Oracle Database 10*g* High Availability

An Oracle White Paper
November 2003

ORACLE®

EXECUTIVE OVERVIEW

Enterprises have used their information technology (IT) infrastructure to provide competitive advantage, increase productivity, and empower users to make faster and more informed decisions. However, with these benefits has come an increasing dependence on that infrastructure. Should a critical application, server or data become unavailable, the entire business can be placed in jeopardy. Revenue and customers can be lost, penalties can be owed, and bad press can have a lasting effect on customers and a company's reputation. Building a high availability IT infrastructure is critical to the success and well being of all enterprises in today's fast moving economy.

Trends in computing technology are also enabling a new IT architecture, referred to as Grid computing, to be deployed. Grid computing is a new computing architecture that effectively pools large numbers of servers and storage into a flexible, on-demand computing resource for all enterprise computing needs. Technology innovations like low-cost blade servers, small and inexpensive multiprocessor servers, modular storage technologies, and open source operating systems like Linux provide the raw material for the Grid. By harnessing these technologies, and leveraging the Grid technology available in the Oracle Database 10*g*, enterprises can deliver extremely high quality of service to their users while vastly reducing their expenditures on IT. The Oracle Database 10*g* enables you to capture the cost advantages of Grid enterprise computing without sacrificing performance, scalability, security, manageability, functionality, or system availability.

In this paper we will first examine the causes of downtime. We will then and look at the technology available within the Oracle Database, with particular emphasis on the new capabilities in the Oracle Database 10*g*, that enable costly downtime avoidance and rapid recovery from failures that can not be prevented.

CAUSES OF DOWNTIME

One of the challenges in designing a highly available IT Grid infrastructure is examining and addressing all the possible causes of downtime. Figure 1 is a taxonomy of downtime which classifies it into two primary categories: unplanned and planned downtime. It is important to consider causes of both unplanned and planned downtime when designing a fault tolerant and resilient IT infrastructure.

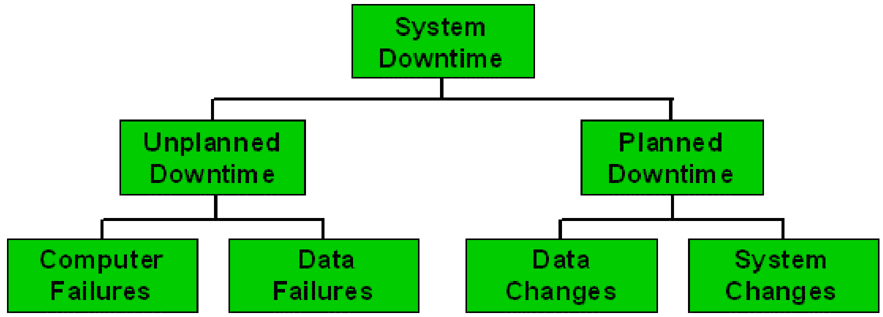

Figure 1: Causes of Downtime

Unplanned downtime is primarily the result of computer failures or data failures. Planned downtime is primarily due to data changes or system changes that must be applied to the production system. In the following sections we will in turn look at each of these four causes of downtime and examine the technology you can apply to avoid them

PROTECTING AGAINST COMPUTER FAILURES

A computer failure occurs when the computer system or database server unexpectedly fails and causes a service interruption. In most cases this is due to hardware breakdown. These type failures are best remedied by taking advantage of fast database crash recovery and cluster technology.

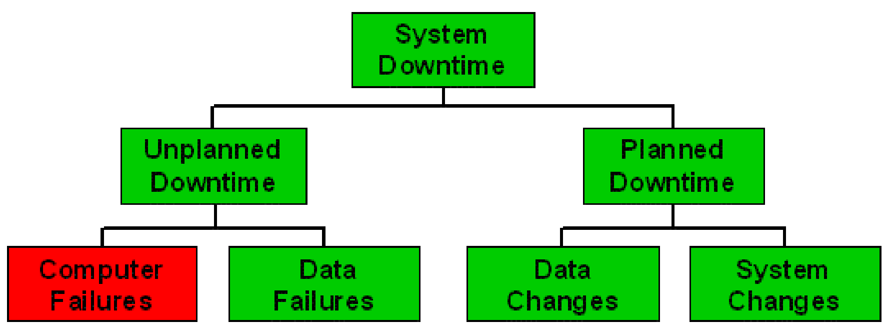

Figure 2: Computer Failures

ENTERPRISE GRIDS WITH REAL APPLICATION CLUSTERS

Real Application Clusters (RAC) enables the enterprise to build database servers across multiple systems that are highly available and highly scalable. In a Real Application Clusters environment Oracle runs on two or more systems in a cluster while concurrently accessing a single shared database. What this provides is a single database system that spans multiple hardware systems yet appears to the application as a single unified database system. This extends tremendous availability and scalability benefits for all your applications.

- Flexibility and cost effectiveness in capacity planning, so that a system can scale to any desired capacity on demand and as business needs change.
- Fault tolerance to failures within the cluster, especially computer failures.

Real Application Clusters enables enterprise Grids. Enterprise Grids are built out of large configurations of standardized, commodity-priced components: processors, servers, network, and storage. RAC is the only technology that can harness these components into useful processing system for the enterprise. Real Application Clusters and the Grid dramatically reduce operational costs and provide new levels of flexibility so that systems become more adaptive, proactive, and agile. Dynamic provisioning of nodes, storage, CPUs, and memory allow service levels to be easily and efficiently maintained while lowering cost still further through improved utilization. In addition, Real Application Clusters is completely transparent to the application accessing the RAC database and does not need to be modified in any way to be deployed on a RAC system.

Real Application Clusters gives users the flexibility to add nodes to the cluster as the demands for capacity increases, scaling the system up incrementally to save costs and eliminating the need to replace smaller single node systems with larger ones. Grid pools of standard low cost computers and modular disk arrays make this solution even more powerful with the Oracle Database 10*g*. It makes the capacity upgrade process much easier and faster since one or more nodes can be added to the cluster, compared to replacing existing systems with new and larger nodes to upgrade systems. The Cache Fusion technology implemented in Real Application Clusters and the InfiniBand support provided in the Oracle Database 10*g* enables capacity to be scaled near linearly without making any changes to your application.

Another key advantage of this cluster architecture is the inherent fault tolerance provided by multiple nodes. Since the physical nodes run independently, the failure of one or more nodes will not affect other nodes in the cluster. Failover can happen to any node on the Grid. In the extreme case, a Real Application Clusters system will still provide database service even when all but one node is down. This architecture allows a group of nodes to be transparently put online or taken off-line, for maintenance, while the rest of the cluster continues to provide database service. RAC provides built in integration with the Oracle Application Server 10*g* for failing over connection pools. With this an application is immediately notified of any failure rather than having to wait tens of minutes for a TCP timeout to occur. The application can immediately take the appropriate recovery action. And Grid load balancing will redistribute load over time.

Real Application Clusters in Oracle Database 10*g* also provides a complete set of clusterware to manage the cluster. The Oracle Database clusterware provides all the features required to run the cluster, including node membership, messaging services, and locking. And because it is a fully integrated stack with common event and management APIs, it can be centrally managed from Oracle's Enterprise Manager. There is no need to purchase additional software to support your cluster and will avoid errors by reducing cross product coordination. It also provides the same interface and operates the same way across all platforms the Oracle Database is available on. Oracle will also continue to support third-party clusterware use with RAC.

RAC also supports a new abstraction referred to as a service. Services represent classes of database users or applications. Business policies are defined and are automatically applied to these services to perform such tasks as allocating nodes for times of peak processing or to automatically handle a server failure. This ensures the application of system resources where and when they are needed to achieve business goals. Parallel hardware systems utilizing inexpensive commodity components have the potential to provide excellent price/performance advantages over traditional mainframe systems in data-intensive decision support applications. Tightly coupled Symmetric Multi-processor systems (SMP) have been the most widely used parallel hardware systems. These systems utilize multiple processors that share common memory and disk resources and hence are also known as 'shared everything' systems. Primary advantages of SMP systems include simplicity of application development and ease of administration. These systems, however, do not provide any inherent fault-tolerance—the failure of a single critical component such as a CPU could bring the entire system down. Further, they are currently somewhat limited in terms of scalability and growth due to limitations in available system bus bandwidth and operating system software scalability.

BOUNDING DATABASE CRASH RECOVERY

One of the most common causes of unplanned downtime is a system fault or crash. System faults are the result of hardware failures, power failures, and operating system or server crashes. The amount of disruption these failures cause will depend upon the number of affected users, and how quickly service is restored. High availability systems are designed to quickly and automatically recover from failures, should they occur. Users of critical systems look to the IT organization for a commitment that recovery from a failure will be fast and will take a predictable amount of time. Periods of downtime longer than this commitment can have direct effects on operations, and lead to lost revenue and productivity.

The Oracle Database provides very fast recovery from system faults and crashes. However, equally important to being fast is being predictable. The Fast-Start Fault Recovery Fault technology included in the Oracle Database automatically bounds database crash recovery time and is unique to the Oracle Database. The database will self-tune checkpoint processing to safeguard the desired recovery time objective. This makes recovery time fast and predictable, and improves the ability to meet service level objectives. Oracle's Fast-Start Fault Recovery can reduce recovery time on a heavily loaded database from tens of minutes to less than 10 seconds.

PROTECTING AGAINST DATA FAILURES

A data failure is the loss, damage, or corruption of critical enterprise data. The causes of data failure are more complex and subtle than computer failure and can be caused by a failure of the storage hardware, human error, corruption, or site failure.

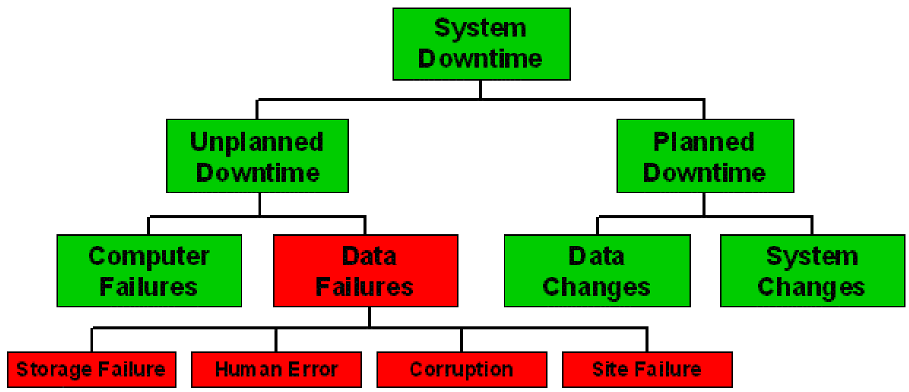

Figure 3: Data Failures

It is extremely important to design a solution to protect against and recover from data failures. A system or network fault may prevent users from accessing data, but data failures without proper backups or recovery technology can result in an a recovery taking many hours to perform, or lost data.

In Oracle Database 10*g* we have significantly enhanced our data protection capabilities. The motivation for many of these enhancements is the new economics around data protection and recovery. Over the last twenty years disk capacity has grown by three orders of magnitude. In the early 1980's a 200MB disk was state of the art. Today a state of the art disk is 200GB – and there is no end in sight for this growth in capacity. It will not be long before 500GB or 1TB disks are available. The flip side of this trend is that disk storage is getting cheaper and cheaper. As disk capacity grows the cost per megabyte has fallen to the point today where a megabyte of storage costs pennies. This makes disk storage very cheap and competitive with tape as a backup media. Plus disk has the additional benefit of being online – i.e., available all the time without delay – and provides random access to the data. These trends allowed Oracle to rethink and rearchitect our recovery strategy to take advantage of these economic dynamics. By making some disk storage available to Oracle you will be able to reduce backup and recovery time from hours to minutes. In essence you can trade inexpensive disk storage for expensive downtime.

PROTECTING AGAINST STORAGE FAILURES

Provisioning storage for a single database instance, let alone an entire enterprise, can be complex. Typically, the process includes: estimating the amount of space you're likely to need; map out what you hope will be an optimal layout (where to put data files, archive files, and so on to avoid hot-spots); create logical volumes; create file systems; define and setup how you will protect and mirror your data; define and implement your backup and recovery plan for this data; install Oracle; and finally, create your database. And then the hard work begins, looking for hotspots that negatively affect performance; moving data files around to reduce contention, and dreading the day you know will come when a disk crash occurs or when you run out of space, and have to add more disks and shift all the files around again to rebalance across your updated storage configuration.

Fortunately, that scenario is changing dramatically, with the new Automatic Storage Management (ASM) feature of the Oracle Database. ASM provides a vertically integrated file system and volume manager directly in the Oracle kernel, resulting in much less work to provision database storage, with a higher level of availability, without the expense, installation and maintenance of specialized storage products, and provides unique capabilities for database applications. ASM spreads its files across all available storage for optimal performance, and it can mirror as well, providing protection against data loss. ASM extends the concept of SAME (stripe and mirror everything) and adds more flexibility in that it can do mirroring at the database file level instead of having to mirror at the entire disk level.

But more importantly, ASM eliminates the complexity associated with managing data and disks; it vastly simplifies the processes of setting up mirroring, adding disks, and removing disks. Rather than managing hundreds, possibly thousands of files (as in a large data warehouse) DBAs using ASM create and administer a larger-grained object, the disk group, which identifies the set of disks that will be managed as a logical unit. The automation of the file naming and placement of the underlying database files save the DBAs time and ensures best practice standards are followed.

ASM's native mirroring mechanism is an option that is used to protect against storage failures. By default mirroring is enabled and triple mirroring is also available. With ASM mirroring, an additional level of data protection can be provided with the use of failure groups. A failure group is a set of disks sharing a common resource (disk controller or an entire disk array) whose failure can be tolerated. Once defined, an ASM failure group will intelligently place redundant copies of the data in separate failure groups to ensure that the data will be available and transparently protected against the failure of any component in the storage subsystem. In addition ASM supports the Hardware Assisted Resilient Data capability (discussed below in the Protecting Against Data Corruptions section) to further protect your data.

PROTECTING AGAINST HUMAN ERRORS

Almost any research done on the causes of downtime identifies human error as the single largest cause of downtime. Human errors like: the inadvertent deletion of important data; or when an incorrect WHERE clause in an UPDATE statement updates many more rows than were intended; need to be prevented wherever possible, and undone when the precautions against them fail. The Oracle Database provides easy to use yet powerful tools that help administrators quickly diagnose and recover from these errors, should they occur. It also includes features that allow end-users to recover from problems without administrator involvement, reducing the support burden on the DBA, and speeding recovery of the lost and damaged data.

GUARDING AGAINST HUMAN ERRORS

The best way to prevent errors is to restrict a user's access to data and services they truly need to conduct their business. The Oracle Database provides a wide range of security tools to control user access to application data by authenticating users and then allowing administrators to grant users only those privileges required to perform their duties. In addition the security model of Oracle Database provides the ability to restrict data access at a row level, using the Virtual Private Database feature, further isolating users from data they do not need not access to.

ORACLE FLASHBACK TECHNOLOGY

When authorized people make mistakes you need the tools to correct these errors. The Oracle Database 10*g* provides a family of human error correction technology called Flashback. Flashback revolutionizes data recovery. In the past it might take minutes to damage a database but hours to recover it. With Flashback the time to correct errors equals the time it took to make the error. It is also extremely easy to use and a single short command can be used to recover the entire database instead of following some complex procedure. Flashback provides a SQL interface to quickly analyze and repair human errors. Flashback provides fine-grained surgical analysis and repair for localized damage -- like when the wrong customer order is deleted. Flashback also allows for correction of more widespread damage yet does it quickly to avoid long downtime -- like when all of this month's customer orders have been deleted. Flashback is unique to the Oracle Database and supports recovery at all levels including the row, transaction, table, tablespace, and database wide.

FLASHBACK QUERY

Oracle Flashback Query, a feature introduced in the Oracle9i Database allows an administrator or user to query any data at some point-in-time in the past. This powerful feature can be used to view and reconstruct lost data that may have been deleted or changed by accident. For example:

```
Select * from EMPLOYEE as of ‘2:00 P.M.’ where …
```

This statement displays rows from the table EMPLOYEE from 2pm today. Developers can use this feature to build self-service error correction into their applications, empowering end-users to undo and correct their errors without delay, rather than burdening administrators to perform this task. Flashback Query is extremely simple to manage, as the database automatically keeps the necessary information to reconstruct data for a configurable time into the past.

FLASHBACK VERSIONS QUERY

The Flashback Versions Query provides a way to view changes made to the database at the row level. It is an extension to SQL and allows the retrieval of all the different versions of a row across a specified time interval. For example:

```
Select * from EMPLOYEE versions between ‘2:00
PM’ and ‘3:00 PM’ where …
```

This statement displays each version of the row – each changed by a different transaction – between 2 and 3 PM today. Using this a DBA can pinpoint when and how data is changed and trace it back to the user, application, or transaction. This allows the DBA to track down the source of a logical corruption in the database and correct it. It also enables the application developer to debug their code.

FLASHBACK TRANSACTION QUERY

Flashback Transaction Query provides a way to view changes made to the database at the transaction level. It is an extension to SQL that allows you to see all changes made by a transaction. For example:

```
Select * from DBA_TRANSACTION_QUERY where xid =
‘000200030000002D’;
```

This statement will show all the resultant changes made by this transaction. In addition, compensating SQL statements are returned and can be used to undo changes made to all rows by this transaction. Again using a precision tool like this the DBA and application developer can precisely diagnose and correct logical problems in the database or application.

FLASHBACK DATABASE

To bring an Oracle database to a previous point in time, the traditional method is to do point in time recovery. However, point-in-time recovery can take hours, or even days, since it requires the whole database to be restored from backup and recovered to the point in time just before the error was introduced into the database. With the size of databases constantly growing, it will take hours or even days just to restore the whole database.

Flashback Database is a new strategy for doing point in time recovery. It quickly rewinds an Oracle database to a previous time to correct any problems caused by logical data corruption or user error. Flashback logs are used to capture old versions of changed blocks. One way to think of it is as a continuous backup or storage snapshot. When recovery needs to be performed the Flashback logs are quickly replayed to restore the database to a point in time before the error and just the changed blocks are restored. It is extremely fast and reduces recovery time form hours to minutes. In addition it is extremely easy to use. A database can be recovered to 2:05 PM by simply issuing the following single command.

```
FLASHBACK DATABASE to '2:05 PM';
```

No restore from tape, no lengthy downtime, and no complicated recovery procedures are required to use it. You can also use Flashback, then open the database read-only and examine its contents. If you determine you Flashbacked to far, or not far enough, you can reissue the flashback command to find the proper point in time before the database was damaged. Flashback is also integrated with Data Guard so you Flashback the production and standby databases together (see the Data Guard section below).

Flashback Database is just like having a rewind or undo button for the database.

FLASHBACK TABLE

Flashback Transaction Query provides a way to view changes made to the database at the transaction level. It is an extension to SQL that allows you to see all changes made by a transaction. For example:

```
FLASHBACK TABLE orders, order_items TO TIMESTAMP
(JUL-07-2003, 02:33:00);
```

This command would rewind any updates to the orders and order_items tables that have been done between the current time and specified timestamp in the past. Flashback Table performs this operation online and in-place and it maintains any referential integrity constraints between the tables.

Flashback Table is just like having a rewind or undo button for a table, or set of related tables.

FLASHBACK DROP

Dropping, or deleting, database objects by accident is a mistake people have, and probably always, will make.. "I thought I was connected to the test database when I deleted that table" has been heard to often by Oracle Support. Users soon realize their mistake but by then it is too late and there is no way to easily recover the dropped tables and its indexes, constraints, and triggers. Objects once dropped were dropped forever. In case of really important tables or other objects (like indexes, partitions or clusters), the DBAs had to perform a point-of-time recovery, which may be very time consuming and lead to loss of recent transactions.

Flashback Drop provides a safety net when dropping objects in Oracle Database 10*g*. When a user drops a table, Oracle will place it in a Recycle Bin. Objects in the Recycle Bin will remain there until user decides to permanently remove them or the space pressure is placed on the tablespace containing the table. The recycle bin is a virtual container where all dropped objects reside. Users can look in the Recycle Bin and "undrop" the dropped table and its dependent objects. For example the table employee and all its dependent objects would be "undropped" by the following command.

```
FLASHBACK TABLE employee BEFORE DROP;
```

Flashback Drop is just like having an undrop button for a table, and its dependent objects.

LOGMINER™ SQL-BASED LOG ANALYZER

Oracle log files contain a wealth of useful information about the activities and history of an Oracle database. Log files contain all of the data needed to perform database recovery. They also record every change made to data and metadata in the database. LogMiner is a fully relational tool that allows redo log files to be read, analyzed, and interpreted using SQL. Analysis of the log files with LogMiner can be used to track or audit changes to data, provide supplemental information for tuning and capacity planning, retrieve critical information for debugging complex applications, or recovering deleted data.

PROTECTING AGAINST DATA CORRUPTIONS

A corruption is created by a faulty component in the IO stack. For example the database issues IOs as the result of an update transaction. The database IOs are passed to: the IO code in the operating system which passes it to the file system; which passes it to the volume manager; which passes it to the device driver; which passes it to the Host-Bus-Adapter; which passes it to the storage controller; which passes it to the disk drive to finally be written. Bugs or a hardware failure in any component in the IO stack could "flip some bits" in the data resulting in corrupt data being written to the database. This corruption could be to database control information or user data either of which could be catastrophic to the functioning and availability of the database. Similarly a disk failure could damage database files requiring backups be used to recover the database.

ORACLE HARDWARE ASSISTED RESILIENT DATA (HARD)

Oracle's Hardware Assisted Resilient Data (HARD) is a comprehensive program designed to prevent data corruptions before they happen. Data corruptions while rare, can have a catastrophic effect on a database, and therefore a business. By implementing Oracle's data validation algorithms inside storage devices, Oracle will prevent corrupted data from being written to the database files on permanent storage. This type of end-to-end high-level software to low-level hardware validation is a unique capability provided by Oracle its our storage partners. Oracle validates and adds protection information to the database blocks and the protection information is validated by the storage device. HARD prevents corruptions from being introduced into the IO path between the database and storage and eliminates a large class of failures that the database industry has previously been powerless to prevent. RAID has gained a wide following in the storage industry by ensuring the physical protection of data, HARD takes data protection to the next level by going beyond protecting physical bits, to protecting business data.

HARD was first made available with the Oracle9i Database but has been extended in the Oracle Database 10*g*. The validation is more comprehensive and all file types and block are protected by HARD including database files, online logs, archive logs and backups. In addition ASM enables HARD without having to use raw disk devices. HARD support is offered in conjunction with many of the major storage vendors.

FLASH BACKUP AND RECOVERY

There is no substitute for backups of enterprise data. Although rare, multiple failures can render even data mirrored in the storage subsystem unusable. Fortunately, Oracle provides online tools to properly backup all your data, to restore data from a previous backup, and to recover changes to that data up to the time just before the failure occurred.

Backing up a large database system is no simple task. A large database can be composed of hundreds of files spread over many different disks. Neglecting to backup a critical file can render the entire database backup unusable. Often these damaged files are not discovered until they are needed. Recovery Manager (RMAN) is a tool that manages the backup, restore, and recovery process for the Oracle Database. It creates and maintains backup policies, and catalogs all backup and recovery activities. All data blocks can be analyzed for corruption during backup and restore to prevent propagation of corrupt data through to the backups. Most importantly, Recovery Manager ensures all necessary data files are backed up, and the database is recoverable.

Recovery Manager automatically keeps track of the files needed to restore the database within a user-specified window. It can automatically restart interrupted operations, handle corrupted log files, and restore an individual data block while the remainder of the database remains online.

RMAN in the Oracle Database 10*g* radically enhances database backup and recovery. RMAN can automatically manage backing up and recovering all your data to the Flash Recovery area. The Flash Recovery Area is a unified disk-based storage location for all recovery related files and activities in an Oracle database. Given the new economics of storage that were previously discussed in this paper, making backups to disk instead of tape, enables faster backups. But more importantly, if database media recovery is required, then your datafile backups are readily available radically speeding database recovery time.

Recovery Manager manages the recovery files in the Flash Recovery Area. RMAN will automatically create all backups in the Flash Recovery Area and manage the space. The archiver will write archive logs to the Flash Recovery Area and RMAN will automatically delete, or move to tape, obsolete backups and archive logs that are no longer required. If you set the RETENTION POLICY to a recovery window of 7 days, then RMAN will retain all backups required to recover the database 7 days back. If recovery to a time further than 7 days in the past is required RMAN will restore the data from tape. And Enterprise Manager provides a complete interface to drive Flash Backup and Recovery including implementing best practices.

Incremental backups have been part of RMAN since it was first released in the Oracle8 Database. Incremental backups provide the capability to backup only the changed blocks since the previous backup. Oracle Database 10*g* delivers the ability for faster incrementals with the implementation of block change tracking. The Oracle Database 10*g* tracks the physical location of all database changes. RMAN automatically use this change tracking information to determine which blocks need to be read during an incremental backup and directly access the block to back it up. The incremental backups can then be merged into a previously created image backup to minimize the time for recovery. A backup strategy based on incrementally updated backups keeps the time required for media recovery to a minimum. By making incremental backups with change tracking part of your backup strategy you can: reduce the amount of time needed for daily backups; save network bandwidth when backing up over a network; recover unlogged changes to database; reduce the backup file storage; and reduce the time for database recovery.

Oracle Database 10*g* backup and recovery also includes many other innovative capabilities including:

- Compression of backups.
- Automated failover to a previous backup when restore discovers a missing or corrupt backup.
- Automated recovery through a previous point in time recovery - recovery through resetlogs.
- Automated creation of new files during recovery.
- Automated channel failover on backup or restore.
- Automated tablespace point-in-time recovery.
- Full DB "begin backup" command for faster mirror split.
- Improved Recovery Parallelism (2 to 4 times).
- Tablespace Rename.
- Proxy (third-party) Backup for archive logs.

- Time window based throttling of backups.
- Cross Platform Transportable Tablespaces.

PROTECTING AGAINST SITE FAILURES

Data protection features provide protection from catastrophic events that cripple processing at a site for an extended period of time. Examples include file corruptions, natural disasters, power and communication outages, and even terrorism. The Oracle Database offers a variety of data protection solutions that provide the ability to create and maintain a local or remote copy of a production database. In the event of a corruption or disaster, users of the data can continue to function by accessing the remote database.

The simplest form of data protection is off-site storage of database backups. In the event a data center is unable to resume services in a reasonable amount of time, the backups can be restored on a system at another site, and users can connect to the backup system. Unfortunately, restoring backups on another system will be time consuming, and the backup may not be completely up-to-date. To more quickly recover and maintain continuous database service even in the event of a disaster, Oracle provides Data Guard.

DATA GUARD

Data Guard should be the foundation of any Oracle Database disaster recovery plan. Data Guard provides the ability to set up and maintain a standby copy of your production database. This standby database can be located a half a world away from the production database or in the same data center. Data Guard includes enhancements to automate complex tasks and provide significant monitoring, alerting and control mechanisms. It enables your database to survive a data center disaster. Data Guard also works transparently across GRID clusters as the servers can be added dynamically to the standby database in the event a failover is required.

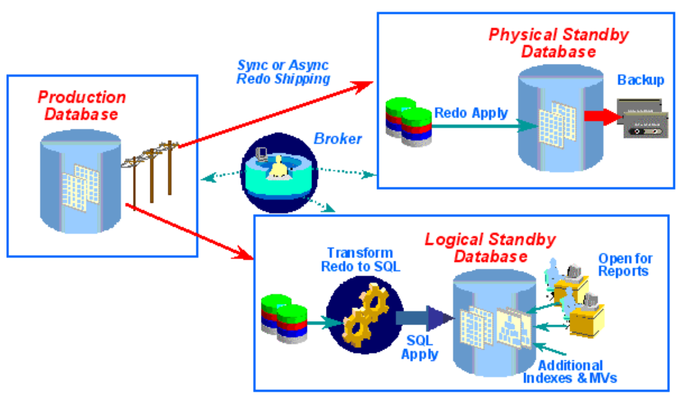

Figure 4: Data Guard Architecture

DATA GUARD REDO APPLY

Data Guard in Redo Apply mode maintains a copy of a production database, called a physical standby database, and keeps it synchronized with the production database. The redo data from the primary database are shipped to the standby and physically applied via media recovery. The standby database is physically identical to the primary (although it may lag the primary). Additionally the standby database can be opened read-only so it can also be used to off load reporting work from the production database. Backup processing may also be offloaded from the production database as backups created at the standby database can be used to perform recovery of the production database.

Physical standby databases are good for providing protection from disasters and data errors. In the event of an error or disaster, the physical standby can be opened, and be used to provide data services to applications and end-users. Because the efficient media recovery mechanism is used to apply changes to the standby database, it is supported with every application, and can easily and efficiently keep up with even the largest transaction workloads.

DATA GUARD SQL APPLY

Data Guard in SQL Apply mode takes the Oracle archive logs, transforms them into SQL transactions, and then applies them to an open standby database. Although the standby database, called a logical standby database, may be physically different from the primary database, it is logically the same as the primary and it can be used to take over processing if the primary database is destroyed. Because transactions are applied using SQL to an open database, the standby can be used concurrently for other tasks, and can have a different physical structure than the production database. For example, the logical standby can be used for decision support, and be optimized for reporting by using additional indexes and materialized views that do not exist on the primary database.

Data Guard SQL Apply is most importantly a data protection feature. Data Guard in SQL Apply compares before-change values in the log files to the before-change values in the logical standby database providing a check against logical corruption. A logical standby database can therefore offer protection from the widest possible range of corruptions.

Because logical standby databases are open for read and write during recovery, it is possible to query the standby database while the changes in the redo logs are being applied.

ZERO DATA LOSS LOG TRANSPORT

The same log transport services are used by both physical and logical standby database components. Traditionally, archive logs are shipped from the primary to the standby as soon as they are created. Data Guard can synchronously write redo log updates directly from the primary to the standby database. This provides a comprehensive “zero data loss” disaster recovery solution. Should a disaster strike at the primary site, all the redo necessary to preserve all transactions will be available for an application at the standby site.

Administrators may also choose to ship redo data asynchronously to the standby site. This will minimize any potential data loss while providing optimal performance over large distances, and provide protection from network failures.

REAL TIME APPLY AND FLASHBACK DATABASE

New in Oracle Database 10*g* Data Guard is the Real Time apply feature and integration with Flashback Database. With the real time apply feature, log apply services can apply redo data on the standby database as soon as it is received from the primary database, without waiting for the current log file to be archived at the standby database. This enables standby databases to be closely synchronized with the primary database, enabling up-to-date and real-time reporting. This also enables faster switchover and failover times, which in turn reduces planned and unplanned downtime for the business. The DBA may also choose to use the Flashback Database feature of Oracle Database 10*g* on both the primary and standby database to quickly revert the databases to an earlier point-in-time to back out user errors. Alternatively, if the administrator decides to failover to a standby database, but those user errors were already applied to the standby database (because Real Time Apply was enabled), the administrator may simply Flashback the standby database to a safe point in time. The use of these two features eliminate the tradeoff some customers make between ensuring the standby database stays current and delaying redo apply to prevent human errors on the production database from propagating to the standby database.

DATA GUARD BROKER

Data Guard includes an easy to use GUI, the Data Guard Manager, which is part of Oracle’s Enterprise Manager Grid Control. A command line interface that provides monitoring, automation, and management of the Data Guard components is also available. With a single mouse click the Data Guard Manager can fail over processing from the primary to either type of standby database. The Data Guard Manager makes it easy for the DBA to manage and operate the standby database. By facilitating activities such as failover and switchover, the possibility of errors is greatly reduced.

AVOIDING PLANNED DOWNTIME

Planned downtime can be just as disruptive to operations, especially in global enterprises that support users in multiple time zones. In this case it is important to design a system to minimize planned interruptions. Planned downtime include routine operations, periodic maintenance, and new deployments.

Routine operations are frequent maintenance tasks that include backups, performance management, user and security management, and batch operations. Periodic maintenance, such as installing a patch or reconfiguring the system, is occasionally necessary to update the database, application, operating system, middleware, or network. New deployments such as major upgrades to the hardware, operating system, database, application, middleware, or network. It is important to consider not only the time to perform the upgrade, but also the effect the changes may have on the overall application.

The Internet has made it easy to share data globally, but brings new challenges and requirements for data availability. As global users access data 24 hours per day, maintenance windows have all but evaporated. Planned downtime is becoming as disruptive as unplanned downtime. There are no longer any windows of time during which users are not affected. When the volume of data stored in a database becomes very large, maintenance operations can be quite time consuming. It is important that these operations be performed without affecting the users of the data.

AVOIDING DOWNTIME FOR DATA CHANGES

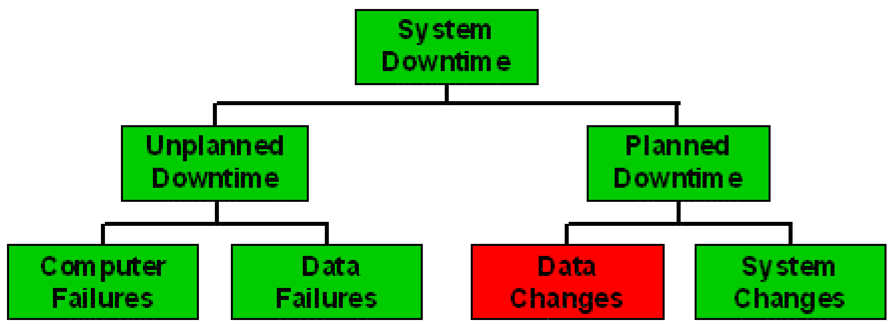

Figure 5: Data Changes

ONLINE SCHEMA AND DATA REORGANIZATION

The Oracle Database supports many maintenance operations without disrupting database operations or users updating or accessing data. Indexes can be added, rebuilt, or defragmented while the database is online and end users are reading or updating data. Similarly, tables can be relocated or defragmented while online. Tables can be redefined, changing table types, adding, dropping or renaming columns, and changing storage parameters without interruption to end-users who are viewing or updating the underlying data. In the Oracle Database 10*g* this capability has been enhanced to:

- support the easy cloning of indexes, grants, constraints, and other characteristics of the table;

- convert from the long to LOB datatype online; and
- allow unique indexes instead of requiring a primary key.

Java™ and PL/SQL™ stored procedures can be updated dynamically and Oracle manages all dependencies, to properly integrate the new procedures into the database, with no impact to end-user operations. In the Oracle Database 10*g* this capability has been enhanced to allow many kinds of tables changes to be made without recompiling stored procedures associated with that table.

PARTITIONED TABLES AND INDEXES

As databases grow larger, they may become extremely cumbersome to manage. The ability to partition database tables and indexes allows administrators to divide large tables up into smaller more manageable pieces. While most operations and schema changes can be made online, partitioning allows maintenance tasks to be performed one partition at a time. This allows the bulk of the data to be unaffected during maintenance. In addition, partitions enable the use of parallel execution to perform most operations much faster.

Another benefit of partitions is fault containment. A failure, such as a media failure or corruption, is contained to partitions resident on the failed disk. Only that partition is affected and needs to be recovered. This not only reduces the time to recover, but allows the other unaffected partitions to remain online while the failed partition is recovered.

Often, not all data in a large table has the same access characteristics. Pending orders may be accessed more frequently than closed orders, or analysis of last quarter's sales may be more common than analysis of sales from a quarter three years ago. Partitioning allows for intelligent storage management of data. Frequently accessed data can be stored on the fastest disks, and heavily accessed data can be striped across many drives.

DYNAMIC RESOURCE PROVISIONING

The Oracle Database continues to broaden support for dynamic reconfiguration enabling it to adapt to changes in demand and hardware with no disruption of service. The Oracle Database dynamically accommodates changes to hardware configurations such as:

- add and remove of processors from an SMP server;
- add and remove nodes in a RAC cluster;
- dynamically grow and shrink its shared memory allocation and automatically tune memory online;
- add and remove database disks online without disturbing database activities;
- automatically rebalance IO load across the database storage; and

- move datafiles online.

These capabilities truly provide no cost system changes and capacity on demand provisioning that is a fundamental requirement for enterprise Grid computing.

AVOIDING DOWNTIME FOR SYSTEM CHANGES

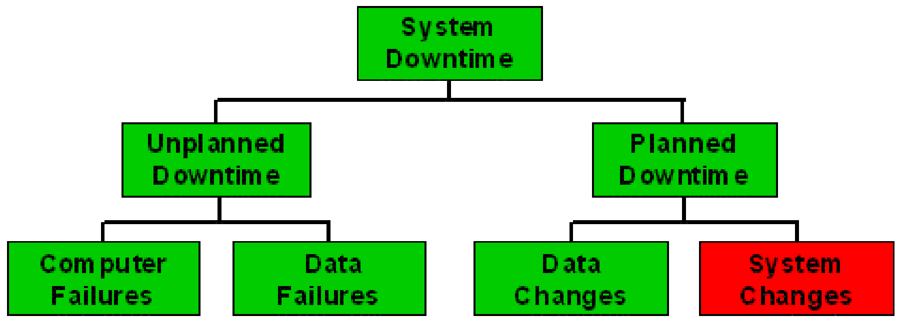

Figure 6: System Changes

ROLLING PATCH UPDATES

The Oracle Database supports the application of patches to the nodes of a Real Application Clusters (RAC) system in a rolling fashion. The method by which this is done is shown below.

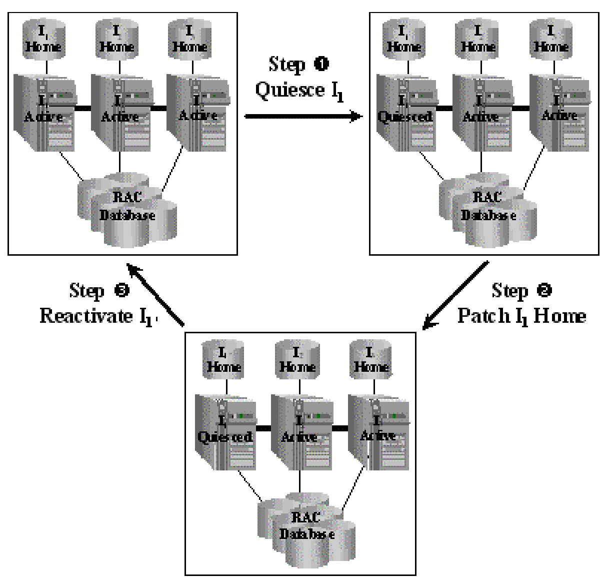

Figure 7: Rolling Patch Updates

A RAC system runs with all nodes actively processing transactions on the behalf of database clients (upper left hand RAC system in figure 7). Step 1 of the patch application procedure is to quiesce the first instance to which the patch is to be applied (instance 1 in this example). In step 2 an Oracle patch tool (opatch) is used to apply the patch to the quiesced instance (the Oracle Home for instance 1 is updated). In step 3 the patched instance is reactivated and rejoins the cluster. The RAC system is now running with one instance at a higher maintenance level than the other nodes in the cluster.

A RAC system can run in this mixed mode for an arbitrary period to ensure the patch corrects the original problem, and has not introduced some other problem. This procedure is then repeated for the remaining nodes in the cluster. When all nodes in the cluster have been patched the rolling patch update is complete and all nodes are running the same version of the Oracle Database software. In addition, opatch has the ability to "rollback" the application of a patch. If some aberrant behavior is observed on the updated instance, the offending patch can be uninstalled, or rolled back, without forcing a cluster-wide outage. The rollback procedure is the same as the patch apply procedure but opatch in this case removes a previously applied patch.

ROLLING RELEASE UPGRADE

The Oracle Database 10*g* supports the installation of database software upgrades, and the application of patchsets, in a rolling fashion – with near zero database downtime – by using Data Guard SQL Apply. The method by which this is done is shown below.

Figure 8: Establish the Standby Database

After instantiating the standby database and configuring Data Guard to replicate changes made on the production database to the standby database, the standby database is upgraded.

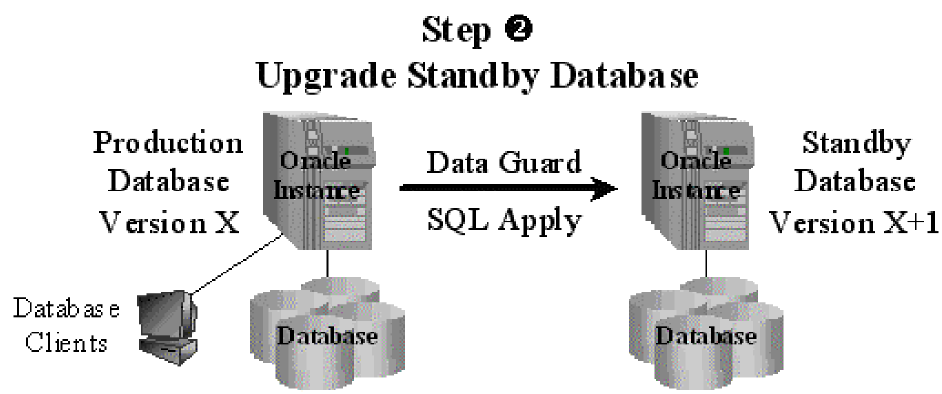

Figure 9: Upgrade the Standby Database

This configuration can run in this mixed mode for an arbitrary period to validate the upgrade in the production environment. When satisfied the upgraded software is operating properly, a database role reversal and switchover can be done.

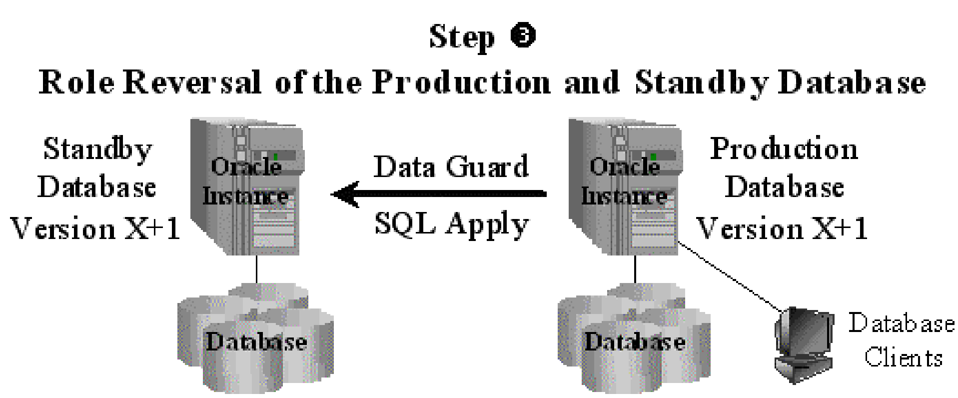

Figure 10: Role Reversal

The role reversal is composed of several sub-steps including: Data Guard switchover making the original standby database the production database; re-pointing the database clients to the new production database; upgrading the standby database; and raising the compatibility level of both databases. There are several points during this process the configuration is run in a mixed mode to validate the upgrade. During those times the upgrade can be aborted and the software downgraded, without data loss. During the rolling upgrade the standby database is available for disaster recovery. For additional data protection during these steps, a second standby database may be used in the Data Guard configuration.

By supporting rolling upgrades and rolling patch updates, Oracle has eliminated a large portion of the maintenance windows DBAs reserve for administrative tasks, and enables 24x7 operation of their enterprise.

MAXIMUM AVAILABILITY ARCHITECTURE (MAA) BEST PRACTICES

Operational best practices are key to the successful implementation of IT infrastructure. Technology alone is not enough. Oracle's Maximum Availability Architecture (MAA) is a fully integrated and proven blueprint for building highly available systems. Enterprises that have based their system architecture on MAA find they can quickly and efficiently design and deploy applications that meet their business requirements for system availability. MAA encompasses specific design and configuration recommendations, which have been extensively reviewed and tested to ensure optimum system availability and reliability. The MAA blueprint examines and details the combined use of key Oracle Database features for high availability including Real Application Clusters, Data Guard, Recovery Manager, and Enterprise Manager. It also addresses the configuration and integration of other critical components of highly available systems including servers, storage, networking, and the application server. Each of these features individually provides Oracle customer's solutions for high availability. The right combination of these, fully integrated and following the Maximum Availability Architecture, results in a system with maximum availability and is the unbreakable solution.

The MAA best practices guide is being extended and will become a standard part of the Oracle Database 10*g* documentation set. You can find more information on MAA at http://otn.oracle.com/deploy/availability/htdocs/maa.htm

CONCLUSION

As the key component of your IT infrastructure the Oracle Database provides the features and tools to ensure data access and availability for mission critical applications. The Oracle Database 10*g* includes important and revolutionary availability features to ensure your data and database are available whenever and wherever it is needed. And the Grid capabilities provided make certain that the cost to deploy your database environment and adapt to changing business needs are assured.

White Paper Title
November 2003
Author: Ron Weiss

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com