed up.

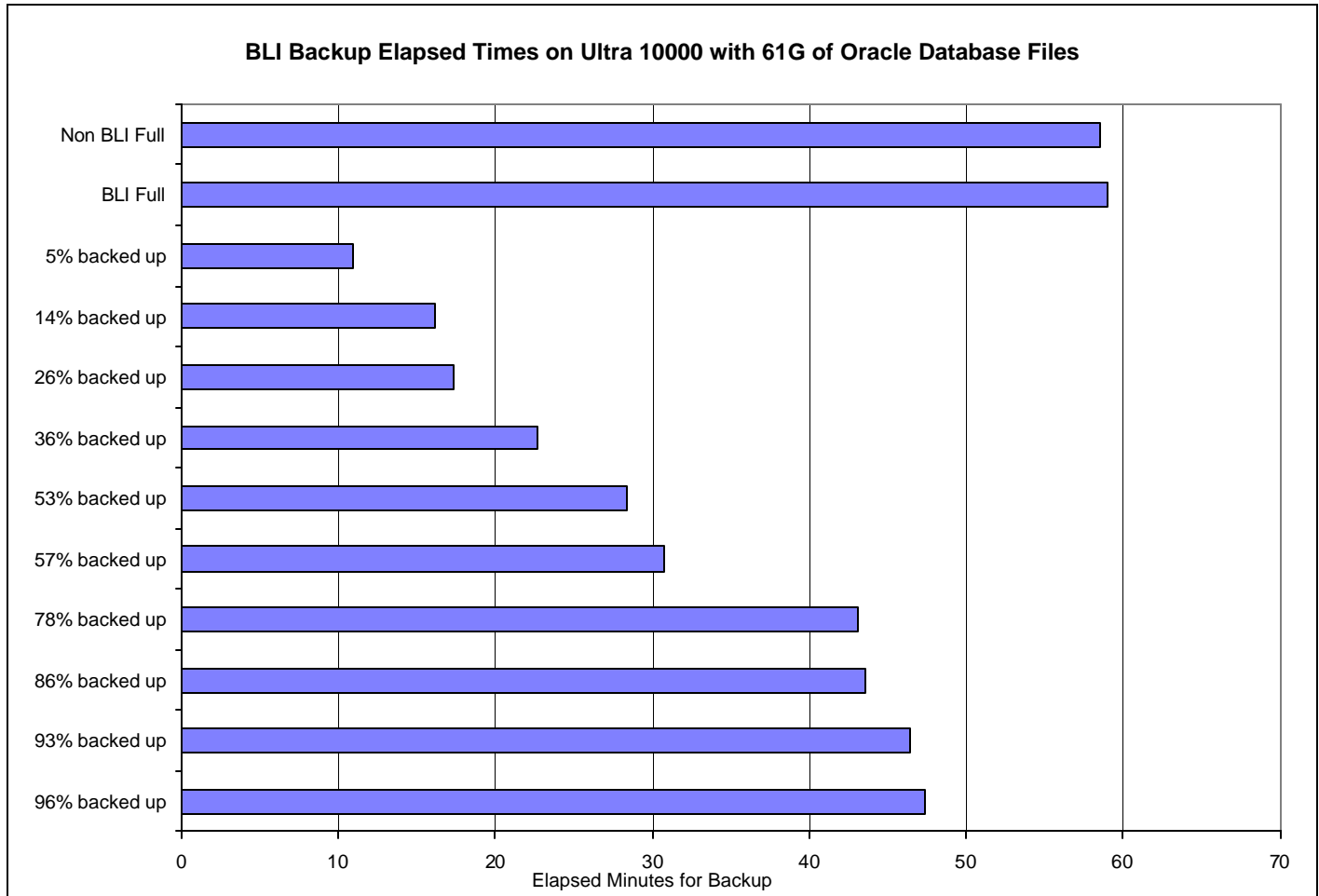


Figure 3: Elapsed times for backup using VxFS based Incremental Backup

- Through the use of Cluster Volume Manager and Cluster File System technology, it is possible to have concurrent access to the same data from multiple servers that share access to the same physical storage. If the file system provides a snapshot capability, this makes it possible for the backup process to run on one server while the application runs on one or more other servers.
- Through the use of off-host data movement capabilities that can be provided by Hardware Storage Subsystems, dedicated “data movers” or possibly fibre channel switches. These systems will be able to copy data directly between fixed disk and removable media, under control of a backup application running on the Application Server.
- Finally, it is possible to accomplish off-host backup by use of data replication over a local or wide area network, and backup of the replica on the remote system. Data replication can be accomplished by Hardware Storage Subsystems as well as software products.

VERITAS will support a variety of off-host backup solutions. At present, the VERITAS Volume Manager supports the EMC Symmetrix “TimeFinder” capability for establishing break-off mirrors on a secondary server. NetBackup supports backup of Oracle databases using the TimeFinder facility.

VERITAS also offers software replication products (the Storage Replicator for Volume Manager and File System), to create copies of data that can be backed up on a secondary host.

VERITAS is actively working with several suppliers of Intelligent Storage Subsystems to support peer-to-peer copy facilities that will be offered by these vendors. In the future, a Clustered version of VxFS will offer yet another solution for performing off-host backup, using commodity hardware systems.

STORAGE CHECKPOINT / ROLLBACK

With the techniques described until now, it is possible to almost completely “close the backup window”. By making the backup process less intrusive, it is possible to take backups more frequently. This will also have a positive impact on recovery time, since the time required to restore the data lost since the backup (through log replay) will be shorter. However, the improvement in recovery time is less dramatic than the improvement in backup time and system overhead.

It is possible to reduce recovery times for data loss that is the result of software or human error (“logical data corruption”), by using copy-on-write snapshot techniques as discussed earlier. Copy-on-write snapshots maintain a complete and consistent image of the data as it was at the time the snapshot was created, and can therefore provide the ability to “rollback” changes. This makes it possible to recover from logical data corruption without the need to restore from a backup!

The benefit of a file system based copy-on-write snapshot facility, as provided by the VERITAS file system, is that roll back can happen for the entire database or for individual files (typically table spaces in the database).

By combining the use of copy-on-write snapshot solution with RAID to protect against physical device failures, lengthy recovery from tape backup is only needed in extremely rare circumstances.

IMPROVED PERFORMANCE AND EASE-OF-ADMINISTRATION

The standard UNIX or NT file system interfaces limit the extent to which databases can minimize the risk of data corruption through human error, and achieve optimal performance.

VERITAS is working with Oracle and Hardware Subsystem vendors to define and implement API’s that aim to make it possible for Oracle to achieve the best possible performance and availability, while at the same time simplifying administration. Below, we will briefly discuss some areas that could be covered in such interfaces:

- I/O policy optimization

In present-day computer system configurations, Oracle does not have access to information about the geometry of the storage configuration underlying the database, and hence can not adapt its I/O policies for optimal performance.

API’s between Hardware Storage Subsystems, Volume Manager, File System, and Oracle can make it possible to pass information about optimal I/O size and alignment from the Hardware Storage Subsystem or (for SW managed arrays) Volume Manager to Oracle. This will allow Oracle to adapt its I/O policy accordingly, for example ensuring that writes to a RAID-5 storage configuration would happen as full-stripe writes.

- Caching and logging optimization

In the same manner as Oracle has no access to information about the geometry of the storage configuration, Hardware Storage Subsystems can not obtain information about the I/O access patterns to be expected. If available, such information could be used to optimize use of the subsystem resources.

API’s can be defined that make it possible for Oracle to provide File System, Volume Manager, and Hardware subsystems with information about expected I/O access patterns, allowing these subsystems to optimize caching and logging policies. Examples of information to be passed through such API’s are sequential I/O hints and identification of data that will no longer be accessed (and can hence be discarded from logs and/or caches).

- Optimized I/O interfaces

Oracle's preferred I/O policies can not be mapped onto standard UNIX or NT File System API's in an optimal manner. By implementing Oracle specific facilities, it is possible to increase performance by reducing the number of system calls and context switches required, and increasing parallelism.

- Improved manageability, and reduced chance of data corruption and human error

Existing File System interfaces make it necessary for the administrator to "manually" perform certain file and table space management tasks, and expose databases to corruption as a result of human error. This is caused by the fact that certain file and table space management functions can not be performed by Oracle directly. Also, it is impossible to protect critical files, that should be accessed and managed by Oracle exclusively, against undesired access by standard file system utilities. Through proper API's, Oracle could perform these functions completely under its control. Access by standard file system utilities to certain critical files could be prohibited. This will significantly improve manageability, and reduce the chance of data corruption as a result of human error.

- Faster recovery

Recovery of mirrored storage configurations following a failure of the system managing the storage configuration, requires that data that may not have been written to all mirrors, be read from one and written to the other mirrors. By making use of Oracle's change logs, it is possible to minimize the amount of data that has to be copied, and therefore speed up recovery and minimize the impact of the recovery process on application performance.

CROSS-PRODUCT INTELLIGENT MANAGEMENT TOOLS

To achieve optimal performance and availability for database applications, a System Administrator has to perform the following tasks:

- Create an initial storage configuration that is as optimal as possible, given the information available at the time the database is created.
- Continuously monitor and analyze performance and suitability of the storage configuration for the application environment, and change the storage configuration to optimize performance, and adapt to changing workloads or performance goals.
- Perform planned (pro-active) and unplanned (reactive) management tasks with minimal disruption to system and data availability.

These tasks are collectively referred to as "Storage Resource Management" or "SRM". There are several classes of SRM products to support the Administrator::

- Monitoring and Event Management products, such as the VERITAS Storage Manager, that automate the re-active and pro-active management tasks.
- Configuration Assistants that support System Administrators in setting up storage configurations to match expected application characteristics and performance and availability objectives.
- Optimization Products, such as the VERITAS Storage Optimizer, that monitor I/O access patterns, as generated by the applications and identify performance bottlenecks and inefficient storage configurations.
- Planning Products that will monitor actual storage related performance against objectives set for the application environment, and support the System Administrator in timely capacity planning.

Products that are in the market today, are still of a generic nature, and limited in scope. It is expected that in the future we will see products that can be configured with knowledge of specific Storage Management products and Databases. There could be knowledge of Oracle specific storage requirements and I/O access patterns, as well as knowledge on the characteristics of specific Hardware Storage Subsystems or other products. By adding product specific knowledge, it will be possible to provide more automation of management tasks and improved ease-of-use for administrators.

PRODUCT SUITES

As discussed earlier in this document, the Storage System consists of a large number of components, supplied by a variety of vendors. We have shown how availability and performance can be enhanced by a combination of new technologies, integration between components from different vendors, and specialization for Oracle.

To further improve manageability, it is beneficial to combine multiple components into Product Suites that provide a complete “End-to-End” storage management solution for Oracle. In addition to combining multiple (Oracle optimized) products into a single package, such Product Suites should be extensively tested and benchmarked against representative Oracle configurations, and documented and installable as a single package.

VERITAS already offers such a Product Suite (under the name “VERITAS Database Edition for Oracle”). It supports Quick I/O, Cached Quick I/O, Block Level Incremental Backup, Checkpoint / Rollback, and other Oracle specific functionality.

In the future, the number of components in such Product Suite will increase to offer more of the capabilities described in this document, and cover a wider range of configurations. Future extensions are expected to include:

- Integration of Hardware Storage Subsystems
- Oracle specific Storage Management interfaces
- Oracle specific versions of SRM products
- Support for Parallel clusters
- Hierarchical Storage Management facilities

CONCLUSION

The main objective for Storage Management is to maximize data availability and performance, while limiting the knowledge and effort required for performing the management tasks. End-to-end Storage Management solutions, that integrate File Systems, Volume Managers, Device Drivers, Backup/Recovery tools, and Hardware Subsystems, and are specialized for Oracle databases, aim to offer an effective answer to this challenge. The discussion in this document is intended to show the value that these solutions already bring today, but also the promise of significant further improvements in the future.