Oracle® Database

High Availability Overview

12*c* Release 2 (12.2)

E49939-06

January 2017

ORACLE®

Oracle Database High Availability Overview, 12*c* Release 2 (12.2)

E49939-06

Primary Author: Virginia Beecher

Contributing Authors: Lance Ashdown, Tulika Das, Viv Schupmann, Janet Stern, Lawrence To

Contributors: Ahmed Abbas, Andrew Babb, Hermann Baer, Tammy Bednar, Peter Belknap, Janet Blowney, Larry Carpenter, Immanuel Chan, Dib Chatterjee, Tim Chien, Donna Cooksey, Mark Dilman, Ray Dutcher, Richard Exley, Craig Foch, Stephan Haisley, Ameet Kini, Frank Kobylanski, Rene Kundersma, Bryn Llewellyn, Barb Lundhild, Rahim Mau, Patricia McElroy, Joe Meeks, Markus Michalewicz, Valarie Moore, Dan Norris, Michael Nowak, Darryl Presley, Hector Pujol, Ashish Ray, Mark Scardina, Jia Shi, Michael T. Smith, Vinay Srihari, Andrew Steinorth, Hubert Sun, Lawrence To, Douglas Utzig, James Viscusi, Tak Wang, Shari Yamaguchi

Contents

Preface

This book introduces you to Oracle best practices for deploying a highly available database environment. It provides an overview of high availability and helps you to determine your high availability requirements. It describes the Oracle Database products and features that are designed to support high availability and describes the primary database architectures that can help your business achieve high availability.

This preface contains these topics:

Audience

This book is intended for chief technology officers, information technology architects, database administrators, system administrators, network administrators, and application administrators who perform the following tasks:

- Plan data centers
- Implement data center policies
- Maintain high availability systems
- Plan and build high availability solutions

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

Knowledge of Oracle Database, Oracle RAC, and Data Guard concepts and terminology is required to understand the configuration and implementation details described in this book. For more information, see the Oracle Database documentation set. These books may be of particular interest:

- *Oracle Database High Availability Best Practices*

  This book typically lags behind the *Oracle Database High Availability Overview* because extensive testing is required to determine the best practices. Until the release 12.2 book is available, you may find some of the methodologies in the *Oracle Database High Availability Best Practices* for release 12.1.0.2 to be useful.

- *Oracle Database Administrator's Guide*
- *Oracle Database 2 Day + Real Application Clusters Guide*
- *Oracle Clusterware Administration and Deployment Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Automatic Storage Management Administrator's Guide*
- *Oracle Data Guard Concepts and Administration*
- *Oracle Database Backup and Recovery User's Guide*

Many books in the documentation set use the sample schemas of the seed database, which is installed by default when you install Oracle Database. See *Oracle Database Sample Schemas* for information about using these schemas.

Also, you can download the Oracle MAA best practice white papers at http://www.oracle.com/goto/maa.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Changes in This Release for Oracle Database High Availability Overview

This preface contains:

Changes in Oracle Database 12*c* Release 2 (12.2)

The following are changes in *Oracle Database High Availability Overview* for Oracle Database 12*c* Release 2 (12.2).

New Features

The following features are new in this release:

- Oracle Sharding MAA

  Sharding is a new architecture pattern for the Oracle Database that is completely compatible with existing MAA reference architectures.

  See Oracle Sharding MAA Reference Architecture (page 2-7) for information about this new architecture.

- Oracle Multitenant MAA

  Oracle Multitenant is fully compliant with and takes direct advantage of high availability features.

  See Oracle Multitenant (page 3-53) for information about the latest benefits.

- Oracle In-Memory MAA

  Enhanced support for Oracle Database In-Memory contribute to its benefits in any of the MAA reference architectures. Support for Oracle Database In-Memory and its enhancements are noted throughout this document.

  Also see the MAA white paper Oracle Database In-Memory High Availability Best Practices.

- Oracle Data Guard

  Many enhancements to Oracle Data Guard contribute to its benefits in an MAA architecture.

  See Oracle Data Guard (page 3-2) for information about the latest benefits.

- Oracle GoldenGate

  Many enhancements to Oracle GoldenGate help you create the best of breed replication solution.

  See Oracle GoldenGate (page 3-12) for information about the latest benefits.

- Application Availability

  Enhancements to Application Continuity, Transaction Guard, Java, OCI, and JDBC for end-to-end application availability.

  See Client and Application Failover (page 3-41) for information about the latest benefits.

- Online Operations

  New and enhanced online operations keep your databases up and running during planned maintenance.

  See Oracle Database High Availability Solutions for Planned Downtime (page 5-1) for information about the latest benefits.

- Oracle Recovery Manager (RMAN)

  Enhancements to RMAN contribute to its benefits in an MAA architecture.

  See Recovery Manager (page 3-17) for information about the latest benefits.

- Upgrade Enhancements

  More support for rolling upgrades contribute to less downtime.

  See Oracle High Availability Solutions for System and Software Maintenance (page 5-18) for information about the latest benefits.

See Also:

The list of availability features in *Oracle Database New Features Guide*

Other Changes

The following are additional changes in the release:

- Oracle MAA Reference Architectures

  Updates to MAA Reference Architectures to reflect both Oracle Database on-premise and Oracle Cloud, and new Oracle Sharding MAA reference architecture.

  See High Availability Architectures (page 7-1).

Changes in Oracle Database 12*c* Release 1 (12.1)

The following are changes in *Oracle Database High Availability Overview* for Oracle Database 12*c* Release 1 (12.1).

New Features

The following features are new in this release:

- Global Data Services

Global Data Services applies the Oracle Real Application Clusters service model to sets of globally distributed, heterogeneous databases, providing load balancing and failover capabilities to database clouds with *global services*, which are services provided by multiple databases.

See Oracle Database with Global Data Services (page 3-51).

- Oracle Flex Clusters and Oracle Flex ASM

  Oracle Clusterware and Oracle Real Application Clusters can be configured in large clusters, called an Oracle Flex Cluster. Oracle Flex ASM decouples the Oracle ASM instance from the database servers. Oracle ASM instances may be run on separate physical servers (from the database servers). Any number of Oracle ASM servers can be clustered together to support a large set of databases.

  See Oracle Real Application Clusters and Oracle Clusterware (page 3-20) and Oracle Automatic Storage Management (page 3-25).

- Far sync

  A Data Guard far sync instance is a remote Data Guard destination that accepts redo from the primary database and then ships that redo to other members of the Data Guard configuration. A far sync instance manages a control file, receives redo into standby redo logs (SRLs), and archives those SRLs to local archived redo logs, but that is where the similarity with standbys ends. A far sync instance does not have user data files, cannot be opened for access, cannot run redo apply, and can never function in the primary role or be converted to any type of standby database.

  See Benefits of Oracle Active Data Guard (page 3-6).

- Consolidation and multitenant architecture

  The multitenant architecture feature enables an Oracle database to contain a portable set of schemas, objects, and related structures that appears logically to an application as a separate database.

  See High ROI Using Oracle Database Consolidation (page 9-6).

- Rolling upgrade using Oracle Active Data Guard

  Rolling Upgrade using Oracle Active Data Guard provides new PL/SQL packages that automate much of the process of performing a database rolling upgrade using a physical standby database.

  See Performing Database Upgrades Using Data Guard and Physical Standby Databases (page 5-27).

- Oracle Active Data Guard enhancements

  Support for Global Temporary Tables, replication of XMLType tables and columns, and enhanced security.

  See Benefits of Oracle Active Data Guard (page 3-6).

- Application failover improvements

  Application availability has been improved with the enhancement of Fast Application Notification, Oracle Service, and with the addition of Global Data Services, Application Continuity, and Transaction Guard.

  See Client and Application Failover (page 3-41).

Other Changes

The following are additional changes in the release:

- Oracle MAA Reference Architectures

  Oracle MAA reference architectures are applicable for a single database or application or for thousands of databases and applications, for DBaaS cloud or database consolidation. The entire contents of this document has been altered to highlight the MAA reference architectures.

  See High Availability Architectures (page 7-1).

1

Overview of High Availability

This chapter contains the following sections:

1.1 What Is High Availability?

Availability is the degree to which an application, service, or function is accessible on demand. Availability is measured by the perception of an application's user. Users experience frustration when their data is unavailable or the computing system is not performing as expected, and they do not understand or care to differentiate between the complex components of an overall solution. Performance failures due to higher than expected usage create the same disruption as the failure of critical components in the architecture. If a user cannot access the system, it is said to be **unavailable**. Generally, the term **downtime** is used to refer to periods when a system is unavailable.

Users who want their systems to be always ready to serve them need **high availability**. A system that is highly available is designed to provide uninterrupted computing services during essential time periods, during most hours of the day, and most days of the week throughout the year; this measurement is often shown as **24x365**. Such systems may also need a high availability solution for planned maintenance operations such as upgrading a system's hardware or software.

Reliability, recoverability, timely error detection, and continuous operations are primary characteristics of a highly available solution:

- **Reliability**: Reliable hardware is one component of a high availability solution. Reliable software—including the database, web servers, and applications—is just as critical to implementing a highly available solution. A related characteristic is resilience. For example, low-cost commodity hardware, combined with software such as Oracle Real Application Clusters (Oracle RAC), can be used to implement a very reliable system. The *resilience* of an Oracle RAC database allows processing to continue even though individual servers may fail.

- **Recoverability**: There may be many ways to recover from a failure. Therefore, it is important to determine what types of failures may occur in your high availability environment and how to recover from those failures quickly in order to meet your business requirements. For example, if a critical table is accidentally deleted from the database, what action should you take to recover it? Does your architecture

provide the ability to recover in the time specified in a service-level agreement (SLA)?

- **Timely error detection**: If a component in your architecture fails, then fast detection is essential to recover from the unexpected failure. Although you may be able to recover quickly from an outage, if it takes an additional 90 minutes to discover the problem, then you may not meet your SLA. Monitoring the health of your environment requires reliable software to view it quickly and the ability to notify the database administrator of a problem.
- **Continuous operation**: Providing continuous access to your data is essential when very little or no downtime is acceptable to perform maintenance activities. Activities, such as moving a table to another location in the database or even adding CPUs to your hardware, should be transparent to the user in a high availability architecture.

More specifically, a high availability architecture should have the following traits:

- Tolerate failures such that processing continues with minimal or no interruption
- Be transparent to—or tolerant of—system, data, or application changes
- Provide built-in preventive measures
- Provide active monitoring and fast detection of failures
- Provide fast recoverability
- Automate detection and recovery operations
- Protect the data to minimize or prevent data loss
- Implement the operational best practices to manage your environment
- Achieve the goals set in SLAs (for example, recovery time objectives (RTOs) and recovery point objectives (RPOs)) for the lowest possible total cost of ownership

1.2 Importance of Availability

The importance of high availability varies among applications. Databases and the Internet have enabled worldwide collaboration and information sharing by extending the reach of database applications throughout organizations and communities. This reach emphasizes the importance of high availability in data management solutions. Both small businesses and global enterprises have users all over the world who require access to data 24 hours a day. Without this data access, operations can stop, and revenue is lost. Users now demand service-level agreements from their information technology (IT) departments and solution providers, reflecting the increasing dependence on these solutions. Increasingly, availability is measured in dollars, euros, and yen, not just in time and convenience.

Enterprises have used their IT infrastructure to provide a 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. If a critical application becomes unavailable, then the business can be in jeopardy. The business might lose revenue, incur penalties, and receive bad publicity that has a lasting effect on customers and on the company's stock price.

It is important to examine the factors that determine how your data is protected and maximize availability to your users.

1.3 Cost of Downtime

The need to deliver increasing levels of availability continues to accelerate as enterprises reengineer their solutions to gain competitive advantage. Most often, these new solutions rely on immediate access to critical business data. When data is not available, the operation can cease to function. Downtime can lead to lost productivity, lost revenue, damaged customer relationships, bad publicity, and lawsuits.

It is not always easy to place a direct cost on downtime. Angry customers, idle employees, and bad publicity are all costly, but not directly measured in currency. On the other hand, lost revenue and legal penalties incurred because SLA objectives are not met can easily be quantified. The cost of downtime can quickly grow in industries that are dependent on their solutions to provide service.

Other factors to consider in the cost of downtime are:

- The maximum tolerable length of a single unplanned outage

  If the event lasts less than 30 seconds, then it may cause very little impact and may be barely perceptible to users. As the length of the outage grows, the effect may grow exponentially and negatively affect the business.

- The maximum frequency of allowable incidents

  Frequent outages, even if short in duration, may similarly disrupt business operations.

When designing a solution, it is important to recognize the true cost of downtime to understand how the business can benefit from availability improvements.

Oracle provides a range of high availability solutions to fit every organization regardless of size. Small workgroups and global enterprises alike are able to extend the reach of their critical business applications. With Oracle and the Internet, applications and data are reliably accessible everywhere, at any time.

1.4 Causes of Downtime

One of the challenges in designing a high availability solution is examining and addressing all of the possible causes of downtime. It is important to consider causes of both unplanned and planned downtime when designing a fault-tolerant and resilient IT infrastructure. Planned downtime can be just as disruptive to operations as unplanned downtime, especially in global enterprises that support users in multiple time zones.

Table 1-1 (page 1-3) describes unplanned outage types and provides examples of each type.

Table 1-1 Causes of Unplanned Downtime

Type	Description	Examples
Site failure	A site failure may affect all processing at a data center, or a subset of applications supported by a data center.	Extended sitewide power failure Sitewide network failure Natural disaster makes a data center inoperable Terrorist or malicious attack on operations or the site

Table 1-1 (Cont.) Causes of Unplanned Downtime

Type	Description	Examples
Clusterwide failure	The whole cluster hosting an Oracle RAC database is unavailable or fails. This includes: • Failures of nodes in the cluster • Failure of any other components that result in the cluster being unavailable and the Oracle database and instances on the site being unavailable	The last surviving node on the Oracle RAC cluster fails and the node or database cannot be restarted Both redundant cluster interconnections fail or clusterware failure Database corruption so severe that continuity is not possible on the current database server Disk storage failure
Computer failure	A computer failure outage occurs when the system running the database becomes unavailable because it has failed or is no longer available. When the database uses Oracle RAC then a computer failure represents a subset of the system (while retaining full access to the data).	Database system hardware failure Operating system failure Oracle instance failure
Network failure	A network failure outage occurs when a network device stops or reduces network traffic and communication from your application to database, database to storage, or any system to system that is critical to your application service processing.	Network switch failure Network interface failure Network cable failures
Storage failure	A storage failure outage occurs when the storage holding some or all of the database contents becomes unavailable because it has shut down or is no longer available.	Disk drive failure Disk controller failure Storage array failure

Table 1-1 (Cont.) Causes of Unplanned Downtime

Type	Description	Examples
Data corruption	A corrupt block is a block that was changed so that it differs from what Oracle Database expects to find. Block corruptions can be categorized as physical or logical: • In a **physical block corruption**, which is also called a media corruption, the database does not recognize the block at all: the checksum is invalid or the block contains all zeros. An example of a more sophisticated block corruption is when the block header and footer do not match. • In a **logical block corruption**, the contents of the block are physically sound and pass the physical block checks; however, the block can be logically inconsistent. Examples of logical block corruption include incorrect block type, incorrect data or redo block sequence number, corruption of a row piece or index entry, or data dictionary corruptions. Block corruptions can also be divided into interblock corruption and intrablock corruption: • In **intrablock corruption**, the corruption occurs in the block itself and can be either a physical or a logical block corruption. • In an **interblock corruption**, the corruption occurs between blocks and can only be a logical block corruption. A data corruption outage occurs when a hardware, software, or network component causes corrupt data to be read or written. The service-level impact of a data corruption outage may vary, from a small portion of the application or database (down to a single database block) to a large portion of the application or database (making it essentially unusable).	Operating system or storage device driver failure Faulty host bus adapter Disk controller failure Volume manager error causing a bad disk read or write Software or hardware defects
Human error	A human error outage occurs when unintentional or other actions are committed that cause data in the database to become incorrect or unusable. The service-level impact of a human error outage can vary significantly, depending on the amount and critical nature of the affected data.	File deletion (at the file system level) Dropped database object Inadvertent data changes Malicious data changes

Table 1-1 (Cont.) Causes of Unplanned Downtime

Type	Description	Examples
Lost or stray writes	A lost or stray write is another form of data corruption, but it is much more difficult to detect and repair quickly. A data block stray or lost write occurs when: • For a **lost write**, an I/O subsystem acknowledges the completion of the block write even though the write I/O did not occur in the persistent storage. On a subsequent block read on the primary database, the I/O subsystem returns the stale version of the data block, which might be used to update other blocks of the database, thereby corrupting it. • For a **stray write**, the write I/O completed but it was written somewhere else, and a subsequent read operation returns the stale value. • For an Oracle RAC system, a read I/O from one cluster node returns stale data after a write I/O is completed from another node (lost write). For example, this occurs if a network file system (NFS) is mounted in Oracle RAC without disabling attribute caching (for example, without using the `noac` option). In this case, the write I/O from one node is not immediately visible to another node because it is cached. Block corruptions caused by stray writes or lost writes can cause havoc to your database availability. The data block may be physically or logically correct but subsequent disk reads will show blocks that are stale or with an incorrect Oracle Database block address.	Operating system or storage device driver failure Faulty host bus adapter Disk controller failure Volume manager error Other application software Lack of network file systems (NFS) write visibility across a cluster
Delay or slowdown	A delay or slowdown occurs when the database or the application cannot process transactions because of a resource or lock contention. A perceived delay can be caused by lack of system resources.	Database or application deadlocks Runaway processes that consume system resources Logon storms or system faults Combination of application peaks with lack of system or database resources. This can occur with one application or many applications in a consolidated database environment without proper resource management. Archived redo log destination or fast recovery area destination becomes full

Table 1-2 (page 1-7) describes planned outage types and provides examples of each type.

Table 1-2 Causes of Planned Downtime

Type	Description	Examples
System and database changes	Planned system changes occur when performing routine and periodic maintenance operations and new deployments. Planned system changes include any scheduled changes to the operating environment that occur outside of the organizational data structure in the database. The service-level impact of a planned system change varies significantly depending on the nature and scope of the planned outage, the testing and validation efforts made before implementing the change, and the technologies and features in place to minimize the impact.	Adding or removing processors to or from an SMP server Adding or removing nodes to or from a cluster Adding or removing disks drives or storage arrays Replacing any Field Replaceable Unit (FRU) Changing configuration parameters Upgrading or patching system hardware and software Upgrading or patching Oracle software Upgrading or patching application software System platform migration Database relocation Moving from 32 bits to 64 bits Migrating to cluster architecture Migrating to new storage
Data changes	Planned data changes occur when there are changes to the logical structure or physical organization of Oracle Database objects. The primary objective of these changes is to improve performance or manageability.	Table definition changes Adding table partitioning Creating and rebuilding indexes
Application changes	Planned application changes can include data changes and schema and programmatic changes. The primary objective of these changes is to improve performance, manageability, and functionality.	Application upgrades

Oracle offers high availability solutions to help avoid both unplanned and planned downtime, and recover from failures. Oracle Database High Availability Solutions for Unplanned Downtime (page 4-1) and Oracle Database High Availability Solutions for Planned Downtime (page 5-1) discuss each of these high availability solutions in detail.

1.5 Roadmap to Implementing the Maximum Availability Architecture

Oracle high availability solutions and sound operational practices are key to the successful implementation of an IT infrastructure. However, technology alone is not enough.

Choosing and implementing an architecture that best fits your availability requirements can be a daunting task. Maximum Availability Architecture (MAA)

simplifies the process of choosing and implementing a high availability architecture to fit your business requirements. The MAA:

- Encompasses redundancy across all components
- Provides protection and tolerance from computer failures, storage failures, human errors, data corruption, lost writes, system delays or slowdowns, and site disasters
- Recovers from outages as quickly and transparently as possible
- Provides solutions to eliminate or reduce planned downtime
- Provides consistent high performance and robust security
- Provides Oracle Engineered System options to simplify deployment and management and achieve higher scalability, performance, and availability
- Achieves SLAs at the lowest possible total cost of ownership
- Applies to On-Premise, Oracle Public Cloud, and hybrid architectures consisting of parts on-premise and part in the cloud
- Provides special consideration to Container or Oracle Multitenant, Oracle Database In-Memory, and Oracle Sharding architectures

To build, implement, and maintain this type of architecture, you need to:

1. Analyze your specific high availability requirements, including both the technical and operational aspects of your IT systems and business processes, as described in High Availability and Data Protection – Getting From Requirements to Architecture (page 2-1).
2. Familiarize yourself with Oracle high availability features, as described in Features for Maximizing Availability (page 3-1).
3. Understand the availability impact for each MAA tier or various high availability features on businesses and applications, as described in Oracle Database High Availability Solutions for Unplanned Downtime (page 4-1), and Oracle Database High Availability Solutions for Planned Downtime (page 5-1).
4. Use operational best practices to provide a successful MAA implementation, as described in Operational Prerequisites to Maximizing Availability (page 6-1).
5. Choose a high availability architecture, as described in High Availability Architectures (page 7-1).
6. Learn how Oracle's Engineered Systems such Oracle Exadata Database Machine, Oracle SuperCluster, Oracle Database Appliance, and Zero Data Loss Recovery Appliance improve MAA, as described in Oracle Engineered Systems (page 8-1).
7. Optimize your return on investment (ROI) of any of the high availability architectures and solutions, with Oracle Grid Computing, Oracle Active Data Guard real time reporting and utilization, Oracle Database multitenant architecture using pluggable databases, or Oracle Virtualization and Oracle Data Cloud, as described in Optimizing Return on Investment (page 9-1).
8. Implement a high availability architecture using the following resources:
    - MAA and high availability best practices white papers and other information

Oracle offers various best practices white papers, customer MAA papers with proof of concepts, customer case studies, recorded web casts, demonstrations, and presentations. These resources provide technical details about the MAA various high availability technologies, along with best practice recommendations for configuring and using such technologies.

These MAA resources are available at http://www.oracle.com/goto/maa

- *Oracle Database High Availability Best Practices*

  This book provides the configuration, repair and planned maintenance best practices for any of the MAA reference architectures. It can help you to configure a new high availability environment, or migrate an existing configuration to create a redundant, reliable system without sacrificing simplicity and performance.

2

High Availability and Data Protection – Getting From Requirements to Architecture

This chapter provides a framework to effectively evaluate the high availability requirements of an enterprise. It contains the following sections:

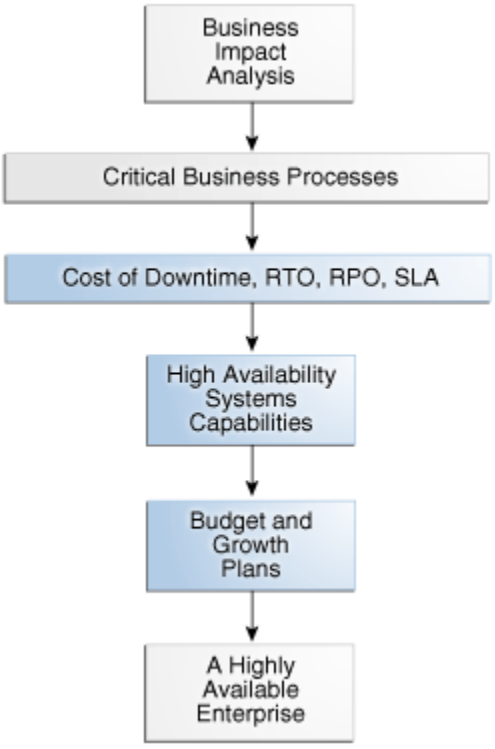
2.1 High Availability Requirements

Any effort to design and implement a high availability strategy for Oracle Database begins by performing a thorough business impact analysis to identify the consequences to the enterprise of downtime and data loss, whether caused by unplanned or planned outages. The term "business impact" is intended to be agnostic of whether the enterprise is a commercial venture, government agency, or not-for-profit institution. In all cases, data loss and downtime can seriously impact the ability of any enterprise to perform its function. Implementing high availability may involve critical tasks such as:

- Retiring legacy systems
- Investing in more capable and robust systems and facilities
- Redesigning the overall IT architecture and operations to adapt to this high availability model
- Modifying existing applications to take full advantage of high availability infrastructures
- Redesigning business processes
- Hiring and training personnel
- Moving parts or an entire application or database into the Oracle Public Cloud
- Balancing the right level of consolidation, flexibility, and isolation
- Understanding the capabilities and limitations of your existing system and network infrastructure

By combining your business analysis with an understanding of the level of investment required to implement different high availability solutions, you can develop a high availability architecture that achieves both business and technical objectives.

Figure 2-1 Planning and Implementing a Highly Available Enterprise



2.2 A Methodology for Documenting High Availability Requirements

The elements of this analysis framework are:

- Business Impact Analysis (page 2-2)
- Cost of Downtime (page 2-3)
- Recovery Time Objective (page 2-3)
- Recovery Point Objective (page 2-4)
- Manageability Goal (page 2-4)
- Total Cost of Ownership and Return on Investment (page 2-4)

2.2.1 Business Impact Analysis

A rigorous business impact analysis:

- Identifies the critical business processes in an organization
- Calculates the quantifiable loss risk for unplanned and planned IT outages affecting each of these business processes
- Outlines the effects of these outages
- Considers essential business functions, people and system resources, government regulations, and internal and external business dependencies
- Is based on objective and subjective data gathered from interviews with knowledgeable and experienced personnel

- Reviews business practice histories, financial reports, IT systems logs, and so on

The business impact analysis categorizes the business processes based on the severity of the impact of IT-related outages. For example, consider a semiconductor manufacturer with chip fabrication plants located worldwide. Semiconductor manufacturing is an intensely competitive business requiring a huge financial investment that is amortized over high production volumes. The human resource applications used by plant administration are unlikely to be considered as mission-critical as the applications that control the manufacturing process in the plant. Failure of the applications that support manufacturing affects production levels and have a direct impact on the financial results of the company.

Similarly, an internal knowledge management system is likely to be considered mission-critical for a management consulting firm, because the business of a client-focused company is based on internal research accessibility for its consultants and knowledge workers. The cost of downtime of such a system is extremely high for this business.

2.2.2 Cost of Downtime

A complete business impact analysis provides the insight needed to quantify the cost of unplanned and planned downtime. Understanding this cost is essential because it helps prioritize your high availability investment and directly influences the high availability technologies that you choose to minimize the downtime risk.

Various reports have been published, documenting the costs of downtime in different industries. Examples include costs that range from millions of dollars for each hour of brokerage operations and credit card sales, to tens of thousands of dollars for each hour of package shipping services.

These numbers are staggering. The Internet can connect the business directly to millions of customers. Application downtime can disrupt this connection, cutting off a business from its customers. In addition to lost revenue, downtime can negatively affect customer relationships, competitive advantages, legal obligations, industry reputation, and shareholder confidence.

2.2.3 Recovery Time Objective

The business impact analysis determines your tolerance to downtime, also known as recovery time objective (RTO). An RTO is defined as the maximum amount of time that an IT-based business process can be down before the organization starts suffering unacceptable consequences (financial losses, customer dissatisfaction, reputation, and so on). RTO indicates the downtime tolerance of a business process or an organization in general.

The RTO requirements are driven by the mission-critical nature of the business. Thus, for a system running a stock exchange, the RTO is zero or near to zero.

An organization is likely to have varying RTO requirements across its various business processes. Thus, for a high volume e-commerce website, for which there is an expectation of rapid response times and for which customer switching costs are very low, the web-based customer interaction system that drives e-commerce sales is likely to have an RTO of zero or close to zero. However, the RTO of the systems that support back-end operations, such as shipping and billing, can be higher. If these back-end systems are down, then the business may resort to manual operations temporarily without a significant visible impact.

The ability to take orders through the e-commerce website immediately (the RTO) may be more important than the RPO, because lost data can be reloaded later.

2.2.4 Recovery Point Objective

The business impact analysis also determines your tolerance to data loss, also known as recovery point objective (RPO). RPO is the maximum amount of data that an IT-based business process can lose without harm to the organization. RPO measures the data-loss tolerance of a business process or an organization in general. This data loss is often measured in terms of time, for example, zero, seconds, hours, or days of data loss.

A stock exchange where millions of dollars worth of transactions occur every minute cannot afford to lose any data. Thus, its RPO must be zero. The web-based sales system in the e-commerce example does not require an RPO of zero, although a low RPO is essential for customer satisfaction. However, its back-end merchandising and inventory update system can have a higher RPO because lost data can be reentered.

2.2.5 Manageability Goal

A **manageability goal** is more subjective than either the RPO or the RTO. It results from an objective evaluation of the skill sets, management resources, and tools available in an organization and the degree to which the organization can successfully manage all elements of a high availability architecture. Just as RPO and RTO measure an organization's tolerance for downtime or data loss, your manageability goal measures the organization's tolerance for complexity in the IT environment. When less complexity is a requirement, simpler methods of achieving high availability are preferred over methods that may be more complex to manage, even if the latter could attain more aggressive RTO and RPO objectives. Understanding manageability goals helps organizations differentiate between what is possible and what is practical to implement.

2.2.6 Total Cost of Ownership and Return on Investment

Understanding total cost of ownership (TCO) and objectives for return on investment (ROI) are essential to selecting a high availability architecture that also achieves the business goals of your organization. TCO includes all costs (such as acquisition, implementation, systems, networks, facilities, staff, training, and support), over the useful life of the solution chosen. Likewise, the ROI calculation captures all of the financial benefits that accrue to a given high availability architecture.

For example, consider a high availability architecture in which IT systems and storage at a remote standby site remain idle with no other business use that can be served by the standby systems. The only return on investment for the standby site is the costs related to downtime avoided by its use in a failover scenario. Contrast this with a different high availability architecture that enables IT systems and storage at the standby site to be used productively while in the standby role (for example, for reports or for off-loading the primary system of the overhead of user queries or distributing read-write workload). The return on investment of such an architecture includes both the cost of downtime avoided and the financial benefits that accrue to its productive use while it also provides for high availability and data protection.

2.3 Mapping Requirements to Architectures

The business impact analysis will document what you already know. Different applications and the databases that support them represent varying degrees of importance to the enterprise. A high level of investment in high availability

infrastructure may not make sense for an application that if down, would not have an immediate impact on the enterprise. So where do you start?

The outcome of the business impact analysis enables databases within an enterprise to be grouped together with other databases having similar RTO and RPO objectives. The groups can then be mapped to a controlled set of high availability reference architectures that most closely addresses the required service levels. Note that in the case where there are dependencies between databases, they are grouped with the database having the most stringent high availability requirement.

2.3.1 Oracle MAA Reference Architectures

Oracle MAA best practices define high availability reference architectures that address the complete range of availability and data protection required by enterprises of all sizes and lines of business.

The Platinum, Gold, Silver, and Bronze MAA reference architectures, or tiers, are applicable to on-premise, private and public cloud configurations, and hybrid cloud. They deliver the service levels described in the following figure.

Figure 2-2 Oracle MAA Reference Architectures

Each tier uses a different MAA reference architecture to deploy the optimal set of Oracle high availability capabilities that reliably achieve a given service level at the lowest cost and complexity. The tiers explicitly address all types of unplanned outages including data corruption, component failure, and system and site outages, as well as planned outages due to maintenance, migrations, or other purposes.

The Oracle Sharding reference architecture uses these same standard Bronze, Silver, Gold, and Platinum reference architectures as building blocks to provide shard-level high availability, given that each shard is a standalone Oracle Database. The Oracle Sharding reference architecture also includes best practices that address any unique considerations for a sharded database.

Container databases (CDBs) using Oracle Multitenant can exist in any tier, Bronze through Platinum, providing higher consolidation density and higher TCO. Typically, the consolidation density is higher with Bronze and Silver tiers, and there is less or zero consolidation when deploying a Platinum tier.

Oracle Database In-Memory can also be leveraged in any of the MAA tiers. Because the In-Memory column store is seamlessly integrated into Oracle Database, all of the high availability benefits that come from the MAA tiers are inherited when implementing Oracle Database In-Memory.

Oracle Engineered Systems can also exist in any of the tiers. Integrating Zero Data Loss Recovery Appliance (Recovery Appliance) as the Oracle Database backup solution for your entire data center reduces RPO and RTO when restoring from backups. Leveraging Oracle Exadata Database Machine as your database platform in the MAA reference architectures provide the best database platform solution with the lowest RTO and brownout.

See Also:

Introduction to MAA Reference Architectures (page 7-1)

Oracle Engineered Systems (page 8-1)

Oracle Database In-Memory High Availability Best Practices MAA white paper

2.3.2 Bronze Reference Architecture

The Bronze tier is appropriate for databases where simple restart or restore from backup is "HA enough." The Bronze tier is based upon a single instance Oracle Database with MAA best practices that use the many capabilities for data protection and high availability included with every Oracle Enterprise Edition license. Oracle-optimized backups using Oracle Recovery Manager (RMAN) provide data protection and are used to restore availability should an outage prevent the database from restarting.

2.3.3 Silver Reference Architecture

The Silver tier provides an additional level of high availability for databases that require minimal or zero downtime in the event of database instance or server failure, as well as many types of planned maintenance. The Silver tier adds clustering technology – either Oracle RAC or Oracle RAC One Node. RMAN provides database-optimized backups to protect data and restore availability should an outage prevent the cluster from restarting.

2.3.4 Gold Reference Architecture

The Gold tier raises the stakes substantially for business critical applications that cannot accept vulnerability to single points-of-failure. This tier adds database-aware replication technologies, Oracle Active Data Guard and Oracle GoldenGate, which synchronize one or more replicas of the production database to provide real time data protection and availability. Database-aware replication substantially enhances high availability and data protection beyond what is possible with storage replication technologies. It also reduces cost while improving return on investment by actively utilizing all replicas at all times.

2.3.5 Platinum Reference Architecture

The Platinum tier introduces several new Oracle Database 12*c* capabilities and previously available products that have been enhanced with the latest release. These capabilities include Application Continuity, for reliable replay of in-flight transactions that masks outages from users; Oracle Active Data Guard Far Sync, for zero data loss protection at any distance; Oracle GoldenGate enhancements for zero downtime upgrades and migrations; and Global Data Services for automated service management and workload balancing in replicated database environments. Each of

these technologies requires additional effort to implement, but they deliver substantial value for the most critical applications where downtime and data loss are not an option.

2.3.6 Oracle Sharding MAA Reference Architecture

Oracle Sharding distributes data and workloads across a pool of independent databases (shards) that are presented to the application as a single logical database, also known as a sharded database.

A sharded database is used to provide linear scalability and fault isolation for suitable applications. A sharded database eliminates the possibility of a single physical database being unable to scale to meet application requirements. Similarly, a sharded database prevents a physical database from being a single point of failure for an application due to unplanned outages or planned maintenance.

The Oracle Sharding reference architecture uses the standard Bronze, Silver, Gold, and Platinum reference architectures as building blocks to provide shard-level high availability given that each shard is a standalone Oracle Database. The Oracle Sharding reference architecture also includes best practices that address any unique considerations for a sharded database.

See Also:

Oracle Database Sharding Reference Architecture (page 7-19)

2.3.7 High Availability and Data Protection Attributes by Tier

Table 2-1 (page 2-7) summarizes the high availability and data protection attributes inherent to each tier. Each tier includes all of the capabilities of the previous tier and builds upon the architecture to handle an expanded fault domain. The various components included and the service levels achieved by each architecture are described in other topics.

Table 2-1 High Availability and Data Protection Attributes by Tier

Outage Class/HA Tier	Unplanned Outages (Local Site)	Planned Maintenance	Data Protection	Unrecoverable Local Outages and Disaster Recovery
Platinum	Zero application outage for Platinum ready applications	Zero application outage	Comprehensive runtime validation combined with manual checks	Zero application outage for Platinum ready applications, in-flight transactions are preserved, zero data loss
Gold	Comprehensive HA and DR	All rolling or online	Comprehensive runtime validation combined with manual checks	Real-time failover, zero or near-zero data loss

Table 2-1 (Cont.) High Availability and Data Protection Attributes by Tier

Outage Class/HA Tier	Unplanned Outages (Local Site)	Planned Maintenance	Data Protection	Unrecoverable Local Outages and Disaster Recovery
Silver	HA with automatic failover	Some rolling, some online, some offline	Basic runtime validation combined with manual checks	Restore from backup, potential to lose data generated since last backup. If the Recovery Appliance is present for both silver and bronze, potential to lose data is zero or near zero..
Bronze	Single Instance, Auto restart for recoverable instance and server failures	Some online, most off-line	Basic runtime validation combined with manual checks	Restore from backup, potential to lose data generated since last backup
Oracle Sharding	Failure impact isolated to the shard enabling highest application availability. For each shard failure, very low application brownout or zero application outage for Platinum Ready applications.	Planned maintenance impact isolated to the shard enabling highest application availability. For shard maintenance, all rolling or online or zero application outage	Each shard is isolated. Each shard has comprehensive runtime validation combined with manual checks	For site failure, impact on application is dependent on active shards in failed site vs. total number of active shards. Each shard can be configured with real time failover, zero or near zero data loss, or zero application outage for Platinum ready applications, in-flight transactions are preserved, zero data loss

3

Features for Maximizing Availability

This chapter describes the Oracle Database features used in MAA solutions.

See Also:

- The overview of high availability in *Oracle Database Concepts*
- The list of new high availability features in *Oracle Database New Features Guide*

3.1 Oracle Data Guard

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data. Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable Oracle databases to survive outages of any kind, including natural disasters and data corruptions. A Data Guard standby database is an exact replica of the production database and thus can be transparently utilized in combination with traditional backup, restoration, flashback, and cluster techniques to provide the highest possible level of data protection and data availability. Data Guard is included in Oracle Enterprise Edition.

A Data Guard configuration consists of one primary database and one or more standby databases. A primary database can be either a single-instance Oracle database or an Oracle RAC database. Similar to a primary database, a standby database can be either a single-instance Oracle database or an Oracle RAC database. Using a backup copy of the primary database, you can create up to 30 standby databases that receive redo directly from the primary database. Optionally you can use a cascaded standby to create Data Guard configurations where the primary transmits redo to a single remote destination, and that destination forwards redo to multiple standby databases. This enables a primary database to efficiently synchronize many more than 30 standby databases if desired.

Note:

Oracle Active Data Guard is an extension of basic Data Guard providing advanced features that off-load various types of processing from a production database, extend zero data loss protection over any distance, and that enhance high availability. Oracle Active Data Guard is licensed separately from Oracle Database Enterprise Edition. Oracle Active Data Guard is discussed more completely in Oracle Active Data Guard (page 3-5).

There are several types of standby databases. Data Guard physical standby database is the MAA best practice for data protection and disaster recovery and is the most common type of standby database used. A physical standby database uses Redo Apply (an extension of Oracle media recovery) to maintain an exact, physical replica of the production database. When configured using MAA best practices, Redo Apply uses multiple Oracle-aware validation checks to prevent corruptions that can impact a primary database from impacting the standby. Other types of Data Guard standby databases include: snapshot standby (a standby open read/write for test or other purposes) and logical standby (used to reduce planned downtime).

Benefits of Using Data Guard

- Continuous Oracle-aware validation of all changes using multiple checks for physical and logical consistency of structures within an Oracle data block and redo, before updates are applied to a standby database. This isolates the standby

database and prevents it from being impacted by data corruptions that can occur on the primary system.

- Transparent operation: There are no restrictions on the use of Data Guard physical standby for data protection. Redo Apply supports all data and storage types, all DDL operations, and all applications (custom and packaged applications), and guarantees data consistency across primary and standby databases.
- Highest performance: Fast redo transport for best recovery point objective, fast apply performance for best recovery time objective. With Oracle Database 12*c* Release 2, multi-instance redo apply provides Oracle RAC scalability for redo apply, eliminating bottlenecks of a single database server. Redo apply can essentially scale up to available CPU, I/O, and network across your Oracle RAC cluster. An observed redo apply rate of 3500 MB per second (12 TB/hour) on 8 node RAC Exadata.
- Fast failover to a standby database to maintain availability should the primary database fail for any reason. Failover is either a manual or automatic operation depending on how Data Guard is configured.
- Integrated client notification framework to enable application clients to connect to a new primary database after a failover occurs.
- Automatic or automated (depending upon configuration) resynchronization of a failed primary database, quickly converting it to a synchronized standby database after a failover occurs.
- Choice of flexible data protection levels to support all network configurations, availability and performance SLAs, and business requirements.
- Management of a primary and all of its standby databases as a single configuration to simplify management and monitoring using either the Data Guard Broker command-line interface or Oracle Enterprise Manager Cloud Control.
- Data Guard Broker 12*c* greatly improves manageability with additional features for comprehensive configuration health checks, resumable switchover operations, streamlined role transitions, support for cascaded standby configurations, and user-configurable thresholds for transport and apply lag to automatically monitor the ability of the configuration to support SLAs for recovery point and recovery time objectives at any instant in time.
- Efficient transport to multiple remote destinations using a single redo stream originating from the primary production database and forwarded by a cascading standby database.
- Snapshot Standby enables a physical standby database to be open read/write for testing or any activity that requires a read/write replica of production data. A snapshot standby continues to receive but does not apply updates generated by the primary. When testing is complete, a snapshot standby is converted back into a synchronized physical standby database by first discarding the changes made during the open read/write, and then applying the redo received from the primary database. Primary data is always protected. Snapshot standby is particularly useful when used in conjunction with Oracle Real Application Testing (workload is captured at the production database for replay and subsequent performance analysis at the standby database-an exact replica of production).
- Reduction of planned downtime by utilizing a standby database to perform maintenance in rolling fashion. The only downtime is the time required to perform

a Data Guard switchover; applications remain available while the maintenance is being performed. (See When to Use Oracle Active Data Guard and Oracle GoldenGate Together (page 3-16) and Table 5-7 (page 5-18) for more details).

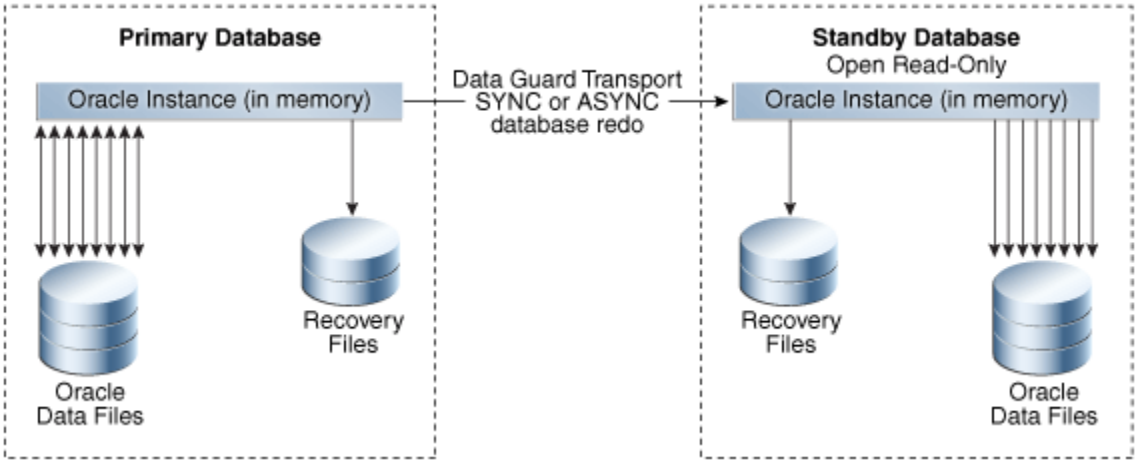
- Increased flexibility for Data Guard configurations where the primary and standby systems may have different CPU architectures or operating systems subject to limitations defined in My Oracle Support note 413484.1.
- Efficient disaster recovery for a container database (CDB). Data Guard failover and switchover completes using a single command at a CDB level regardless of how many pluggable databases (PDBs) are consolidated within the CDB.
- Enables a specific administration privilege, SYSDG, to handle standard administration duties for Data Guard. This new privilege is based on the least privilege principle, in which a user is granted only the necessary privileges required to perform a specific function and no more. The SYSDBA privilege continues to work as in previous releases.
- The Oracle Database In-Memory column store is supported on standby databases in an Active Data Guard environment. (new in Oracle 12*c* Release 2)
- Further improves performance and availability of Data Warehouses in a Data Guard configuration by tracking information from `NOLOGGING` operations so they can be repaired with the new RMAN command `RECOVER DATABASE NOLOGGING`. (new in Oracle 12*c* Release 2)
- Improves the impact multiple SYNC transport destinations have on the primary database through the use of a new parameter `DATA_GUARD_SYNC_LATENCY`. This parameter defines the maximum amount of time (in seconds) that the Primary database must wait before disconnecting subsequent destinations after at least one synchronous standby has acknowledged receipt of the redo. (new in Oracle 12*c* Release 2)
- Data Guard Broker improves manageability by supporting destinations of different Endianess than the primary in addition to enhancing management of alternate destinations. (new in Oracle 12*c* Release 2)
- Data Guard improves protection and Return To Operations (RTO) and Recovery Point Objectives (RPO) through multiple features (new in Oracle 12*c* Release 2) including:
  - Multi Instance Redo Apply (MIRA) provides scalable redo apply performance across Oracle RAC instances reducing RTO for even higher production OLTP or batch workloads
  - Compare primary and standby database blocks using the new `DBMS_DBCOMP` package to help identify lost writes so they can be resolved efficiently.
  - Fast Start Failover (FSFO) has the robustness of highly available zero data loss configurations with support for Maximum Protection mode while giving the flexibility of multiple observers and multiple failover targets for high availability in any configuration. FSFO can also be configured to automatically fail over to the standby with the detection of a lost write on the primary .
  - RPO is improved with no data loss failovers after a storage failure in ASYNC configurations and Data Guard Broker support for Application Continuity, improving the user experience during Data Guard role transitions.

- Oracle Data Guard Broker further improves the management of databases by supporting destinations of different endianness than the primary in addition to enhancing management of alternate archive destinations when the primary destination is unavailable.

3.1.1 Oracle Active Data Guard

Oracle Active Data Guard is Oracle's strategic solution for real time data protection and disaster recovery for the Oracle database using a physical replication process. Oracle Active Data Guard also provides high return on investment in disaster recovery systems by enabling a standby database to be open read-only while it applies changes received from the primary database. Oracle Active Data Guard is a separately licensed product that provides advanced features that greatly expand Data Guard capabilities included with Oracle Enterprise Edition.

Figure 3-1 Oracle Active Data Guard Architecture



Oracle Active Data Guard enables administrators to improve performance by offloading processing from the primary database to a physical standby database that is open read-only while it applies updates received from the primary database. Offload capabilities of Oracle Active Data Guard 12*c* were enhanced to include read-only reporting and ad-hoc queries (including DML to global temporary tables and unique global or session sequences), data extracts, fast incremental backups, redo transport compression, efficient servicing of multiple remote destinations, and the ability to extend zero data loss protection to a remote standby database without impacting primary database performance. Oracle Active Data Guard also increases high availability by performing automatic block repair and enabling High Availability Upgrades (new automation in Oracle Database 12*c* for more easily implementing database rolling upgrades).

Note:

Oracle Active Data Guard is licensed separately as a database option license for Oracle Database Enterprise Edition. All Oracle Active Data Guard capabilities are also included in an Oracle Golden Gate license for Oracle Enterprise Edition. This provides customers with the choice of a standalone license for Oracle Active Data Guard, or licensing Oracle GoldenGate to acquire access to all advanced Oracle replication capabilities.

Benefits of Oracle Active Data Guard

Oracle Active Data Guard inherits all of the benefits previously listed for Data Guard, plus the following:

- Improves primary database performance: Production-offload to an Oracle Active Data Guard standby database of read-only applications, reporting, and ad hoc queries. Any application compatible with a read-only database can run on an Oracle Active Data Guard standby. Oracle also provides integration that enables the offloading of many Oracle E-Business Suite Reports, PeopleTools reporting, Oracle Business Intelligence Enterprise Edition (OBIEE), and Oracle TopLink applications to an Oracle Active Data Guard standby database.
- Oracle Active Data Guard 12*c* provides new support for DML to global temporary tables and the use of sequences at the standby database. This significantly expands the number of read-only applications that can be off-loaded from production databases to an Oracle Active Data Guard standby database.
- The unique ability to easily scale read performance using multiple Oracle Active Data Guard standby databases, also referred to as a Reader Farm.
- Production-offload of data extracts using Oracle Data Pump or other methods that read directly from the source database.
- Production-offload of the performance impact from network latency in a synchronous, zero data loss configuration where primary and standby databases are separated by hundreds or thousands of miles. Oracle Active Data Guard 12*c* far sync utilizes a lightweight instance (control file and archive log files, but no recovery and no data files), deployed on a system independent of the primary database. The far sync instance is ideally located at the maximum distance from the primary system that an application can tolerate the performance impact of synchronous transport to provide optimal protection. Data Guard transmits redo synchronously to the far sync instance and far sync forwards the redo asynchronously to a remote standby database that is the ultimate failover target. If the primary database fails, the same failover command used for any Data Guard configuration, or mouse click using Oracle Enterprise Manager Cloud Control, or automatic failover using Data Guard Fast-Start Failover executes a zero data loss failover to the remote destination. This transparently extends zero data loss protection to a remote standby database just as if it were receiving redo directly from the primary database, while avoiding the performance impact to the primary database of WAN network latency in a synchronous configuration.
- Production-offload of the overhead of servicing multiple remote standby destinations using far sync. In a far sync configuration, the primary database ships a single stream of redo to a far sync instance using synchronous or asynchronous transport. The far sync instance is able to forward redo asynchronously to as many as 29 remote destinations with zero incremental overhead on the source database.
- Data Guard maximum availability supports the use of the `NOAFFIRM` redo transport attribute. A standby database returns receipt acknowledgment to its primary database as soon as redo is received in memory. The standby database does not wait for the Remote File Server (RFS) to write to a standby redo log file.

  This feature provides increased primary database performance in Data Guard configurations using maximum availability and SYNC redo transport. Fast Sync isolates the primary database in a maximum availability configuration from any performance impact due to slow I/O at a standby database. This new FAST SYNC feature can work with a physical standby target or within a far sync configuration.

- Production-offload of CPU cycles required to perform redo transport compression. Redo transport compression can be performed by the far sync instance if the Data Guard configuration is licensed for Oracle Advanced Compression. This conserves bandwidth with zero incremental overhead on the primary database.
- Production-offload and increased backup performance by moving fast incremental backups off of the primary database and to the standby database by utilizing Oracle Active Data Guard support for RMAN block change tracking.
- Increased high availability using Oracle Active Data Guard automatic block repair to repair block corruptions, including file header corruptions, detected at either the primary or standby, transparent to applications and users.
- Increased high availability by reducing planned downtime for upgrading to new Oracle Database patch sets and database releases using the additional automation provided by high availability Upgrade, new with Oracle Active Data Guard 12*c*
- Connection preservation on an Active Data Guard standby through a role change facilitates improved reporting and improves the user experience. The connections pause while the database role changes to a primary database and resume, improving the user experience.
- The Oracle Enterprise Manager Diagnostic tool can be used with Active Data Guard to capture and send performance data to the Automatic Workload Repository, while the SQL Tuning Advisor allows primary database SQL statement tuning to be offloaded to a standby database.
- Active Data Guard support for the Oracle Database In-Memory option enables reporting to be offloaded to the standby database while reaping the benefits the In-Memory option provides, including tailored column stores for the standby database workload.

See Also:

Rolling Upgrade Using Oracle Active Data Guard (page 3-10)

3.1.2 Data Guard Advantages Over Traditional Solutions

Data Guard provides a number of advantages over traditional solutions, including the following:

- Fast, automatic or automated database failover for data corruptions, lost writes, and database and site failures, with recovery times of potentially seconds with Data Guard as opposed to hours with traditional solutions
- Zero data loss over wide area network using Oracle Active Data Guard Far Sync
- Offload processing for redo transport compression and redo transmission to up to 29 remote destinations using Oracle Active Data Guard Far Sync
- Automatic corruption repair automatically replaces a physical block corruption on the primary or physical standby by copying a good block from a physical standby or primary database
- Most comprehensive protection against data corruptions and lost writes on the primary database

- Reduced downtime for storage, Oracle ASM, Oracle RAC, system migrations and some platform migrations, and changes using Data Guard switchover
- Reduced downtime with Data Guard rolling upgrade capabilities
- Ability to off-load primary database activities—such as backups, queries, or reporting—without sacrificing the RTO and RPO ability to use the standby database as a read-only resource using the real-time query apply lag capability, including Database In-Memory column support in Oracle Database 12*c* Release 2
- Ability to integrate non-database files using Oracle Database File System (DBFS) or Oracle Automatic Storage Management Cluster File System (Oracle ACFS) as part of the full site failover operations (see Oracle Replication Technologies for Non-Database Files (page 3-38))
- No need for instance restart, storage remastering, or application reconnections after site failures
- Transparency to applications
- Transparent and integrated support (application continuity and transaction guard) for application failover
- Effective network utilization
- Database In-Memory support
- Integrated service and client failover that reduces overall application RTO
- Enhanced and integrated Data Guard awareness with existing Oracle technologies such as Oracle RAC, RMAN, Oracle GoldenGate, Enterprise Manager, health check (orachk), DBCA.

For data resident in Oracle databases, Data Guard, with its built-in zero-data-loss capability, is more efficient, less expensive, and better optimized for data protection and disaster recovery than traditional remote mirroring solutions. Data Guard provides a compelling set of technical and business reasons that justify its adoption as the disaster recovery and data protection technology of choice, over traditional remote mirroring solutions.

3.1.3 Data Guard and Planned Maintenance

Data Guard standby databases can be used to reduce planned downtime by performing maintenance in a rolling fashion. Changes are implemented first at the standby database. The configuration is allowed to run with the primary at the old version and standby at the new version until there is confidence that the new version is ready for production. A Data Guard switchover is then performed, transitioning production to the new version. The only database downtime is the time required to perform the switchover.

There are several approaches to performing maintenance in a rolling fashion using a Data Guard standby. Customer requirements and preferences determine which approach is used. The following approaches are discussed in this document:

- Data Guard Redo Apply and Standby-First Patching (page 3-9)
- Data Guard Transient Logical Rolling Upgrades (page 3-9)
- Rolling Upgrade Using Oracle Active Data Guard (page 3-10)

3.1.3.1 Data Guard Redo Apply and Standby-First Patching

Beginning with Oracle Database 10*g*, there has been increased flexibility in cross-platform support using Data Guard Redo Apply. In certain Data Guard configurations, primary and standby databases are able to run on systems having different operating systems (for example, Windows and Linux), word size (32bit/64bit), or hardware architectures. Redo Apply can also be used to migrate to Oracle Automatic Storage Management (ASM), to move from single instance Oracle databases to Oracle RAC, to perform technology refresh, or to move from one data center to the next.

Beginning with Oracle Database 11*g* Release 2 (11.2), Standby-First Patch Apply (physical standby using Redo Apply) can support different software patch levels between a primary database and its physical standby database for the purpose of applying and validating Oracle patches in a rolling fashion. Patches eligible for Standby-First patching include:

- Database Patch Set Update (PSU)
- Database Critical Patch Update (CPU)
- Database bundled patch
- Oracle Exadata Database Machine bundled patch
- Exadata Storage Server Software patch
- Any operating system, system firmware, or system changes compatible with the existing Oracle database version

Standby-First Patch Apply is supported for certified software patches for Oracle Database Enterprise Edition 11*g* Release 2 (11.2) and later.

In each of the types of planned maintenance previously described, the configuration begins with a primary and physical standby database (in the case of migration to a new platform, or to ASM or Oracle RAC, the standby is created on the new platform). After all changes are implemented at the physical standby database, Redo Apply (physical replication) is used to synchronize the standby with the primary. A Data Guard switchover is used to transfer production to the standby (the new environment).

See Also:

- My Oracle Support Note 413484.1 for information about mixed platform combinations supported in a Data Guard configuration.
- My Oracle Support Note 1265700.1 for more information about Standby First Patch Apply and the README for each patch to determine if a target patch is certified as being a Standby-First Patch.

3.1.3.2 Data Guard Transient Logical Rolling Upgrades

There are numerous types of maintenance tasks that are unable to use Redo Apply (physical replication) to synchronize the original version of a database with the changed or upgraded version. These tasks include:

- Database patches or upgrades that are not Standby-First Patch Apply-eligible. This includes database patch-sets (11.2.0.2 to 11.2.0.4) and upgrade to new Oracle Database releases (11.2.0.4 to 12.1.0.1 or 12.2).
- Maintenance must be performed that modifies the physical structure of a database that would require downtime (for example, adding partitioning to non-partitioned tables, changing Basicfile LOBs to Securefile LOBs, changing XML-CLOB to Binary XML, or altering a table to be OLTP-compressed).

All of the previous types of maintenance can be performed in a rolling fashion using a Data Guard standby database by using Data Guard SQL Apply (logical replication) to synchronize the old and new versions of the database. Prior to Oracle Database 11*g* this required creating a logical standby database, performing the maintenance on the logical standby, resynchronizing the standby with the primary, and then switching over. Additionally if a physical standby was being used for disaster recovery, then a new physical standby database would have to be created from a backup of the production database at the new version. This represented a number of logistical and cost challenges when upgrading a multi-terabyte database.

Beginning with Oracle Database 11*g*, database rolling upgrades can use a new procedure called Transient Logical that begins and ends with a physical standby database. SQL Apply is only used during the phase when Data Guard is synchronizing across old and new versions. A new logical standby database does not need to be created if there is already a physical standby in place. A new physical standby database does not need to be created from a backup of the production database at the new version after the maintenance is complete. Similar to the traditional process of upgrading a Data Guard configuration having an in-place physical standby, the original primary is upgraded or changed using redo from the new primary database and Redo Apply (a single catalog upgrade migrates both primary and standby databases to the new Oracle release).

Transient Logical upgrades require that the primary database be at Oracle Database 11*g* release 1 (11.1) or later and that the database meet the pre-requisites of SQL Apply.

Oracle provides a Bourne shell script that automates a number of the manual steps required by the Transient Logical rolling upgrade process.

Databases that use Oracle Database Vault can be upgraded to new Oracle Database releases and patch sets by using Oracle Data Guard database rolling upgrades (transient logical standby only).

See Also:

Oracle MAA white paper Oracle Database Rolling Upgrades: Using a Data Guard Physical Standby Database

---

3.1.3.3 Rolling Upgrade Using Oracle Active Data Guard

Oracle Database 12*c* introduces rolling upgrade using Oracle Active Data Guard to provide a simpler, automated, and easily repeatable method for reducing planned downtime than represented by the manual Transient Logical rolling upgrade procedure previously described. Rolling upgrade using Oracle Active Data Guard transforms the 42 or more steps required by the manual procedure into several easy-to-use DBMS_ROLLING PL/SQL packages. Rolling upgrades performed using the DBMS_ROLLING PL/SQL package are supported on a multitenant container database (CDB).

A rolling upgrade using Oracle Active Data Guard uses the following steps:

- Call `DBMS_ROLLING.INIT_PLAN`
  - Generates an upgrade plan with a configuration-specific set of instructions to guide the administrator through the upgrade process
- Call `DBMS_ROLLING.SET_PARAMETER`
  - Modifies parameters of the rolling upgrade
- Install new software at all databases participating in the upgrade
- Call `DBMS_ROLLING.START_PLAN`
  - Configures primary and standby databases participating in the upgrade
- Upgrade or make changes to the standby database
- Call `DBMS_ROLLING.SWITCHOVER`
  - Switchover moves the production to the new version
  - Switchover is the only downtime required
- Restart former primary using new binaries if appropriate
- Call `DBMS_ROLLING.FINISH_PLAN`
  - Completes the upgrade of the old primary and any additional standby databases in the Data Guard configuration and resynchronizes with the new primary

Rolling upgrade using Oracle Active Data Guard has the following benefits:

- Provides a simple specify-compile-execute protocol
  - Catches configuration errors at the compilation step
  - Runtime errors are detected during execution
- The state is kept in the database
  - Enables a reliable, repeatable process
- Runtime steps are constant regardless of how many databases are involved
- Handles failure at the original primary database
- Enables data protection for the upgraded primary at all times

Rolling upgrade using Oracle Active Data Guard requires an Oracle Active Data Guard license, the primary database be at Oracle Database 12*c* Release 1 (12.1) or later, and that the database satisfy prerequisites of SQL Apply. If the primary database is on an earlier Oracle Database release, use a Data Guard physical standby database.

See Also:

Oracle MAA white paper Oracle Database Rolling Upgrades: Using a Data Guard Physical Standby Database

Oracle Data Guard Concepts and Administration

3.2 Oracle GoldenGate

Oracle GoldenGate is Oracle's strategic logical replication solution for data distribution and data integration. Oracle GoldenGate offers a real-time, log-based change data capture and replication software platform. The software provides capture, routing, transformation, and delivery of transactional data across heterogeneous databases in real time.

Unlike replication solutions from other vendors, Oracle GoldenGate is more closely integrated with Oracle Database while also providing an open, modular architecture ideal for replication across heterogeneous database management systems. This combination of attributes eliminates compromise, making Oracle GoldenGate the preferred replication solution for addressing requirements that span Oracle Database and non-Oracle Database environments.

A typical environment includes a capture, pump, and delivery process. Each of these processes can run on most of the popular operating systems and databases, including Oracle Database and non-Oracle databases. All or a portion of the data can be replicated, and the data within any of these processes can be manipulated for not only heterogeneous environments but also different database schemas, table names, or table structures. Oracle GoldenGate also supports bidirectional replication with preconfigured conflict detection and resolution handlers to aid in resolving data conflicts.

3.2.1 Oracle GoldenGate 12*c*

Oracle GoldenGate 12*c* Release 2 offers significant new features that greatly enhance its replication capabilities and integration with Oracle Database. The new features include:

- End-to-end replication lag provides end-to-end replication lag views without requiring you to manually implement tables that must be continually updated. New commands are available to simplify this replication configuration and provide extra features including:
  - Unidirectional lag from source to target
  - Bidirectional lag when you set up an active-active replication that provides both incoming and outgoing lag
  - The `GG_LAG` database view to view the end-to-end lag information
- Automated remote trail file recovery by pump automatically handles when a target system is restored to a previous point in time. This feature also handles most cases where target trail files are inadvertently deleted or corrupted by automatically regenerating the missing target trail data when the source trail data is available, and by intelligently skipping any duplicate transactions when applying the change data.

- You can continue using GGSCI to start and stop the manager when GoldenGate processes are under Oracle Grid Infrastructure Agents (XAG) management with `XAGENABLE`.
- If Extract is configured in a downstream deployment, where redo is shipped from the source database to be mined on the downstream database, it is possible to fetch any required data from an active standby database instead of using the source database. Fetching is done when Extract is unable to reconstruct an update operation from the redo data, or when a `FETCHCOLS` clause is specified as part of the TABLE parameter.
- A new Extract parameter, `TRANLOGOPTIONS HANDLEDLFAILOVER`, only permits extract from redo data that has been applied to the Oracle Data Guard standby. When an Oracle GoldenGate source database, where integrated Extract is connected, is protected by an active standby database, where there is a potential for data loss (Data Guard Maximum Performance Mode using ASYNC redo transport), it is important to ensure Extract will not extract redo data that has not yet been applied to the standby database. Doing so leads to logical data inconsistencies in the event of a data loss failover because the Oracle GoldenGate target database will contain data that is missing from the source database.

See Also:

Administering Oracle GoldenGate for Windows and UNIX for information about monitoring replication lag

Fusion Middleware Reference for Oracle GoldenGate for Windows and UNIX for information about `XAGENABLE`

Fusion Middleware Reference for Oracle GoldenGate for Windows and UNIX for information about `FETCHUSERID` and `FETCHUSEDIDALIAS`

Transparent Role Transitions With Oracle Data Guard and Oracle GoldenGate MAA white paper

---

3.2.2 Oracle GoldenGate and Maximum Availability Architecture

Oracle GoldenGate logical replication enables all databases in an Oracle GoldenGate configuration, both source and target databases, to be open read-write. This makes it a key component of MAA for addressing a broad range of high availability challenges for zero downtime maintenance, cross platform migration, and continuous data availability, specifically:

- Zero or near zero downtime maintenance. In this architecture, Oracle GoldenGate provides greater flexibility than the capabilities provided by Data Guard. Oracle GoldenGate source and target databases can have a different physical and logical structure, can reside on different hardware and operating system architectures, can span wide differences in Oracle Database releases (for example, 9*i* to 12*c*), or be a mix of Oracle and non-Oracle systems. This allows for the modernization of 24x7 servers and allows new Oracle features to be implemented without impacting the availability of the databases. Maintenance is first performed on a target database while production runs on the source. After the maintenance is complete, production can be moved to the source all at once, similar to a Data Guard switchover. Optionally, bidirectional replication can be used to gradually move users over to the new system to create the perception of zero downtime. In either case, Oracle GoldenGate replication can be enabled in the reverse direction to keep

the original source database synchronized during a transition period, making it simple to effect a planned fall-back to the previous version if needed, with minimal downtime and no data loss.

- Zero or near-zero downtime migrations when a Data Guard solution is not applicable. Platform or database migrations can be carried out using Oracle GoldenGate as the data synchronization method between the old and new systems. Once the database has been instantiated on another host, Oracle GoldenGate is configured to replicate changes from the production database. A guaranteed restore point can be created on the migrated database so that after user testing the database can be flashed back, and Oracle GoldenGate can apply any outstanding data changes from the production database before moving the application users to the new database, similar to a snapshot standby database. If desired, bi-directional replication can also be configured from the migrated database back to the production database for use as a fallback solution.
- Zero or near-zero downtime application upgrades. Application upgrades that modify back-end database objects typically result in significant planned downtime while maintenance is being performed. Oracle GoldenGate replication enables data transformations that map database objects used by a previous version of an application to objects modified by the new version of an application. This enables database maintenance to be performed on a separate copy of the production database without impacting the availability of the application. After the maintenance is complete and Oracle GoldenGate has finished synchronizing old and new versions, users can be switched to the new version of the application.
- Oracle GoldenGate enables read-write access to a replica database while it is being synchronized with its source database. This is most often used to offload reporting to a copy of a production database when the reporting application requires a read-write connection to database in order to function. This is also relevant to disaster recovery environments where the nature of the technology used for the application tier requires an active read-write connection to the DR database at all times in order to meet recovery time objectives.
- Active-Active replication. Oracle GoldenGate supports an active-active multi-directional configuration, where there are two or more systems with identical sets of data that can be changed by application users on either system. Oracle GoldenGate replicates transactional data changes from each database to the others to keep all sets of data current.

See Also:

Oracle GoldenGate Documentation

3.2.3 Oracle GoldenGate with Oracle Real Application Clusters

When using Oracle Real Application Clusters (RAC), Oracle GoldenGate can be configured so that it seamlessly moves between Oracle RAC nodes in the event of database instance failure or during applicable maintenance operations.

This ability provides high availability with Oracle GoldenGate and it is possible to patch and upgrade the Oracle GoldenGate software on one or more nodes in the cluster without affecting the node where Oracle GoldenGate is currently running. Then at a predetermined time, Oracle GoldenGate can be switched to one of the

upgraded nodes. The switch is done without reconfiguring Oracle GoldenGate because configuration information is shared across the Oracle RAC cluster.

See Also:

Oracle GoldenGate with Oracle Real Application Clusters Configuration MAA white paper for information on about configuring Oracle GoldenGate in an Oracle RAC configuration

3.3 Best Practice: Oracle Active Data Guard and Oracle GoldenGate

While Oracle Active Data Guard and Oracle GoldenGate are each capable of maintaining a synchronized copy of an Oracle database, each has unique characteristics that result in high availability architectures that can use one technology or the other, or both at the same time, depending upon requirements. Examples of MAA Best Practice guidelines for use cases relevant to Oracle Database 12*c* are as follows:

3.3.1 When to Use Oracle Active Data Guard

Use Oracle Active Data Guard when the emphasis is on simplicity, data protection, and availability:

- Simplest, fastest, one-way replication of a complete Oracle database.
- No restrictions: Data Guard Redo Apply supports all data and storage types and Oracle features; transparent replication of DDL
- Features optimized for data protection: Detects silent corruptions that can occur on source or target; automatically repairs corrupt blocks
- Synchronized standby open read-only provides simple read-only offloading for maximum ROI
- Transparency of backups: A Data Guard primary and standby are physically exact copies of each other; RMAN backups are completely interchangeable
- Zero data loss protection at any distance, without impacting database performance
- Minimizing planned downtime and risk using standby first patching, database rolling upgrades, and select platform migrations
- Reduce risk of introducing change by dual purposing a DR system for testing using Data Guard Snapshot Standby
- Integrated automatic database and client failover
- Integrated management of a complete configuration: Data Guard Broker command line interface or Oracle Enterprise Manager Cloud Control

3.3.2 When to Use Oracle GoldenGate

Use Oracle GoldenGate when the emphasis is on advanced replication requirements not addressed by Oracle Active Data Guard:

- Any requirement where the replica database must be open read/write while synchronizing with the primary database

- Any data replication requirements such as multimaster and bidirectional replication, subset replication, many-to-one replication, and data transformations.
- When data replication is required between endian format platforms or across-database major versions.
- Maintenance and migrations where zero downtime or near zero downtime is required. Oracle GoldenGate can be used to migrate between application versions, for example, from Application 1.0 to Application 2.0 without downtime.
- Database rolling upgrades where it is desired to replicate from new version down to the old version for the purpose of fast fall-back if something is wrong with the upgrade.
- Zero downtime planned maintenance where bidirectional replication is used to gradually migrate users to the new version, creating the perception of zero downtime. Note that bidirectional replication requires avoiding or resolving update conflicts that can occur on disparate databases.

3.3.3 When to Use Oracle Active Data Guard and Oracle GoldenGate Together

Oracle Active Data Guard and Oracle GoldenGate are not mutually exclusive. The following are use cases of high availability architectures that include the simultaneous use of Oracle Active Data Guard and Oracle GoldenGate:

- An Oracle Active Data Guard standby is utilized for disaster protection and database rolling upgrades for a mission critical OLTP database. At the same time, Oracle GoldenGate is used to replicate data from the Data Guard primary database (or from the standby database using Oracle GoldenGate ALO mode) for ETL update of an enterprise data warehouse.
- Oracle GoldenGate subset replication is used to create an operational data store (ODS) that extracts, transforms, and aggregates data from numerous data sources. The ODS supports mission critical application systems that generate significant revenue for the company. An Oracle Active Data Guard standby database is used to protect the ODS, providing optimal data protection and availability.
- Oracle GoldenGate bidirectional replication is utilized to synchronize two databases separated by thousands of miles. User workload is distributed across each database based upon geography, workload, and service level using Oracle 12*c* Global Data Services (GDS). Each Oracle GoldenGate copy has its own local synchronous Data Guard standby database that enables zero data loss failover if an outage occurs. Oracle GoldenGate capture and apply processes are easily restarted on the new primary database following a failover because the primary and standby are an exact, up-to-date replica of each other.
- An Oracle Active Data Guard standby database used for disaster protection is temporarily converted into an Oracle GoldenGate target for the purpose of performing planned maintenance not supported by Data Guard. For example, a Siebel application upgrade requiring modification of back-end database objects which require comprehensive testing before switching users over to the new system.
- Oracle Active Data Guard is used to protect a production environment when a major database version upgrade is required offering zero or near-zero downtime (for example, Oracle 11.2.0.3 to 12*c*.) A second primary/standby environment is created using the new database version, and Oracle GoldenGate is used to replicate

data from the production environment to the copy with one-way or bidirectional replication. When Oracle GoldenGate has completed synchronizing the old and new environments, production is switched to the new environment and the old environment is decommissioned. This provides zero or minimal downtime depending upon configuration, eliminates risk by providing complete isolation between the old and new environment, and avoids any impact to data protection and availability SLAs if problems are encountered during the upgrade process.

See Also:

"Transparent Role Transitions With Oracle Data Guard and Oracle GoldenGate" MAA Best Practices white paper

3.4 Recovery Manager

Recovery Manager (RMAN) provides a comprehensive foundation for efficiently backing up and recovering the database. RMAN eliminates operational complexity while providing superior performance and availability of the database.

RMAN determines the most efficient method of executing the requested backup, restoration, or recovery operation and then submits these operations to the Oracle Database server for processing. RMAN and the server automatically identify modifications to the structure of the database and dynamically adjust the required operation to adapt to the changes.

RMAN is the standard interface to backup and restore from Recovery Appliance, local disk (ZFS storage), tape, and cloud object store.

RMAN provides the following benefits:

- Support for Oracle Sharding - RMAN support for every independent database (shard) (new in Oracle Database 12*c* Release 2)
- Enhancement for Sparse Databases - allows backup and restore to operate on `SPARSE` backup sets and or image copies (new in Oracle Database 12*c* Release 2)
- Over the Network Standby Database repair of `NONLOGGED` operation - new syntax for validation and repair on Standby - `VALIDATE/RECOVER .. NONLOGGED BLOCK;` (new in Oracle Database 12*c* Release 2)
- `RMAN DUPLICATE` feature enhanced to support creation of Far Sync from Primary and backup (new in Oracle Database 12*c* Release 2)
- `RMAN DUPLICATE` Using Encrypted Backups - RMAN enhanced support non Auto-login wallet based encrypted backups with a new `SET` command - enables interrupt-free cloning (new in Oracle Database 12*c* Release 2)
- Support for cross-platform backup and restore over the network (new in Oracle Database 12*c* Release 2)
- Network-enabled restoration allows the `RESTORE` operations to copy data files directly from one database to another over the network
- Simplified table restoration with the `RECOVER TABLE` command
- Support for Oracle Multitenant, including backup and recovery of individual pluggable databases

- Support for cross-platform Oracle Multitenant, including backup and recovery of individual PDBs (new in Oracle Database 12*c* Release 2)
- Automatic channel failover on backup and restore operations
- Automatic failover to a previous backup when the restore operation discovers a missing or corrupt backup
- Automatic creation of new database files and temporary files during recovery
- Automatic recovery through a previous point-in-time recovery—recovery through reset logs
- Block media recovery, which enables the data file to remain online while fixing the block corruption
- Fast incremental backups using block change tracking
- Fast backup and restore operations with intrafile and interfile parallelism
- Enhanced security with a virtual private recovery catalog
- Merger of incremental backups into image copies, providing up-to-date recoverability
- Optimized backup and restoration of required files only
- Retention policy to ensure that relevant backups are retained
- Ability to resume backup and restore operations in case of failure
- Automatic backup of the control file and the server parameter file, ensuring that backup metadata is available in times of database structural changes and media failure and disasters
- Easily reinstantiate a new database from an existing backup or directly from the production database (thus eliminating staging areas) using the `DUPLICATE` command.

See Also:

Oracle Database Backup and Recovery User's Guide

3.5 Oracle Secure Backup

Oracle Secure Backup is a centralized backup management solution supporting disk and tape targets, providing heterogeneous data protection in distributed UNIX, Linux, Windows, and Network Attached Storage (NAS) environments.

By protecting file system and Oracle Database data, Oracle Secure Backup provides a complete tape backup solution for your IT environment.

Oracle Secure Backup is tightly integrated with RMAN to provide the media management layer for RMAN. With optimized integration points, Oracle Secure Backup and RMAN provide the fastest and most efficient tape backup capability for Oracle Database.

You can back up distributed servers to local and remote tape or disk devices from a central Oracle Secure Backup administrative server using backup policies, calendar-

based scheduling for *lights out* operations, or on-demand backup for immediate requirements. With its highly scalable client/server architecture, Oracle Secure Backup provides local and remote data protection, using Secure Sockets layer (SSL) for secure intradomain communication and two-way server authentication.

Oracle Secure Backup provides the following benefits:

- Optimized performance achieving 25-40% faster Oracle Database backups than comparable media management products with up to 10% less CPU utilization
  - Unused block and undo block compression
  - Shared tape buffers with RMAN
- Policy-based management that allows backup administrators to exercise precise control over the backup domain
- Dynamic drive sharing for increased tape resource use
- Heterogeneous Storage Area Network (SAN) support, enabling NAS, UNIX, Windows, and Linux to share tape drives and media
- File system backup at the file, directory, file system, or raw partition level with full, incremental, and offsite backup scheduling
- Integration with Oracle Enterprise Manager, providing an intuitive, familiar interface
- Backup encryption to tape with policy-based encryption key management leveraging either Oracle Secure Backup host-based encryption or hardware encryption (tape drive)
- Broad tape-device support for new and legacy tape devices in SAN and SCSI environments
- Disk-pool devices to use disk volumes as a backup target
- Network Data Management Protocol (NDMP) support for highly efficient backup of NAS files
- Scalable, low-cost licensing model that reduces IT costs and operational considerations
- Enhanced data throughput Reliable Datagram Socket over Remote Direct Memory Access (RDS/RDMA) over InfiniBand networks for maximum backup and restore performance in Exadata Database Machine environments
- Oracle-aware backup and restoration on Non-Uniform Memory Access (NUMA) machines, ensuring OSB and Oracle Database background processes communicate in the same NUMA region for optimal performance

See Also:

Oracle Secure Backup Administrator's Guide

3.6 Oracle Real Application Clusters and Oracle Clusterware

Oracle RAC and Oracle Clusterware enable Oracle Database to run any packaged or custom application across a set of clustered servers. This capability provides the highest levels of availability and the most flexible scalability. If a clustered server fails, then Oracle Database continues running on the surviving servers. When more processing power is needed, you can add another server without interrupting access to data.

Oracle RAC enables multiple instances that are linked by an interconnect to share access to an Oracle database. In an Oracle RAC environment, Oracle Database runs on two or more systems in a cluster while concurrently accessing a single shared database. The result is a single database system that spans multiple hardware systems, enabling Oracle RAC to provide high availability and redundancy during failures in the cluster. Oracle RAC accommodates all system types, from read-only data warehouse systems to update-intensive online transaction processing (OLTP) systems.

Oracle Clusterware is software that, when installed on servers running the same operating system, enables the servers to be bound together to operate as if they are one server, and manages the availability of user applications and Oracle databases. Oracle Clusterware also provides all of the features required for cluster management, including node membership, group services, global resource management, and high availability functions:

- For high availability, you can place Oracle databases (single-instance or Oracle RAC databases), and user applications (Oracle and non-Oracle) under the management and protection of Oracle Clusterware so that the databases and applications restart when a process fails or so that a failover to another node occurs after a node failure.
- For cluster management, Oracle Clusterware presents multiple independent servers as if they are a single-system image or one virtual server. This single virtual server is preserved across the cluster for all management operations, enabling administrators to perform installations, configurations, backups, upgrades, and monitoring functions. Then, Oracle Clusterware automatically distributes the execution of these management functions to the appropriate nodes in the cluster.

Oracle Clusterware is a requirement for using Oracle RAC. Oracle Clusterware is the only clusterware that you need for most platforms on which Oracle RAC operates. Although Oracle Database continues to support third-party clusterware products on specified platforms, using Oracle Clusterware provides these main benefits:

- Dispenses with proprietary vendor clusterware
- Uses an integrated software stack from Oracle that provides disk management with local or remote Oracle Automatic Storage Management (Oracle Flex ASM) to data management with Oracle Database and Oracle RAC
- Can be configured in large clusters, called an Oracle Flex Cluster.

In addition, Oracle Database features, such as Oracle services, use the underlying Oracle Clusterware mechanisms to provide their capabilities.

Oracle Clusterware requires two clusterware components: a voting disk to record node membership information and the Oracle Cluster Registry (OCR) to record cluster configuration information. The voting disk and the OCR must reside on shared

storage. Oracle Clusterware requires that each node be connected to a private network over a private interconnect.

See Also:

Oracle Real Application Clusters Administration and Deployment Guide

3.6.1 Benefits of Using Oracle Clusterware

Oracle Clusterware provides the following benefits:

- Tolerates and quickly recovers from computer and instance failures.
- Simplifies management and support by means of using Oracle Clusterware together with Oracle Database. By using fewer vendors and an all Oracle stack you gain better integration compared to using third-party clusterware.
- Performs rolling upgrades for system and hardware changes. For example, you can apply Oracle Clusterware upgrades, patch sets, and interim patches in a rolling fashion.

  When you upgrade to Oracle Database 12c, Oracle Clusterware and Oracle ASM binaries are installed as a single binary called the Oracle Grid Infrastructure. You can upgrade Oracle Clusterware in a rolling manner from Oracle Clusterware 10*g* and Oracle Clusterware 11*g*; however, you can only upgrade Oracle ASM in a rolling manner from Oracle Database 11*g* release 1 (11.1).
- Automatically restarts failed Oracle processes.
- Automatically manages the virtual IP (VIP) address. When a node fails, the node's VIP address fails over to another node on which the VIP address can accept connections.
- Automatically restarts resources from failed nodes on surviving nodes.
- Controls Oracle processes as follows:
  - For Oracle RAC databases, Oracle Clusterware controls all Oracle processes by default.
  - For Oracle single-instance databases, Oracle Clusterware enables you to configure the Oracle processes into a resource group that is under the control of Oracle Clusterware.
- Provides an application programming interface (API) for Oracle and non-Oracle applications that enables you to control other Oracle processes with Oracle Clusterware, such as restart or react to failures and certain rules.
- Manages node membership and prevents split-brain syndrome in which two or more instances attempt to control the database.
- Using server weight-based node eviction allows for aligning the choice of which node gets evicted in case of certain failures in the cluster with business requirements, ensuring that the most important workload is kept alive for as long as possible, assuming an equal choice between servers.

- Provides the ability to perform rolling release upgrades of Oracle Clusterware, with no downtime for applications.

See Also:

Oracle Clusterware Administration and Deployment Guide

3.6.2 Benefits of Using Oracle Real Application Clusters and Oracle Clusterware

Together, Oracle RAC and Oracle Clusterware provide all of the Oracle Clusterware benefits listed in Benefits of Using Oracle Clusterware (page 3-21) plus the following benefits:

- Provides better integration and support of Oracle Database by using an all Oracle software stack compared to using third-party clusterware.
- Relocate Oracle Service automatically. Plus, when you perform additional fast application notification (FAN) and client configuration, distribute FAN events so that applications can react immediately to achieve fast, automatic, and intelligent connection and failover.
- Detect connection failures fast and automatically, and remove terminated connections for any Java application using Oracle Universal Connection Pool (Oracle UCP) Fast Connection Failover and FAN events.
- Balance work requests using Oracle UCP runtime connection load balancing.
- Use runtime connection load balancing with Oracle UCP, Oracle Call Interface (OCI), and Oracle Data Provider for .NET (ODP.NET).
- Distribute work across all available instances using load balancing advisory.
- You can configure a database so that Oracle Clusterware is aware of the CPU requirements and limits for the given database. Oracle Clusterware uses this information to place the database resource only on servers that have a sufficient number of CPUs, amount of memory, or both.
- Allow the flexibility to increase processing capacity using commodity hardware without downtime or changes to the application.
- Provide comprehensive manageability integrating database and cluster features.
- Provide scalability across database instances.
- Implement Fast Connection Failover for nonpooled connections.

3.6.3 Oracle RAC Advantages Over Traditional Cold Cluster Solutions

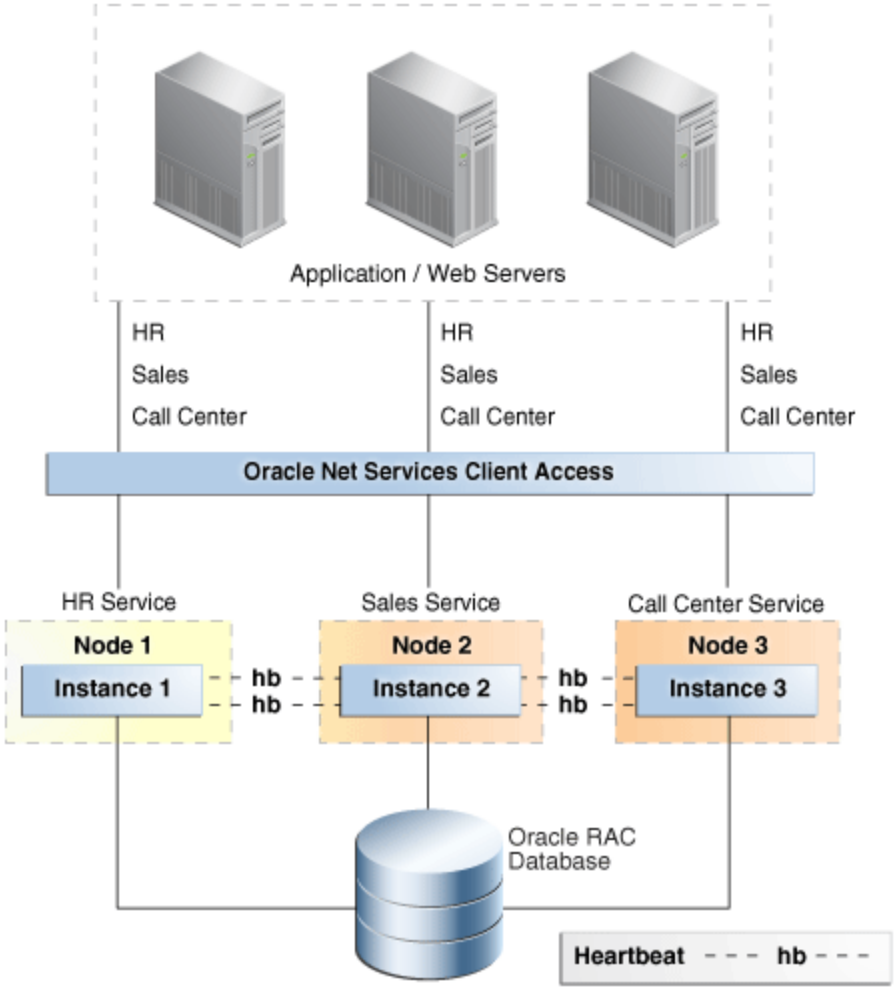
- Scalability across database instances
- Flexibility to increase processing capacity using commodity hardware without downtime or changes to the application
- Ability to tolerate and quickly recover from computer and instance failures (measured in seconds)

- Application brownout can be zero or seconds compared to minutes and hours with cold cluster solutions
- Optimized communication in the cluster over redundant network interfaces, without using bonding or other technologies

  Oracle Grid Infrastructure and Oracle RAC make use of Redundant Interconnect Usage that distributes network traffic and ensures optimal communication in the cluster. This functionality is available starting with Oracle Database 11*g* Release 2 (11.2.0.2). In previous releases, technologies like bonding or trunking were used to make use of redundant networks for the interconnect.
- Rolling upgrades for system and hardware changes
- Rolling patch upgrades for some interim patches, security patches, CPUs, and cluster software
- Fast, automatic, and intelligent connection and service relocation and failover
- Comprehensive manageability integrating database and cluster features with Grid Plug and Play and policy-based cluster and capacity management
- Load balancing advisory and run-time connection load balancing help redirect and balance work across the appropriate resources
- Oracle Quality of Service (QoS) Management for policy-based run-time management of resource allocation to database workloads to ensure service levels are met in order of business need under dynamic conditions. This is accomplished by assigning a service to a server pool where the database is running. Resources from the pool are used to make sure the required capacity is available.
- Oracle Enterprise Management support for Oracle ASM and Oracle ACFS, Grid Plug and Play, Cluster Resource Management, Oracle Clusterware and Oracle RAC Provisioning and patching.
- SCAN (Single Client Access Name) support as a single name to the clients connecting to Oracle RAC that does not change throughout the life of the cluster, even if you add or remove nodes from the cluster.

Figure 3-2 (page 3-24) shows Oracle Database with Oracle RAC architecture. This figure shows Oracle Database with Oracle RAC architecture for a partitioned three-node database. An Oracle RAC database is connected to three instances on different nodes. Each instance is associated with a service: HR, Sales, and Call Center. The instances monitor each other by checking "heartbeats." Oracle Net Services provide client access to the Application/web server tier at the top of the figure.

Figure 3-2 Oracle Database with Oracle RAC Architecture



Note:

Since Oracle version 11.2 Oracle RAC One Node or Oracle RAC is the preferred solution over Oracle Clusterware (Cold Cluster Failover) because it is a more complete and feature-rich solution.

3.7 Oracle RAC One Node

Oracle Real Application Clusters One Node (Oracle RAC One Node) is a single instance of an Oracle RAC database that runs on one node in a cluster. This feature enables you to consolidate many databases into one cluster with minimal overhead, protecting them from both planned and unplanned downtime. The consolidated databases reap the high availability benefits of failover protection, online rolling patch application, and rolling upgrades for the operating system and Oracle Clusterware.

Oracle RAC One Node enables better availability than cold failover for single-instance databases because of the Oracle technology called *online database relocation*, which intelligently migrates database instances and connections to other cluster nodes for high availability and load balancing. Online database relocation is performed using the Server Control Utility (SRVCTL).

Oracle RAC One Node provides the following:

- Always available single-instance database services
- Built-in cluster failover for high availability
- Live migration of instances across servers
- Online rolling patches and rolling upgrades for single-instance databases
- Online upgrade from single-instance to multiple-instance Oracle RAC
- Better consolidation for database servers
- Enhanced server virtualization
- Lower cost development and test platform for full Oracle RAC
- Relocation of Oracle RAC primary and standby databases configured with Data Guard. This functionality is available starting with Oracle Database 11*g* Release 2 (11.2.0.2).

Oracle RAC One Node also facilitates the consolidation of database storage, standardizes your database environment, and, when necessary, enables you to transition to a full, multiple-instance Oracle RAC database without downtime or disruption.

3.8 Oracle Automatic Storage Management

Oracle ASM provides a vertically integrated file system and volume manager directly in the Oracle Database kernel, resulting in:

- Significantly less work to provision database storage
- Higher level of availability
- Elimination of the expense, installation, and maintenance of specialized storage products
- Unique capabilities for database applications

For optimal performance, Oracle ASM spreads files across all available storage. To protect against data loss, Oracle ASM extends the concept of SAME (stripe and mirror everything) and adds more flexibility because it can mirror at the database file level rather than at the entire disk level.

More important, Oracle ASM simplifies the processes of setting up mirroring, adding disks, and removing disks. Instead of managing hundreds or possibly thousands of files (as in a large data warehouse), database administrators using Oracle ASM create and administer a larger-grained object called a **disk group**. The disk group identifies the set of disks that are managed as a logical unit. Automation of file naming and placement of the underlying database files save administrators time and ensure adherence to standard best practices.

The Oracle ASM native mirroring mechanism (two-way or three-way) protects against storage failures. With Oracle ASM mirroring, you can provide an additional level of data protection 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. After it is defined, an Oracle ASM failure group intelligently places redundant copies of the data in separate failure groups. This ensures that the data is

available and transparently protected against the failure of any component in the storage subsystem.

By using Oracle ASM, you can:

- Mirror and stripe across drives and storage arrays.
- Automatically remirror from a failed drive to remaining drives.
- Automatically rebalance stored data when disks are added or removed while the database remains online.
- Support Oracle database files and non-database files using Oracle Automatic Storage Management Cluster File System (Oracle ACFS).
- Allow for operational simplicity in managing database storage.
- Manage the Oracle Cluster Registry (OCR) and voting disks.
- Provide preferred read capability on disks that are local to the instance, which gives better performance for an extended cluster.
- Support very large databases.
- Support Oracle ASM rolling upgrades.
- Improve availability and reliability using the Oracle ASM disk scrubbing process to find and repair logical data corruptions using mirror disks.
- Support finer granularity in tuning and security.
- Provide fast repair after a temporary disk failure through Oracle ASM Fast Mirror Resync and automatic repair of block corruptions if a good copy exists in one of the mirrors.
- Provide disaster recovery capability for the file system by enabling replication of Oracle ACFS across the network to a remote site.
- Patch the Oracle ASM instance without impacting the clients that are being serviced using Oracle Flex ASM. A database instance can be directed to access Oracle ASM metadata from another location while the current Oracle ASM instance it is connected to is taken offline for planned maintenance.
- Monitor and manage the speed and status of Oracle ASM Disk Resync and Rebalance operations.
- Bring online multiple disks simultaneously and manage performance better by controlling resync parallelism using the Oracle ASM Resync Power Limit. Recover faster after a cell or disk failure, and the instance doing the resync is failing; this is made possible by using a Disk Resync Checkpoint which enables a resync to resume from where it was interrupted or stopped instead of starting from the beginning.
- Automatically connect database instances to another Oracle ASM instance using Oracle Flex ASM. The local database instance can still access the required metadata and data if an Oracle ASM instance fails due to an unplanned outage.
- Use flex diskgroups to prioritize high availability benefits across multiple databases all using the same diskgroup. Some of the key HA benefits are file extent redundancy, rebalance power limit, and rebalance priority. With flex diskgroups,

you can set different values for the above features for different databases, resulting in prioritization across multiple databases within one diskgroup.

- Use flex diskgroups to implement quoto_groups across multiple databases sharing one diskgroup which helps in space management and protection.
- Use flex diskgroups to create point-in-time database clones using the ASM split mirror feature.
- Use preferred reads with stretch clusters to improve performance by affinitizing reads to a site.

See Also:

Oracle Automatic Storage Management Administrator's Guide for more information about ACFS

3.9 Fast Recovery Area

The **fast recovery area** is a unified storage location for all recovery-related files and activities in Oracle Database. After this feature is enabled, all RMAN backups, archived redo log files, control file autobackups, flashback logs, and data file copies are automatically written to a specified file system or Oracle ASM disk group, and the management of this disk space is handled by RMAN and the database server.

Performing a backup to disk is faster because using the fast recovery area eliminates the bottleneck of writing to tape. More important, if database media recovery is required, then data file backups are readily available. Restoration and recovery time is reduced because you do not need to find a tape and a free tape device to restore the needed data files and archived redo log files.

The fast recovery area provides the following benefits:

- Unified storage location of related recovery files
- Management of the disk space allocated for recovery files, which simplifies database administration tasks
- Fast, reliable, disk-based backup and restoration

See Also:

Oracle Database Backup and Recovery User's Guide

3.10 Corruption Prevention, Detection, and Repair

Data block corruptions can be very disruptive and challenging to repair. Corruptions can cause serious application and database downtime and data loss when encountered and worse yet it can go undetected for hours, days and even weeks leading to even longer application downtime once detected. Unfortunately, there is not one way to comprehensively prevent, detect, and repair data corruptions within the database because the source and cause of corruptions can be anywhere in memory, hardware, firmware, storage, operating system, software, or user error. Worse yet, third-party solutions that do not understand Oracle data block semantics and how Oracle changes data blocks do not prevent and detect data block corruptions well. Third party remote

mirroring technologies can propagate data corruptions to the database replica (standby) leading to a double failure, data loss, and much longer downtime. Third party backup and restore solutions cannot detect corrupted backups or bad sectors until a restore or validate operation is issued, resulting in longer restore times and once again potential data loss.

Oracle MAA has a comprehensive plan to prevent, detect, and repair all forms of data block corruptions including physical block corruptions, logical block corruptions, stray writes, and lost writes. These additional safeguards provide the most comprehensive Oracle data block corruption prevention, detection, and repair solution. Details of this plan are described in the My Oracle Support note "Best Practices for Corruption Detection, Prevention, and Automatic Repair - in a Data Guard Configuration."

Table 3-1 (page 3-28) outlines block corruption checks for various manual operational checks and runtime and background corruption checks. Database administrators and the operations team can incorporate manual checks such as running RMAN backups, RMAN "check logical" validations or running the `ANALYZE VALIDATE STRUCTURE` command on important objects. Manual checks are especially important to validate data that are rarely updated or queried.

Runtime checks are far superior in that they catch corruptions almost immediately or during runtime for actively queried and updated data. Runtime checks can prevent corruptions or automatically fix corruptions resulting in better data protection and higher application availability. A new background check has been introduced in Exadata to automatically scan and scrub disks intelligently with no application overhead and to automatically fix physically corrupted blocks.

Table 3-1 Summary of Block Corruption Checks

Checks	Capabilities	Physical Block Corruption	Logical Block Corruption
Manual checks	Dbverify, Analyze	Physical block checks	Logical intra-block and inter-object consistency checks
Manual checks	RMAN	Physical block checks during backup and restore operations	Intra-block logical checks
Manual checks	RMAN and Recovery Appliance	Physical block checks during backup and restore operations Ongoing implicit backup validation by the Recovery Appliance	Intra-block logical checks
Manual checks	ASM Scrub	Physical block checks	Some logical intra-block checks

Table 3-1 (Cont.) Summary of Block Corruption Checks

Checks	Capabilities	Physical Block Corruption	Logical Block Corruption
Runtime checks	Oracle Active Data Guard	1. Continuous physical block checking at standby during transport and apply 2. Strong database isolation eliminates single point database failure 3. Automatic repair of block corruptions, including file block headers in Oracle Database 12*c* Release 2 4. Automatic database failover	1. With DB_LOST_WRITE_PROTECT enabled, detection of lost writes (11.2 and higher). With 11.2.0.4 and Data Guard broker, ability to shutdown the primary when lost writes are detected on the primary database. 2. With DB_BLOCK_CHECKING enabled on the standby, additional intra-block logical checks
Runtime checks	Database	With DB_BLOCK_CHECKSUM, in-memory data block and redo checksum validation	With DB_BLOCK_CHECKING, in-memory intra-block check validation
Runtime checks	ASM	Implicit data corruption detection for reads and writes and automatic repair if good ASM extent block pair is available during writes	
Runtime checks	DIX + T10 DIF	Checksum validation from operating system to HBA controller to disk (firmware). Validation for reads and writes for certified Linux, HBA and disks.	
Runtime checks	Hardware and Storage	Limited checks due to lack of Oracle integration. Checksum is most common.	Limited checks due to lack of Oracle integration. Checksum is most common
Runtime checks	Exadata	Comprehensive HARD checks on writes	HARD checks on writes

Table 3-1 (Cont.) Summary of Block Corruption Checks

Checks	Capabilities	Physical Block Corruption	Logical Block Corruption
Background checks	Exadata	Automatic HARD disk scrub and repair. Detects and fixes bad sectors.	

See Also:

Oracle Database Reference for more information about the views and initialization parameters

Oracle Database High Availability Best Practices for more information about preventing, detecting, and repairing data corruption

My Oracle Support Note 1302539.1

Causes of Downtime (page 1-3)

3.11 Data Recovery Advisor

Data Recovery Advisor automatically diagnoses persistent (on-disk) data failures, presents appropriate repair options, and runs repair operations at your request.

You can use Data Recovery Advisor to troubleshoot primary databases, logical standby databases, physical standby databases, and snapshot standby databases.

Data Recovery Advisor includes the following functionality:

- Failure diagnosis

  The first symptoms of database failure are usually error messages, alarms, trace files and dumps, and failed health checks. Assessing these symptoms can be complicated, error-prone, and time-consuming. Data Recovery Advisor automatically diagnoses data failures and informs you about them.

- Failure impact assessment

  After a failure is diagnosed, you must understand its extent and assess its impact on applications before devising a repair strategy. Data Recovery Advisor automatically assesses the impact of a failure and displays it in an easily understood format.

- Repair generation

  Even if a failure was diagnosed correctly, selecting the correct repair strategy can be error-prone and stressful. Moreover, there is often a high penalty for making poor decisions in terms of increased downtime and loss of data. Data Recovery Advisor automatically determines the best repair for a set of failures and presents it to you.

- Repair feasibility checks

  Before presenting repair options, Data Recovery Advisor validates them with respect to the specific environment and availability of media components required

to complete the proposed repair, including restoring files directly from the primary or standby database to complete the proposed repair.

- Repair automation

  If you accept the suggested repair option, Data Recovery Advisor automatically performs the repair, verifies that the repair was successful, and closes the appropriate failures.

- Validation of data consistency and database recoverability

  Data Recovery Advisor can validate the consistency of your data, and backups and redo stream, whenever you choose.

- Early detection of corruption

  Through Health Monitor, you can schedule periodic runs of Data Recovery Advisor diagnostic checks to detect data failures before a database process executing a transaction discovers the corruption and signals an error. Early warnings can limit the damage caused by corruption.

- Integration of data validation and repair

  Data Recovery Advisor is a single tool for data validation and repair.

Note:

Data Recovery Advisor only supports single-instance databases. Oracle RAC databases are not supported. See *Oracle Database Backup and Recovery User's Guide* for more information about Data Recovery Advisor supported database configurations.

See Also:

Oracle Database Backup and Recovery User's Guide

3.12 State Object Quarantine

State object quarantine enables the database instance to continue operating even if there are bad objects.

A state object is a session-level structure that contains metadata about the status of database resources such as processes, sessions, and transactions in the SGA. If a process or session terminates, then PMON uses the state object to release the held resources to the operating system.

In some cases, PMON can quarantine corrupted, unrecoverable state objects so that the database instance is not immediately forced to terminate. PMON continues to perform as much cleanup as possible on the quarantined object. The `V$QUARANTINE` view contains metadata such as the type of object, amount of memory consumed, Oracle error causing the quarantine, and so on.

This helps overall CDB or non-CDB database and application availability by isolating, or putting into quarantine, resources in memory so that it does not affect the rest of the database, preventing an instance or database from aborting. The impact may be limited to one session instead of the entire database and application. Accessing a corrupted memory structure in the SGA such as library cache or row cache memory objects can result in ORA-600 errors or ORA-7445 which can result in a database or

instance crash. With State Object Quarantine, the session gets an error and the corrupted resource is quarantined to ensure other processes, including critical background processes, are not impacted.

Related Topics:

Oracle Database Reference

3.13 Oracle Security Features

The best protection against human errors is to prevent their occurrence. The best way to prevent human errors is to restrict user access to only those data and services required to perform business functions. Oracle Database provides a wide range of security tools to control access to application data by authenticating database users and then enabling administrators to grant them only those privileges required to perform their duties.

In addition, the Oracle Database security model provides the ability to restrict data access at a row level using Oracle Virtual Private Database, thereby further isolating database users from data that they do not need to access.

Oracle Database provides the following security benefits:

- Authentication control to validate the identities of entities using networks, databases, and applications. Network sessions between databases, such as redo transport sessions, are also authenticated.
- Authorization control to provide limits to access and actions linked by database user identities and roles.
- Access control to objects, providing protection regardless of the entity seeking to access or alter them.
- Auditing control to monitor and gather data about specific database activities, investigate suspicious activity, deter users (or others) from inappropriate activities, and detect problems with authorization or access control implementation.
- Security policy management using profiles.
- Encryption of data residing in the database and backups, or transferred to and from databases.
- Administration of Data Guard configurations can be delegated to a class of users who would not be granted SYSDBA privileges.
- Protecting your data at rest with Transparent Data Encryption can be achieved easily with online conversion to TDE.

See Also:

- *Oracle Database Security Guide*
- *Oracle Data Guard Concepts and Administration*

3.14 Oracle Flashback Technology

Oracle Flashback technology is a group of Oracle Database features that let you view past states of database, database objects, transactions or rows or to rewind the database, database objects, transactions or rows to a previous state without using point-in-time media recovery.

With flashback features, you can:

- Perform queries to show data as it looked at a previous point in time
- Perform queries that return metadata that shows a detailed history of changes to the database
- Recover tables or rows to a previous point in time
- Automatically track and archive transactional data changes
- Roll back a transaction and its dependent transactions while the database remains online
- Undrop a table
- Recover a database to a point-in-time without a restore operation

Other than the flashback database feature, most Oracle Flashback features use the Automatic Undo Management (AUM) system to obtain metadata and historical data for transactions. They rely on undo data, which are records of the effects of individual transactions. For example, if a user runs an UPDATE statement to change a salary from 1000 to 1100, then Oracle Database stores the value 1000 in the undo data.

Undo data is persistent and survives a database shutdown. By using flashback features, you can use undo data to query past data or recover from logical damage. Besides using it in flashback features, Oracle Database uses undo data to perform these actions:

- Roll back active transactions
- Recover terminated transactions by using database or process recovery
- Provide read consistency for SQL queries

Oracle Flashback can address and rewind data that is compromised due to various human or operator errors that inadvertently or maliciously change data, cause bad installations and upgrades, and result in logical errors in applications. These problems are addressed in the following phases, and use features such as flashback transaction, flashback drop, flashback table, and flashback database.

Phase 1: Detection of logical failure, which is usually done by the application.

Phase 2: Error investigation using features such as flashback query, flashback version query, and flashback transaction query and the `DBMS_FLASHBACK` package.

Phase 3: Error recovery.

3.14.1 Oracle Flashback Query

Oracle Flashback Query (Flashback Query) provides the ability to view data as it existed in the past by using the Automatic Undo Management system to obtain metadata and historical data for transactions. Undo data is persistent and survives a

database malfunction or shutdown. The unique features of Flashback Query not only provide the ability to query previous versions of tables, they also provide a powerful mechanism to recover from erroneous operations.

Uses of Flashback Query include:

- Recovering lost data or undoing incorrect, committed changes. For example, rows that were deleted or updated can be immediately repaired even after they were committed.
- Comparing current data with the corresponding data at some time in the past. For example, by using a daily report that shows the changes in data from yesterday, it is possible to compare individual rows of table data, or find intersections or unions of sets of rows.
- Checking the state of transactional data at a particular time, such as verifying the account balance on a certain day.
- Simplifying application design by removing the need to store certain types of temporal data. By using Flashback Query, it is possible to retrieve past data directly from the database.
- Applying packaged applications, such as report generation tools, to past data.
- Providing self-service error correction for an application, enabling users to undo and correct their errors.

See Also:

Oracle Database Development Guide

3.14.2 Oracle Flashback Version Query

Oracle Flashback Version Query is an extension to SQL that you can use to retrieve the versions of rows in a given table that existed at a specific time interval. Oracle Flashback Version Query returns a row for each version of the row that existed in the specified time interval. For any given table, a new row version is created each time the `COMMIT` statement is executed.

Oracle Flashback Version Query is a powerful tool that database administrators (database administrators) can use to run analysis to determine the source of problems. Additionally, application developers can use Oracle Flashback Version Query to build customized applications for auditing purposes.

See Also:

Oracle Database Development Guide

3.14.3 Oracle Flashback Transaction

Oracle Flashback Transaction backs out a transaction and its dependent transactions. The `DBMS_FLASHBACK.TRANSACTION_BACKOUT()` procedure rolls back a transaction and its dependent transactions while the database remains online. This recovery operation uses undo data to create and execute the compensating transactions that return the affected data to its original state. You can query the

DBA_FLASHBACK_TRANSACTION_STATE view to see whether the transaction was backed out using dependency rules or forced out by either:

- Backing out nonconflicting rows
- Applying undo SQL

Oracle Flashback Transaction increases availability during logical recovery by quickly backing out a specific transaction or set of transactions and their dependent transactions. You use one command to back out transactions while the database remains online.

See Also:

- *Oracle Database Development Guide*
- *Oracle Database PL/SQL Packages and Types Reference*

3.14.4 Oracle Flashback Transaction Query

Oracle Flashback Transaction Query provides a mechanism to view all of the changes made to the database at the transaction level. When used in conjunction with Oracle Flashback Version Query, it offers a fast and efficient means to recover from a human or application error. Oracle Flashback Transaction Query increases the ability to perform online diagnosis of problems in the database by returning the database user that changed the row, and performs analysis and audits on transactions.

See Also:

Oracle Database Development Guide

3.14.5 Oracle Flashback Table

Oracle Flashback Table recovers a table to a previous point in time. It provides a fast, online solution for recovering a table or set of tables that were changed by a human or application error. In most cases, Oracle Flashback Table alleviates the need for administrators to perform more complicated point-in-time recovery operations. The data in the original table is not lost when you use Oracle Flashback Table because you can return the table to its original state.

See Also:

Oracle Database Backup and Recovery User's Guide

3.14.6 Oracle Flashback Drop

Dropping objects by accident is a problem for database users and database administrators. Although there is no easy way to recover dropped tables, indexes, constraints, or triggers, Oracle Flashback Drop provides a safety net when you are dropping objects. When you drop a table, it is automatically placed into the Recycle Bin. The Recycle Bin is a virtual container where all dropped objects reside. You can continue to query data in a dropped table.

See Also:

Oracle Database Backup and Recovery User's Guide

3.14.7 Restore Points

When an Oracle Flashback recovery operation is performed on the database, the DBA must determine the point in time—identified by the system change number (SCN) or time stamp—to which you can later flash back the data. Oracle Flashback restore points are labels that you can define to substitute for the SCN or transaction time used in Flashback Database, Flashback Table, and Oracle Recovery Manager (RMAN) operations. Furthermore, a database can be flashed back through a previous database recovery and opened with an `OPEN RESETLOGS` command by using guaranteed restore points. Guaranteed restore points allow major database changes—such as database batch jobs, upgrades, or patches—to be quickly undone by ensuring that the undo required to rewind the database is retained.

Using the restore points feature provides the following benefits:

- The ability to quickly restore to a consistent state, to a time before a planned operation that has gone awry (for example, a failed batch job, an Oracle software upgrade, or an application upgrade)
- The ability to resynchronize a snapshot standby database with the primary database
- A quick mechanism to restore a test or cloned database to its original state

See Also:

Oracle Database Backup and Recovery User's Guide

3.14.7.1 Oracle Flashback Database

Oracle Flashback Database is the equivalent of a fast rewind button, quickly returning a database to a previous point in time without requiring a time consuming restore and roll forward using a backup and archived logs. The larger the size of the database, the greater the advantage of using Oracle Flashback Database for fast point in time recovery.

Enabling Oracle Flashback Database provides the following benefits:

- Fast point in time recovery to repair logical corruptions, such as those caused by administrative error.
- Useful for iterative testing when used with Oracle restore points. A restore point can be set, database changes implemented, and test workload run to assess impact. Oracle Flashback Database can then be used to discard the changes and return the database to the original starting point, different modifications can be made, and the same test workload run a second time to have a true basis for comparing the impact of the different configuration changes.
- Data Guard uses Oracle Flashback Database to quickly reinstantiate a failed primary database as a new standby (after a failover has occurred), without requiring the failed primary to be restored from a backup.

- Flashback database operates at the CDB level or the PDB level. See Flashback Pluggable Database (page 3-37) for more information.

See Also:

Oracle Database Backup and Recovery User's Guide

3.14.8 Flashback Pluggable Database

You can rewind a PDB to a previous SCN. The `FLASHBACK PLUGGABLE DATABASE` command, which is available through SQL or Recovery Manager, is analogous to `FLASHBACK DATABASE` in a non-CDB.

Flashback PDB protects an individual PDB against data corruption, widespread user errors, and redo corruption. The operation does not rewind data in other PDBs in the CDB.

In releases prior to Oracle Database 12*c* Release 2 (12.2), you could create a restore point—an alias for an SCN—only when connected to the root. Now you can use `CREATE RESTORE POINT ... FOR PLUGGABLE DATABASE` to create a PDB restore point, which is only usable within a specified PDB. As with CDB restore points, PDB restore points can be normal or guaranteed. A guaranteed restore point never ages out of the control file and must be explicitly dropped. If you connect to the root, and if you do not specify the `FOR PLUGGABLE DATABASE` clause, then you create a CDB restore point, which is usable by all PDBs.

A special type of PDB restore point is a clean restore point, which you can only create when a PDB is closed. For PDBs with shared undo, rewinding the PDB to a clean restore point is faster than other options because it does not require restoring backups or creating a temporary database instance.

Related Topics:

Oracle Database Backup and Recovery User's Guide

3.14.9 Block Media Recovery Using Flashback Logs or Physical Standby Database

After attempting to automatically repair corrupted blocks, block media recovery can optionally retrieve a more recent copy of a data block from the flashback logs to reduce recovery time. Automatic block repair allows corrupt blocks on the primary database to be automatically repaired as soon as they are detected, by using good blocks from a physical standby database.

Furthermore, a corrupted block encountered during instance recovery does not result in instance recovery failure. The block is automatically marked as corrupt and added to the RMAN corruption list in the `V$DATABASE_BLOCK_CORRUPTION` table. You can subsequently issue the RMAN `RECOVER BLOCK` command to fix the associated block. In addition, the RMAN `RECOVER BLOCK` command restores blocks from a physical standby database, if it is available.

See Also:

Oracle Database Backup and Recovery User's Guide for block media repair

Oracle Database Backup and Recovery Reference for the RMAN `RECOVER BLOCK` command

3.14.10 Flashback Data Archive

The Flashback Data Archive is stored in a tablespace and contains transactional changes to every record in a table for the duration of the record's lifetime. The archived data can be retained for a much longer duration than the retention period offered by an undo tablespace, and used to retrieve very old data for analysis and repair.

See Also:

Oracle Database Development Guide

3.15 Oracle Data Pump and Data Transport

Oracle Data Pump technology enables very high-speed movement of data and metadata from one database to another. Data Pump is used to perform the following planned maintenance activities:

- Database migration to a different platform
- Database migration to pluggable databases
- Database upgrade

See Oracle High Availability Solutions for System and Software Maintenance (page 5-18) for more information about using this technology for planned maintenance.

The Data Pump features that enable the planned maintenance activities listed above are the following:

- Full transportable export/import to move an entire database to a different database instance
- Transportable tablespaces to move a set of tablespaces between databases

3.16 Oracle Replication Technologies for Non-Database Files

Table 3-2 (page 3-38) describes the Oracle replication technologies for non-database files.

Table 3-2 Oracle Replication Technologies for Non-Database Files

Technology	Recommended Usage	Comments
Oracle Database File System (page 3-39)	Recommended for providing stronger synchronization between database and non-database systems.	Can be integrated with the database to maintain complete consistency between the database changes and the file system changes All data stored in the database and can be used with Oracle Active Data Guard to provide both disaster recovery and read-only access Can take advantage all of the Oracle database features

Table 3-2 (Cont.) Oracle Replication Technologies for Non-Database Files

Technology	Recommended Usage	Comments
Oracle ASM Cluster File System (page 3-40)	Recommended to provide a single-node and cluster-wide file system solution integrated with Oracle ASM, Oracle Clusterware, and Oracle Enterprise Manager technologies. Provides a loosely coupled full stack replication solution when combined with Data Guard or Oracle GoldenGate.	Oracle ACFS establishes and maintains communication with the Oracle ASM instance to participate in Oracle ASM state transitions including Oracle ASM instance and disk group status updates and disk group rebalancing. Supports many database and application files, including executables, database trace files, database alert logs, application reports, BFILEs, and configuration files. Other supported files are video, audio, text, images, engineering drawings, and other general-purpose application file data. Can provide near-time consistency between database changes and file system changes when point-in-time recovery happens Can be exported and accessed by remote clients using standard NAS File Access Protocols such as NFS and CIFS.
Oracle Solaris ZFS Storage Appliance Replication (page 3-40)	Recommended for disaster recovery protection for non-database files, and specifically for Oracle Fusion Middleware critical files stored outside of the database.	Replicates all non-database objects, including Oracle Fusion Middleware binaries configuration Can provide near time consistency between database changes and file system changes when point-in-time recovery happens

3.16.1 Oracle Database File System

Oracle Database File System (DBFS) takes advantage of the features of the database to store files, and the strengths of the database in efficiently managing relational data, to implement a standard file system interface for files stored in the database. With this interface, storing files in the database is no longer limited to programs specifically written to use BLOB and CLOB programmatic interfaces. Files in the database can now be transparently accessed using any operating system (OS) program that acts on files. For example, extract, transform, and load (ETL) tools can transparently store staging files in the database.

Oracle DBFS provides the following benefits:

- Full stack integration recovery and failover: By storing file system files in a database structure, it is possible to easily perform point-in-time recovery of both database objects and file system data.
- Disaster Recovery System Return on Investment (ROI): All changes to files contained in DBFS are also logged through the Oracle database redo log stream and thus can be passed to a Data Guard physical standby database. Using Oracle Active Data Guard technology, the DBFS file system can be mounted read-only

using the physical standby database as the source. Changes made on the primary are propagated to the standby database and are visible once applied to the standby.

- File system backups: Because DBFS is stored in the database as database objects, standard RMAN backup and recovery functionality can be applied to file system data. Any backup, restore, or recovery operation that can be performed on a database or object within a database can also be performed against the DBFS file system.

3.16.2 Oracle ASM Cluster File System

Oracle ASM Cluster File System (ACFS) is a multiplatform, scalable file system, and storage management technology that extends Oracle Automatic Storage Management (Oracle ASM) functionality to support customer files maintained outside of Oracle Database. Oracle ACFS supports many database and application files, including executables, database trace files, database alert logs, application reports, BFILEs, and configuration files. Other supported files are video, audio, text, images, engineering drawings, and other general-purpose application file data.

Oracle ACFS takes advantage of the following Oracle ASM functionality:

- Oracle ACFS dynamic file system resizing
- Maximized performance through direct access to Oracle ASM disk group storage
- Balanced distribution of Oracle ACFS across Oracle ASM disk group storage for increased I/O parallelism
- Data reliability through Oracle ASM mirroring protection mechanisms

An additional feature of Oracle ACFS is Oracle ACFS Replication which, similar to Data Guard for the database, enables replication of Oracle ACFS file systems across the network to a remote site, providing disaster recovery capability for the file system. Oracle ACFS replication captures file system changes written to disk for a primary file system and records the changes in files called replication logs. These logs are transported to the site hosting the associated standby file system where background processes read the logs and apply the changes recorded in the logs to the standby file system. After the changes recorded in a replication log are successfully applied to the standby file system, the replication log is deleted from the sites hosting the primary and standby file systems.

Oracle Data Guard and Oracle ACFS can be combined to provide a full stack high availability solution with Data Guard protecting the database with a standby database and Oracle ACFS replicating the file system changes to the standby host. For planned outages the file system and the database remain consistent to a point in time with zero data loss. See the Oracle MAA white paper Full Stack Role Transition - Oracle ACFS and Oracle Data Guard at http://www.oracle.com/technetwork/database/availability/data-guard-acfs-2931205.pdf for more information.

3.16.3 Oracle Solaris ZFS Storage Appliance Replication

The Oracle Solaris ZFS Storage Appliance series supports snapshot-based replication of projects and shares from a source appliance to any number of target appliances manually, on a schedule, or continuously for the following use cases:

- Disaster recovery: Replication can be used to mirror an appliance for disaster recovery. In the event of a disaster that impacts the service of the primary appliance (or even an entire data center), administrators activate the service at the

disaster recovery site, which takes over using the most recently replicated data. When the primary site is restored, data changed while the disaster recovery site was in service can be migrated back to the primary site, and normal service is restored. Such scenarios are fully testable before a disaster occurs.

- Data distribution: Replication can be used to distribute data (such as virtual machine images or media) to remote systems across the world in situations where clients of the target appliance would not ordinarily be able to reach the source appliance directly, or such a setup would have prohibitively high latency. One example uses this scheme for local caching to improve latency of read-only data (such as documents).
- Disk-to-disk backup: Replication can be used as a backup solution for environments in which tape backups are not feasible. Tape backup might not be feasible, for example, because the available bandwidth is insufficient or because the latency for recovery is too high.
- Data migration: Replication can be used to migrate data and configuration between Oracle Solaris ZFS Storage appliances when upgrading hardware or rebalancing storage. Shadow migration can also be used for this purpose.

The architecture of Oracle Solaris ZFS Storage Appliance also makes it an ideal platform to complement Data Guard for disaster recovery of Oracle Fusion Middleware. Oracle Fusion Middleware has a number of critical files that are stored outside of the database. These binaries, configuration data, metadata, logs and so on also require data protection to ensure availability of the Oracle Fusion Middleware. For these, the built-in replication feature of the ZFS Storage Appliance is used to move this data to a remote disaster recovery site.

Benefits of the Oracle Solaris ZFS Storage Appliance when used with Oracle Fusion Middleware include:

- Leverages remote replication for Oracle Fusion Middleware
- Provides ability to quickly create clones and snapshots of databases to increase ROI of DR sites

3.17 Client and Application Failover

A highly available architecture requires the application tier to transparently fail over to a surviving instance or database advertising the required service. This ensures that applications are generally available or minimally impacted in the event of node failure, instance failure, data corruption, or database failures. Transparent client failover enables applications to fail over to another available Oracle RAC instance or to another database (such as in the case of a Data Guard role transition or Oracle GoldenGate).

Client failover encompasses failure notification, connection cleanup, automatic retries and reconnection to a database service residing in another Oracle RAC instance or database and possibly retry the database request.

At a high level, the following components are used to provide for seamless client failover:

- **Services**

  Oracle Database provides a powerful automatic workload management facility, called services, to enable the enterprise grid vision. Services are entities that you can define in Oracle databases that enable you to group database workloads, route

work to the optimal instances that are assigned to offer the service, and achieve high availability for planned and unplanned actions.

- **High Availability Framework**

  An Oracle RAC component that enables Oracle Database to maintain components in a running state.

- **Fast Application Notification (FAN)**

  FAN is a high availability notification mechanism that Oracle RAC uses to notify other processes about configuration-level and service-level information that includes service status changes, such as UP or DOWN events. FAN also provides load advisory notifications. The Oracle client drivers and Oracle connection pools respond to FAN events and take immediate action. FAN UP and DOWN events can apply to instances, services, and nodes.

- **Transaction Guard**

  Transaction Guard is a tool that provides a protocol and an API for at-most-once execution of transactions in case of unplanned outages and duplicate submissions.

- **Application Continuity**

  Application Continuity provides a general purpose infrastructure that replays the in-flight request when a recoverable error is received, masking many system, communication, and storage outages and hardware failures. Unlike other recovery technologies, this feature attempts to recover the transactional and non-transactional session states beneath the application, so that the outage appears to the application as a delayed execution.

- **Connection Load Balancing**

  Connection Load Balancing is a feature of Oracle Net Services that balances incoming connections across all of the instances that provide the requested database service.

  With run-time connection load balancing, applications can use load balancing advisory events to provide better service to users. Oracle JDBC, Oracle Universal Connection Pool for Java, OCI session pool, ODP.NET, and Oracle WebLogic Server Active GridLink for Oracle RAC clients are automatically integrated to take advantage of load balancing advisory events. The load balancing advisory informs the client about the current service level that an instance is providing for a service

- **Fast Connection Failover**

  Fast Connection Failover is the ability of Oracle Clients to provide rapid failover of connections by subscribing to FAN events.

- **Transparent Application Failover (TAF)**

  Transparent Application Failover is a run-time failover for high availability environments that refers to the failover and re-establishment of application-to-service connections. It enables client applications to automatically reconnect to the database if the connection fails, and, optionally, resume a SELECT statement that was in progress. This reconnection happens automatically from within the Oracle Call Interface (OCI) library.

- **Single Client Access Name (SCAN)**

SCAN provides a single name to the clients connecting to Oracle RAC that does not change throughout the life of the cluster, even if you add or remove nodes from the cluster. Clients connecting with SCAN can use a simple connection string, such as a thin JDBC URL or EZConnect, and still achieve the load balancing and client connection failover.

- **Global Data Services**

  Global Data Services (GDS) is a new capability of Oracle Database that extends the concept of services to a globally replicated configuration involving a combination of single-instance, Oracle RAC, Oracle Active Data Guard, and Oracle GoldenGate. This enables services to be deployed anywhere within this globally replicated configuration, supporting load balancing, high availability, database affinity, and so on.

- **Connection Time Failover**

  Oracle Net supports connect descriptors with multiple lists of addresses, each with its own characteristics. Connection time failover allows for a new connection attempt to fail over to a different address if the connection to the first address fails.

See Also:

Oracle Database Concepts for information about how the database processes transactions

Oracle Real Application Clusters Administration and Deployment Guide for information about Dynamic Database Services

Oracle Database 2 Day + Real Application Clusters Guide for information about Dynamic Database Services

Oracle Database Global Data Services Concepts and Administration Guide

3.17.1 Client Failover Processing for Connections

At a high level, automating client failover in an **MAA environment** includes relocating database services to available resources, notifying clients that a failure has occurred, potentially breaking them out of TCP timeout, and redirecting application connections to available resources where the database service is active. See My Oracle Support note 1617163.1, "Client and Application Failover Validation Matrix," for configuration and best practices with a particular client or application.

The components described in the introduction to this chapter that are used to process the failover of application connections depend on the configuration of your MAA environment.

Table 3-3 Client Failover Processing for Connections

MAA Configuration	Service Relocation	Application Notification	Session Failover and Recovery[1]
Single Instance with Data Guard	• Start service after Data Guard failover using a trigger written on the `DB_ROLE_CHANGE` system event • Use Global Data Services	Configure your operating system for efficient TCP timeouts on the hosts that run the application layer	Configure Transparent Oracle Failover (TAF) for OCI clients. If not using TAF, you can include Transaction Guard in your application for OCI, JDBC Thin, or ODP.
Oracle RAC Database or Oracle RAC One Node	• Use services managed by Oracle Clusterware	Configure for Fast Application Notification	Configure Transparent Oracle Failover (TAF) for OCI clients. Configure Application Continuity for Thin JDBC Clients., If not using these, you may include Transaction Guard in your application for OCI, JDBC Thin, or ODP. (TAF and AC include Transaction Guard)
Oracle RAC Database with Data Guard	• Use services managed by Oracle Clusterware for service failover within each cluster • Use role based services by Oracle Clusterware for service failover between sites	Configure for Fast Application Notification	Configure Transparent Oracle Failover (TAF) for OCI clients. Configure Application Continuity for JDBC thin clients. If not using these you can include Transaction Guard in your application for OCI, JDBC Thin, or ODP. (TAF and AC include Transaction Guard)
Replicated Databases	• Use Global Data Services	Configure your operating system for efficient TCP timeouts on the hosts that run the application layer	Configure Transparent Oracle Failover (TAF) for OCI clients using BASIC only.

1 Application continuity is also an option for all of these configurations except Replicated Databases, as long as the client is JDBC, UCP, or AGL.

The following sections provide more information about service relocation and application notification.

See Also:

My Oracle Support note 1617163.1 for configuration and best practices with a particular client or application

3.17.1.1 Services

A service name is a logical representation of a service used for client connections. When a client connects to a listener, it requests a connection to a service. When a database instance starts, it registers itself with a listener as providing one or more services by name. A single service, as known by a listener, can identify one or more database instances in an Oracle RAC or Data Guard environment. A single database instance can register one or more services with a listener.

3.17.1.1.1 Service Usage in a Single-Instance Database and Data Guard Environment

The application should connect to the database using a primary specific service name, that is a user-created service that is only active on the primary database. In the event of a Data Guard failover, this service migrates to any database that currently holds the primary role. This can be accomplished in single-instance environments that do not have Oracle Clusterware installed by creating a trigger that executes based on the `ON_STARTUP` system event. This trigger should check the `DATABASE_ROLE` value of the `V$DATABASE` view, and if the value is `PRIMARY`, then start the user created service.

3.17.1.1.2 Service Usage in an Oracle RAC Database Environment

Resource profiles are automatically created when you define a service. A resource profile describes how Oracle Clusterware should manage the service and which instance the service should failover to if the preferred instance stops. Resource profiles also define service dependencies for the instance and the database. Due to these dependencies, if you stop a database, then the instances and services are automatically stopped in the correct order.

When you define a service for an administrator-managed database, you define which instances usually support that service using `SRVCTL` with the `-preferred` parameter. These are known as the preferred instances. You can also define other instances to support a service if the service's preferred instance fails using `SRVCTL` with the `-available` parameter. These are known as available instances.

When you specify preferred instances, you are specifying the number of instances on which a service usually runs. This is the maximum cardinality of the service. Oracle Clusterware attempts to ensure that the service runs on the number of instances for which you have configured the service. Afterward, due to either instance failure or planned service relocations, a service may be running on an available instance.

If an instance fails, then you cannot control to which available instance Oracle Clusterware relocates the services if there are multiple instances in the list. During a planned operation, however, you can manually direct the service to any instance in either the preferred or the available list not currently offering the service.

3.17.1.1.3 Service Usage in an Oracle RAC Database and Data Guard Environment

If you configured Data Guard in your Oracle RAC environment, then you can define a role for each service using `SRVCTL` with the `-l` parameter. When you specify a role for a service, Oracle Clusterware automatically starts the service only when the database role matches the role you specified for the service. Valid roles are PRIMARY,

PHYSICAL_STANDBY, LOGICAL_STANDBY, and SNAPSHOT_STANDBY and you can specify more than one role for a service.

If multiple databases in the cluster offer the same service name, then Oracle RAC balances connections to that service across all such databases. This is useful for standby and active Data Guard databases, but if you want client connections to a service to be directed to a particular database, then the service name must be unique within the cluster (not offered by any other database).

See Also:

Oracle Data Guard Concepts and Administration for more information about database roles

---

3.17.1.1.4 Service Usage in a Replicated Environment or Oracle Active Data Guard Environment

The Global Data Services framework is the software infrastructure for global services. This framework automates and centralizes configuration, maintenance, and monitoring of a database cloud, and enables load balancing and failover for global services. The framework manages these virtualized resources with minimal administrative overhead, enabling the cloud to handle additional client requests.

The Global Data Services framework is built around the following preexisting Oracle Database technologies:

- Oracle Active Data Guard

  Enables high-performance farms of read-only databases.

- Data Guard Broker

  Enables creation, management, and monitoring of Data Guard configurations that include a primary database and up to 30 standby databases.

- Oracle GoldenGate

  Enables replication updates among multiple databases.

3.17.1.2 Fast Application Notification

With FAN, the continuous, dynamic database services built into Oracle RAC, Data Guard, and Global Data Services are extended to applications and mid-tier servers. When the state of a database service changes (for example, up, down, or not restarting), the new status is posted to interested subscribers through FAN events. Oracle drivers and applications use these events to achieve very fast detection of failures, balancing of connection pools following failures, and balancing of connection pools again when the failed components are repaired. For example, when the service at an instance starts, the FAN event is used immediately to route work to that resource. When the service at an instance or node fails, the FAN event is used immediately to interrupt applications to recover.

To solve high availability problems with database connections, Oracle Clusterware and Data Guard Broker post a FAN event, and also executes server-side callouts, immediately when a service changes state. A FAN event payload contains the relevant information that describes the status of the service on Oracle RAC. On receipt of the FAN event, applications can terminate sessions in communication with the failed instance or node, notify sessions waiting to resume operation, and reorganize in coming work when additional resources are available. To know which sessions to

process, every session using Oracle Database has a unique connection signature. The session signatures match the FAN payload.

For planned outages, use any connection pool with FAN configured: OCI, UCP, ICC, WebLogic Server Active Grid Link, or ODP.Net. In addition to using FAN with connection pools, for the thin Java driver, beginning with Oracle Database 12*c* Release 2, FAN is automatically enabled by placing the ons.jar and simpleFAN.jar files on the `CLASSPATH`, and by using the recommended TNS format.

The FAN planned event drains the work at request boundaries. The customer's code can then retry for a new connection thereby avoiding outages with planned maintenance even without using a pool. Immediately, the FAN event is received for a planned down, the idle connections are removed from the pool for that service or instance, and the active (borrowed) connections are marked for release when they are returned to the pool. This effectively drains the work for planned outages with no interruption to the users. For the thin Java driver, connections are closed when your code checks the connection status after a FAN down event was received. See My Oracle Support Document 1593712.1 for the steps required to gracefully perform planned maintenance operations without application interruption.

FAN is also used for posting advisories for runtime connection load balancing, Web Affinity, and Data Dependent Routing.

In Oracle Database 12*c* Release 2 (12.2), Java containers, frameworks, and applications can use new APIs to subscribe to Oracle Database RAC FAN events for building high availability solutions They can receive FAN events (high availability and load balancing advisories) through the JDBC driver .without the need for a UCP

See Also:

Oracle Real Application Clusters Administration and Deployment Guide for information about Dynamic Database Services

Oracle Real Application Clusters Administration and Deployment Guide for information about automatically enabling FAN

Oracle Database JDBC Developer's Guide for JDBC subscription information

My Oracle Support Document 1593712.1 for information about planned maintenance without application interruption

3.17.1.2.1 Enabling FAN for Oracle Clients

Oracle integrated FAN with many of the common client application environments that are used to connect to Oracle RAC databases. The easiest way to use FAN is to use an integrated Oracle client. Oracle Database 12*c* Release 2 client drivers are FAN-aware, and FAN is enabled by default. This includes the JDBC Thin driver and Oracle Data Provider for Net (ODP.NET) drivers.

Due to the integration with FAN, Oracle integrated clients are more aware of the current status of an Oracle RAC cluster. This prevents client connections from waiting or trying to connect to instances or services that are no longer available. When instances start, Oracle RAC uses FAN to notify the connection pool so that the connection pool can create connections to the recently started instance and take advantage of the additional resources that this instance provides.

Oracle client drivers that are integrated with FAN can:

- Remove terminated connections immediately when a service is declared as DOWN at an instance, and immediately when nodes are declared as DOWN
- Report errors to clients immediately when Oracle Database detects the NOT RESTARTING state, instead of making the client wait while the service repeatedly attempts to restart

Oracle connection pools that are integrated with FAN can:

- Balance connections across all of the Oracle RAC instances when a service starts; this is preferable to directing the sessions that are defined for the connection pool to the first Oracle RAC instance that supports the service
- Balance work requests at run time using load balancing advisory events

---

See Also:

Oracle Real Application Clusters Administration and Deployment Guide for information about how to enable FAN for all Oracle clients

Oracle Database JDBC Developer's Guide for information about JDBC support for Oracle RAC FAN APIs

Oracle Call Interface Programmer's Guide for information about HA event notification

---

3.17.1.2.2 Considerations for Applications That Cannot Use FAN

Configure your operating system for efficient TCP timeouts on the hosts that run the application layer. The OS TCP timeouts should be set to the amount of time it takes for the database layer to failover and the database services to be started. Consult your operating system manuals for how to properly configure TCP timeout.

Configure reconnection logic within the application to respond appropriately in the event of an exception. For example, when a session from the connection pool receives an exception that results in a disconnection (such as an ORA-3113 error), the application should automatically attempt to reconnect that session. The reconnection attempts should be configured such that they will continue for the length of time that it takes to failover the database layer and bring the application services online.

3.17.2 Transaction Failover and Protection

Transaction failover and protection technologies include Transaction Guard and Application Continuity.

3.17.2.1 Transaction Guard

Transaction Guard is a generic tool for applications to provide a reliable, known outcome for transactions following planned and unplanned outages. Applications use a new concept called the logical transaction ID to determine the outcome of the last transaction open in a database session following an outage. Without using Transaction Guard, applications that attempt to retry operations following outages can cause logical corruption by committing duplicate transactions.

Failing to recognize that the last submission has committed, will commit sometime soon, or has not run to completion can lead applications that attempt to replay to cause duplicate transaction submissions and other forms of logical corruption because the software might try to re-issue already persisted changes.

Without Transaction Guard, if a transaction was started and a commit was issued, the commit message that is sent back to the client is not durable. The client is left not knowing whether the transaction committed or not. The transaction cannot be validly resubmitted if the non-transactional state is incorrect or if it is already committed. In the absence of guaranteed commit and completion information, resubmission can lead to transactions applied more than once and in the incorrect state.

Starting with Oracle Database 12*c* Release 2, Transaction Guard supports XA optimizations. Transaction Guard supports local transactions and XA transactions that use `TMONEPHASE` during the commit operation. When the application issues an XA transaction that uses `TMTWOPHASE`, the Transaction Guard disables itself for that transaction and automatically re-enables to prepare itself for the next transaction. This allows Transaction Guard to support the following XA transactions:

- Local transactions that use autocommit
- Local transactions that use an explicit commit
- XA transactions that commit with `TMONEPHASE` flag

See Also:

Oracle Database 2 Day + Real Application Clusters Guide for information about Transaction Guard with Oracle RAC and Dynamic Database Services

Oracle Database Development Guide for information about SQL processing for application developers

Oracle Database Development Guide for information about using Transaction Guard

Oracle Database PL/SQL Packages and Types Reference for information about the DBMS_APP_CONT package

3.17.2.2 Application Continuity

A highly available architecture requires the ability of the application tier to transparently fail over to a surviving instance or database advertising the required service. This ensures that applications are generally available or minimally impacted in the event of all recoverable failures including database sessions, nodes, instances, networks, data corruption, and read and write time outs. Application Continuity masks hardware, software, network, storage errors, and time outs in a high availability environment running either Oracle RAC, Oracle RAC One, or Active Data Guard for instance or site failover. Application Continuity attempts to mask recoverable outages by replaying the request at another available Oracle RAC instance or to another database (such as in the case of a Data Guard role transition).

Application Continuity encompasses:

- FAN: failure notification
- Connection cleanup
- Automatic reconnection and retries of database service residing in another Oracle RAC instance or database
- Replay of the in-flight request

Masking outages of the database session is a complex task for application development and, as a result, errors and time outs are often exposed to the users. Application Continuity attempts to mask outages from users and applications by recovering the database session following recoverable outages, unplanned and planned. Application Continuity performs this recovery beneath the application so that the outage appears to the application as a delayed execution. For the recovery to succeed, the data and messages restored to the client by Application Continuity must be the same as those that the application has seen and potentially made decisions on.

Application Continuity is started for outages that are recoverable, typically related to underlying software, foreground, hardware, communications, network, or storage layers. Application Continuity is used to improve the user experience when handling both unplanned outages and planned outages.

Application Continuity is available for applications using Java, OCI, and ODP.NET, Unmanaged Driver

- Oracle WebLogic Server for non-XA and XA data sources
- Oracle Universal Connection Pool, used standalone or as a data source for third party Application Servers including IBM WebSphere and Apache Tomcat
- JDBC applications using the JDBC PooledConnection interface
- Oracle JDBC-Thin Replay Driver
- Oracle Tuxedo for non-XA data sources
- Oracle OCI Session Pool
- ODP.NET, Unmanaged Driver
- Oracle SQL*Plus

Beginning in Oracle Database 12*c* Release 2 (12.2), Application Continuity supports Oracle XA data sources and `FAILOVER_RESTORE` to automatically restore initial states before failover starts, for those applications that pre-color connections.

Application Continuity provides incomplete request recovery of in-flight work, masking system failures, communication failures, hardware failures, and storage outages from the user, providing an improved user experience, higher application availability, and improved developer productivity.

See Also:

Oracle Database Concepts for information about transactions

Oracle Database Development Guide for information about transactions

Oracle Real Application Clusters Administration and Deployment Guide for information about Dynamic Database Services

Oracle Database 2 Day + Real Application Clusters Guide for information about Dynamic Database Services

Oracle Database JDBC Developer's Guide for XA data source interfaces and APIs

Oracle Call Interface Programmer's Guide

Oracle Data Provider for .NET Developer's Guide for Microsoft Windows

---

3.17.3 Oracle Database with Global Data Services

Global Data Services enables administrators to automatically and transparently manage client workloads across replicated databases that offer common services. A database service is a named representation of one or more database instances. Services enable you to group database workloads and route a particular work request to an appropriate instance. A global service is a service provided by multiple databases synchronized through data replication.

Global Data Services provides dynamic load balancing, failover, and centralized service management for a set of replicated databases that offer common services. The set of databases can include Oracle RAC and noncluster Oracle databases interrelated through Oracle Data Guard, databases consolidated under Oracle Multitenant, Oracle GoldenGate, or any other replication technology.

The benefits of Global Data Services include the following:

- Enables you to centrally manage global resources, including globally distributed multiple database configurations
- Provides global scalability, availability, and runtime load balancing
- Supports seamless failover
- Enables you to dynamically add databases to the GDS configuration and dynamically migrate global services
- Enables optimal resource utilization

The global services management framework is the software infrastructure for global services. This framework automates and centralizes configuration, maintenance, and monitoring of a GDS configuration, and enables load balancing and failover for services. The framework manages these virtualized resources with minimal administrative overhead, enabling the configuration to handle additional client requests.

The global services management framework is built around the following preexisting Oracle Database technologies:

- Oracle Real Application Clusters (Oracle RAC)

  Enables dynamic load balancing and workload management in a cluster

- Oracle Active Data Guard

  Enables high-performance farms of read-only databases

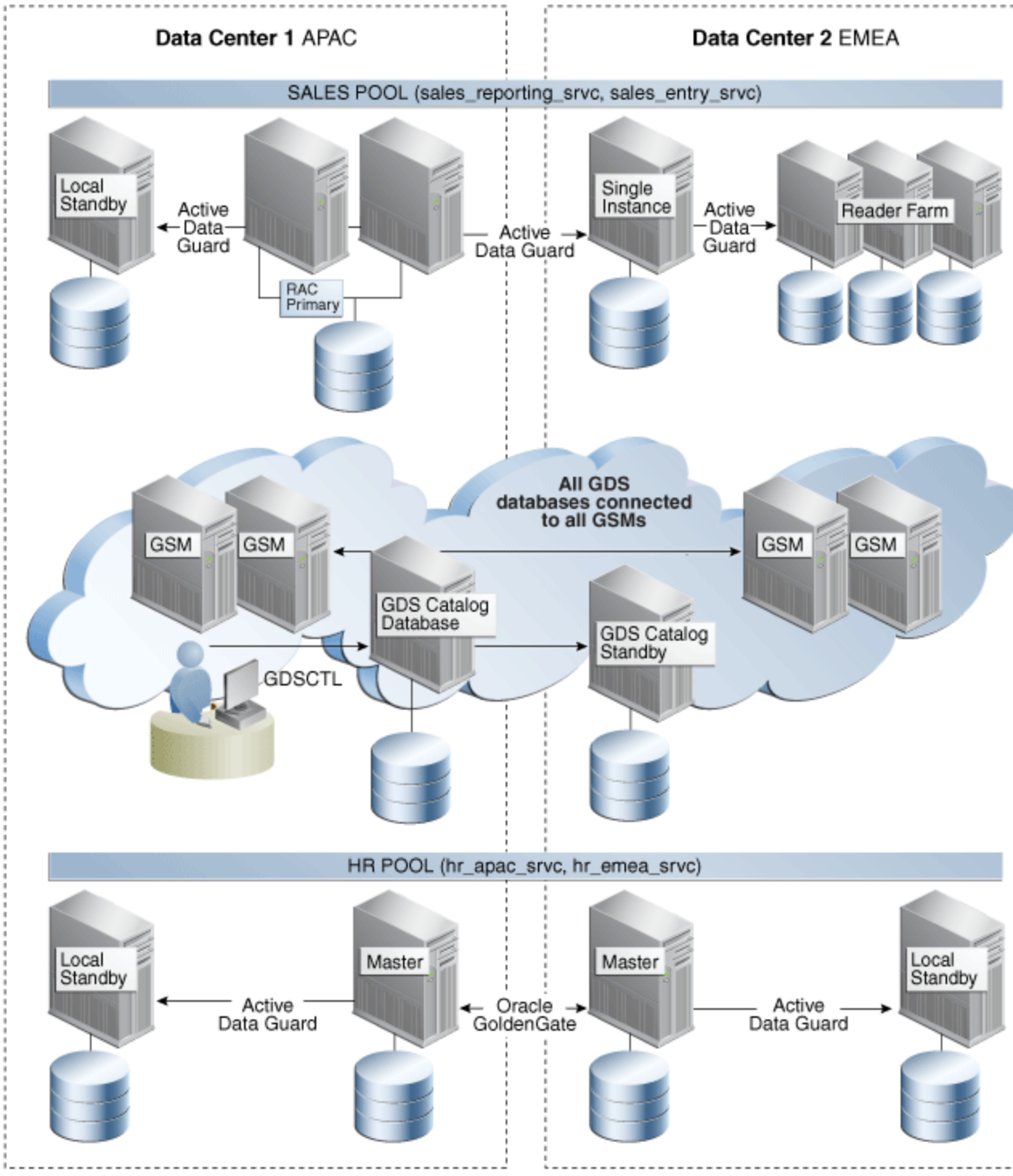
- Data Guard Broker

  Enables creation, management, and monitoring of Data Guard configurations that include a primary database and up to 30 standby databases

- Oracle GoldenGate

  Enables replication updates among multiple databases

Figure 3-3 Global Data Services Architecture



See Also:

Oracle Database Global Data Services Concepts and Administration Guide

3.18 Oracle Multitenant

Oracle Multitenant is the optimal database consolidation method from Oracle Database 12*c* onward. The multitenant architecture combines the best attributes of each of the previous consolidation methods without their accompanying tradeoffs.

Oracle Multitenant helps reduce IT costs by simplifying consolidation, provisioning, upgrades and more. This new architecture allows a container database (CDB) to hold many pluggable databases (PDBs). To applications, these PDBs appear as a standalone database, and no changes are required to the application in order to access the PDB. By consolidating multiple databases as PDBs into a single CDB, you are provided with the ability to manage "many as one". The flexibility remains to operate on PDBs in isolation should your business require it.

Oracle Multitenant is fully compliant with and takes direct advantage of high availability features such as Oracle Real Application Clusters, Oracle Data Guard, and Oracle GoldenGate, just like any non-container database (non-CDB), meaning it can be used in any of the Oracle MAA reference architectures. Grouping multiple PDBs with the same high availability requirements into the same CDB ensures that all of those PDBs and their applications are managed and protected with the same technologies and configurations.

Benefits of Using Oracle Multitenant

- High consolidation density - Many PDBs can be stored in a single CDB. These PDBs share background processes and memory structures letting you run more PDBs than you would non-CDBs, because the overhead for each non-CDB is removed or reduced. In Oracle Database 12*c* Release 2 you can store up to 4095 PDBs in a CDB. Each PDB can also have a different characterset from other PDBs within the same CDB, as long as the CDB root character set is a superset of all of the PDBs' character sets. Logical standby databases also support this mix of character sets to allow rolling upgrades with a transient logical standby database.

- Online provisioning operations, including clones, refreshable clones, and PDB relocation - A PDB can be unplugged from one CDB and plugged into another. A PDB can also be cloned either into the same CDB or into a different CDB. Cloning can be used to create a "gold image" or seed database for DBaaS or SaaS environments. This PDB can then be rapidly cloned to easily set up database environments for new customers. The following functionality is available as of Oracle Database 12*c* Release 2:

  – Near Zero Downtime PDB Relocation – This feature significantly reduces the downtime of relocating a PDB from one CDB to another by using clone functionality. The source PDB remains open and functional while the relocation takes place. The application outage is reduced to a very short window while the source PDB is brought to a consistent state, and the destination PDB is synchronized and brought online. This functionality also takes advantage of another new feature, Listener Redirects, which allows you to keep the same connect descriptor for applications and connect to the destination PDB even after it has been relocated.

  – Online provisioning and cloning – Clones of PDBs can be created without requiring the source PDB to be placed in read only-mode. The source PDB can be left in read-write mode and accessible to applications for the duration of the clone operation.

  - Refreshable Clone PDB – Clones of PDBs can be created in such a way as to be refreshed with changes with changes made to the source PDB applied either automatically at set intervals or manually. For a clone to be refreshable it must remain in read-only mode. The clone can be converted into an ordinary PDB by opening it read-write. Refreshable clones are well suited to be used as test masters for Exadata storage snapshots.

- New patching and upgrade options -When you upgrade or patch a CBD, all of the PDBs in that container are also upgraded or patched. If you need isolation, you can unplug a PDB and plug it into a CDB at a later version.

- Database backup and recovery - By consolidating multiple databases as PDBs, operations such as backup and disaster recovery are performed at the container level. Oracle Multitenant also provides the flexibility to backup and restore individual PDBs with no impact to other running PDBs in the same CDB.

- Operation with Oracle Data Guard - Data Guard configurations are maintained at the CDB level. When a Data Guard role transition (either failover or switchover) is performed, all PDBs are transitioned to the new primary database. There is no need to create or manage multiple Data Guard configurations for each PDB as would be required for single databases. Existing tools such as Data Guard Standby First Patching and Data Guard Transient Logical Rolling Upgrade can still be used to reduce downtime and are performed at the container level, so all PDBs will be maintained in a single operation. The following functionality has been added in Oracle Database 12*c* Release 2:

  - PDB Migration with Data Guard Broker – The Data Guard broker has been enhanced to provide automation for migrating PDBs from one CDB, either the primary database or the standby database, to another CDB. This can be used for straight migration of a PDB from one CDB to another running at either at the same version or a CDB running at a higher version to start the upgrade process. This automation can also be used to affect a single PDB failover by using the PDBs files at a standby database to plug into a different CDB at the same version.

  - Subset Standby - A subset standby enables users of Oracle Multitenant to designate a subset of the PDBs in a CDB for replication to a standby database. This provides a finer granularity of designating which standby databases will contain which PDBs.

- Operation with Oracle GoldenGate - All of functionality provided by Oracle GoldenGate also exists for Oracle Multitenant. GoldenGate also provides the flexibility to operate at the PDB level, allowing replication to occur for a subset of the PDBs in a CDB. GoldenGate can be used for minimal to zero downtime upgrades either at the CDB level or at an individual PDB level.

- Resource management - Just as Oracle Resource Manager can control resource utilization between single databases, it can also control resource utilization of individual PDBs in a container. This can ensure that a single PDB does not access more than its assigned share of system resources. In Oracle Database 12*c* Release 2 you have the ability to specify guaranteed minimums and maximums for SGA, buffer cache, shared pool, and PGA memory at the PDB limit.

- Operation with Oracle Flashback Database - If fast point-in-time recovery is required, the initial release of Oracle Multitenant enables using Flashback Database at the CDB level. Oracle Multitenant enables Flashback Database to be used on an individual PDB without impacting the availability of other PDBs. Flashback

Database can performed at the CDB level which will flashback all of the PDBs in the container. Starting in Oracle Database 12*c* Release 2, individual PDBs can be flashed back using the Flashback Pluggable Database feature. When flashing back an individual PDB all other PDBs remain unaffected.

See Also:

Oracle Database Administrator's Guide for information about Oracle Multitenant

Flashback Pluggable Database (page 3-37)

3.19 Oracle Sharding

Oracle Sharding is a scalability and availability feature for custom-designed OLTP applications explicitly designed to run on a sharded database.

Oracle sharding enables distribution and replication of data across a pool of Oracle databases that share no hardware or software. The pool of databases is presented to the application as a single logical database. Applications elastically scale (data, transactions, and users) to any level, on any platform, simply by adding additional databases (shards) to the pool. Scaling up to 1000 shards is supported in the first release with Oracle Database 12*c* Release 2.

Oracle Sharding provides superior run-time performance and simpler life-cycle management compared to home-grown deployments that use a similar approach to scalability. It also provides the advantages of an enterprise DBMS, including relational schema, SQL, and other programmatic interfaces, support for complex data types, online schema changes, multi-core scalability, advanced security, compression, high-availability, ACID properties, consistent reads, developer agility with JSON, and much more.

See Also:

Oracle Database Sharding Reference Architecture (page 7-19)

Oracle Database Administrator's Guide for more information about managing an Oracle Sharding environment

3.20 Oracle Restart

Oracle Restart is a new feature in Oracle 11*g* Release 2 (11.2) that enhances the availability of a single-instance (nonclustered) Oracle database and its components. Oracle Restart is used in single-instance environments only. For Oracle Real Application Clusters (Oracle RAC) environments, the functionality to automatically restart components is provided by Oracle Clusterware.

If you install Oracle Restart, it automatically restarts the database, the listener, and other Oracle components after a hardware or software failure or whenever the database's host computer restarts. It also ensures that the Oracle components are restarted in the proper order, in accordance with component dependencies.

Oracle Restart periodically monitors the health of components—such as SQL*Plus, the Listener Control utility (LSNRCTL), ASMCMD, and Oracle Data Guard—that are integrated with Oracle Restart. If the health check fails for a component, Oracle Restart shuts down and restarts the component.

Oracle Restart runs out of the Oracle Grid Infrastructure home, which you install separately from Oracle Database homes.

Integrated client failover applications depend on role based services and Fast Application Notification events, managed by Oracle clusterware, to alert the application to failures. Single instance databases must have Oracle Restart to achieve integrated client failover.

See Also:

Oracle Database Administrator's Guide for information about installing and configuring the Oracle Restart feature

Oracle Grid Infrastructure Installation Guide for your platform

3.21 Oracle Site Guard

Oracle Site Guard is a disaster-recovery solution that enables administrators to automate complete site switchover or failover.

Oracle Site Guard orchestrates and automates the coordinated failover of Oracle Fusion Middleware, Oracle Fusion Applications, and Oracle Databases. It is also extensible to include other data center software components.

Oracle Site Guard integrates with underlying replication mechanisms that synchronize primary and standby environments and protect mission critical data. It comes with a built-in support for Oracle Data Guard for Oracle database, and Oracle Sun ZFS. Oracle Site Guard can also support other storage replication technologies.

See Also:

Oracle Enterprise Manager Oracle® Site Guard Administrator's Guide

3.22 Zero Data Loss Recovery Appliance

The cloud-scale Zero Data Loss Recovery Appliance, commonly known as Recovery Appliance, is an engineered system designed to dramatically reduce data loss and backup overhead for all Oracle databases in the enterprise. Integrated with Recovery Manager (RMAN), the Recovery Appliance enables a centralized, incremental-forever backup strategy for large numbers of databases, using cloud-scale, fault-tolerant hardware and storage. The Recovery Appliance continuously validates backups for recoverability.

Recovery Appliance provides is the MAA preferred backup and recovery appliance because:

- Elimination of data loss when restoring from Recovery Appliance
- Minimal backup overhead
- Improved end-to-end data protection visibility
- Cloud-scale protection
- Integrates very well with all MAA reference architectures including Oracle Sharding tier

See Also:

Zero Data Loss Recovery Appliance (page 8-5)

Zero Data Loss Recovery Appliance Administrator's Guide

4

Oracle Database High Availability Solutions for Unplanned Downtime

Oracle Database offers an integrated suite of high availability solutions that increase availability and eliminate or minimize both planned and unplanned downtime. These solutions help enterprises maintain business continuity 24 hours a day, 7 days a week. However, the Oracle high availability solutions go beyond reducing downtime by providing solutions to increase system use on the primary and secondary systems and to help improve overall performance, scalability, and manageability.

Table 4-1 (page 4-1) describes the various Oracle high availability solutions for unplanned downtime. The table shows how the features discussed in the subsequent sections can be used to address various causes of unplanned downtime. Where several Oracle solutions are listed, the MAA recommended solution is indicated in the Oracle Solution column.

Table 4-2 (page 4-6) describes the high availability solutions in each of the MAA service-level tiers for the MAA reference architectures and multitenant architectures.

Table 4-1 Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
Site failures	Oracle Data Guard (page 3-2) (MAA recommended) and Oracle Application Failover solution	Integrated client and application failover Fastest and simplest database replication Supports all data types Zero data loss by eliminating propagation delay Oracle Active Data Guard supports read-only services and DML on global temporary tables and sequences to off-load more work from the primary Database In-Memory support
	Oracle GoldenGate (page 3-12)	Flexible logical replication solution (target is open read/write) Active-active high availability (with conflict resolution) Heterogeneous platform and heterogeneous database support
	Recovery Manager (page 3-17), Zero Data Loss Recovery Appliance (page 3-56) and Oracle Secure Backup (page 3-18)	Fully managed database recovery and integration with Oracle Secure Backup Recovery Appliance provides end-to-end data protection for backups and reduces data loss for database restores Non-real-time recovery

Table 4-1 (Cont.) Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
Instance or computer failures	Oracle Real Application Clusters and Oracle Clusterware (page 3-20) (MAA recommended)	Integrated client and application failover Automatic recovery of failed nodes and instances Lowest application brownout with Oracle Real Application Clusters
	Oracle RAC One Node (page 3-24)	Integrated client and application failover Online database relocation migrates connections and instances to another node Better database availability than traditional cold failover solutions
	Oracle Data Guard (page 3-2)	Integrated client and application failover Fastest and simplest database replication Supports all data types Zero data loss by eliminating propagation delay Oracle Active Data Guard supports read-only services and DML on global temporary tables and sequences to off-load more work from the primary Database In-Memory support
	Oracle GoldenGate (page 3-12)	Flexible logical replication solution (target is open read/write) Active-Active high availability (with conflict resolution) Heterogeneous platform and heterogeneous database support
Storage failures	Oracle Automatic Storage Management (page 3-25) (MAA recommended)	Mirroring and online automatic rebalancing places redundant copies of the data in separate failure groups.
	Oracle Data Guard (page 3-2) (MAA recommended)	Integrated client and application failover Fastest and simplest database replication Supports all data types Zero data loss by eliminating propagation delay Oracle Active Data Guard supports read-only services and DML on global temporary tables and sequences to off-load more work from the primary Database In-Memory support

Table 4-1 (Cont.) Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
	Recovery Manager (page 3-17) with Fast Recovery Area (page 3-27), and Zero Data Loss Recovery Appliance (page 3-56) (MAA recommended)	Fully managed database recovery and managed disk and tape backups
	Oracle GoldenGate (page 3-12)	Flexible logical replication solution (target is open read/write) Active-active high availability (with conflict resolution) Heterogeneous platform and heterogeneous database support
Data corruption	Corruption Prevention, Detection, and Repair (page 3-27) (MAA recommended)	Different levels of data and redo block corruption prevention and detection at the database level
	Database initialization settings such as `DB_BLOCK_CHECKING`, `DB_BLOCK_CHECKSUM`, and `DB_LOST_WRITE_PROTECT`	

Table 4-1 (Cont.) Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
Data corruption	Oracle Data Guard (page 3-2) (MAA recommended) Oracle Active Data Guard Automatic Block Repair `DB_LOST_WRITE_PROTECT` initialization parameter	In a Data Guard configuration with an Oracle Active Data Guard standby, physical block corruptions detected by Oracle at a primary database are automatically repaired using a good copy of the block retrieved from the standby, and vice versa. The repair is transparent to the user and application. Strong database isolation of data corruptions with Oracle Active Data Guard. With MAA recommended initialization settings, Oracle Active Data Guard and Oracle Exadata Database Machine, achieve most comprehensive full stack corruption protection. With `DB_LOST_WRITE_PROTECT` enabled, a lost write that occurred on the primary database is detected either by the physical standby database or during media recovery of the primary database, recovery is stopped to preserve the consistency of the database. Failing over to the standby database using Data Guard will result in some data loss. With Oracle Database 12*c* Release 1 (12.1.0.2) and Data Guard Broker, Data Guard Broker's `PrimaryLostWrite` property supports SHUTDOWN and CONTINUE (as in Oracle Database 11*g* Release 2 (11.2.0.4)), plus FAILOVER and FORCEFAILOVER options when lost writes are detected on the primary database. See *Oracle Data Guard Broker* If a lost write is detected on the standby database, you can restore the affected file and restart Redo Apply if the lost write is isolated and the hardware problem is corrected. `DB_LOST_WRITE_PROTECT` initialization parameter provides lost write detection. **Note:** Lost writes can corrupt the entire database, which may require that you rebuild the affected database after resolving the hardware issue. Database In-Memory support

Table 4-1 (Cont.) Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
	Dbverify, Analyze, Data Recovery Advisor (page 3-30) and Recovery Manager (page 3-17), Zero Data Loss Recovery Appliance (page 3-56), and ASM Scrub with Fast Recovery Area (page 3-27) (MAA recommended)	These tools allow the administrator to execute manual checks to help detect and potentially repair from various data corruptions. Dbverify and Analyze conducts physical block and logical intra-block checks. Analyze can conduct inter-object consistency checks. Data Recovery Advisor automatically detects data corruptions and recommends the best recovery plan. RMAN operations can conduct both physical and inter-block logical checks. RMAN can execute online block-media recovery using flashback logs, backups, or the standby database to help recover from physical block corruptions. Recovery Appliance does periodic backup validation that helps ensure that your backups are valid. Recovery Appliance allows you to input your recovery window requirements and alerts you when those SLAs cannot be met with your existing backups managed by Recovery Appliance. ASM Scrub detects and attempts to repair physical and logical data corruptions with the ASM pair in normal and high redundancy disks groups.
Data corruption	Oracle Exadata Database Machine (page 8-1) and Oracle Automatic Storage Management (page 3-25) (MAA recommended) DIX + T10 DIF Extensions (MAA recommended where applicable)	If Oracle ASM detects a corruption and has a good mirror, Oracle ASM returns the good block and repairs the corruption during a subsequent write I/O. Exadata provides implicit HARD enabled checks to prevent data corruptions caused by bad or misdirected storage I/O. Exadata provides automatic HARD disk scrub and repair. Detects and fixes bad sectors. DIX +T10 DIF Extensions provides end to end data integrity for reads and writes through a checksum validation from a vendor's host adapter to the storage device
	Oracle GoldenGate (page 3-12)	Flexible logical replication solution (target is open read/write). Logical replica can be used as a failover target if partner replica is corrupted. Active-active high availability (with conflict resolution) Heterogeneous platform and heterogeneous database support
Human errors	Oracle Security Features (page 3-32) (MAA recommended)	Restrict access to prevent human errors

Table 4-1 (Cont.) Outage Types and Oracle High Availability Solutions for Unplanned Downtime

Outage Scope	Oracle Solution	Benefits
	Oracle Flashback Technology (page 3-33) (MAA recommended)	Fine-grained error investigation of incorrect results Fine-grained and database-wide rewind and recovery capabilities
Delays or slow downs	Oracle Database and Oracle Enterprise Manager Oracle Data Guard (page 3-2) (MAA recommended) and Oracle Application Failover solution	Oracle Database automatically monitors for instance and database delays or cluster slow downs and attempts to remove blocking processes or instances to prevent prolonged delays or unnecessary node evictions. Oracle Enterprise Manager or a customized application heartbeat can be configured to detect application or response time slowdown and react to these SLA breaches. For example, you can configure the Enterprise Manager Beacon to monitor and detect application response times. Then, after a certain threshold expires, Enterprise Manager can call the Data Guard `DBMS_DG.INITIATE_FS_FAILOVER` PL/SQL procedure to initiate a failover. See the section about "Application Initiated Fast-Start Failover" in *Oracle Data Guard Broker*. Database In-Memory support
File system data	Oracle Replication Technologies for Non-Database Files (page 3-38)	Enables full stack failover that includes non-database files

If you are managing many databases in DBaaS, we recommend using the MAA tiers and Oracle Multitenant as described in Oracle MAA Reference Architectures (page 2-5). Table 4-2 (page 4-6) identifies various unplanned outages that can impact a database in multitenant architecture. It also identifies the Oracle HA solution to address that outage that is available in each of the HA tiers.

Table 4-2 Unplanned Outage Matrix for MAA Reference Architectures and Multitenant Architectures

Event	Solutions by MAA Tier	Recovery Window (RTO)	Data Loss (RPO)
Instance Failure	BRONZE: Oracle Restart	Minutes if instance can restart	Zero
	SILVER: Oracle RAC or optionally Oracle RAC One Node	Seconds with Oracle RAC Minutes with Oracle RAC One Node	Zero
	GOLD: Oracle RAC	Seconds	Zero

Table 4-2 (Cont.) Unplanned Outage Matrix for MAA Reference Architectures and Multitenant Architectures

Event	Solutions by MAA Tier	Recovery Window (RTO)	Data Loss (RPO)
	PLATINUM: Oracle RAC with Application Continuity	Zero Application Outage	Zero
Permanent Node Failure (but storage available)	BRONZE: Restore and recover	Hours to Day	Zero
	SILVER: Oracle RAC	Seconds	Zero
	SILVER: Oracle RAC One Node	Minutes	Zero
	GOLD: Oracle RAC	Seconds	Zero
	PLATINUM: Oracle RAC with Application Continuity	Zero Application Outage	Zero
Storage Failure	ALL: Automatic Storage Management	Zero downtime	Zero
Data corruptions	BRONZE/ SILVER: Basic protection Some corruptions require recover restore and recovery of pluggable database (PDB), entire multitenant container database (CDB) or non-container database (non-CDB)	Hour to Days	Since last backup if unrecoverable Zero or Near Zero with Recovery Appliance
	GOLD/ PLATINUM: Comprehensive corruption protection and Auto Block Repair with Oracle Active Data Guard	Zero with auto block repair Seconds to minutes if corruption due to lost writes and using Data Guard Fast Start failover.	Zero unless corruption due to lost writes

Table 4-2 (Cont.) Unplanned Outage Matrix for MAA Reference Architectures and Multitenant Architectures

Event	Solutions by MAA Tier	Recovery Window (RTO)	Data Loss (RPO)
Human error	ALL: Logical failures resolved by flashback drop, flashback table, flashback transaction, flashback query flashback pluggable database, and undo.	Dependent on detection time but isolated to PDB and applications using those objects.	Dependent on logical failure
	All: Comprehensive logical failures impacting an entire database and PDB that requires RMAN point in time recovery (PDB) or flashback pluggable database	Dependent on detection time	Dependent on logical failure
	GOLD/ PLATINUM: With Oracle GoldenGate, you can fail over just one PDB	Dependent on detection time but actual failover can take seconds	Dependent on logical failure
Database unusable, system, site or storage failures, wide spread corruptions or disasters	BRONZE/ SILVER: Restore and recover	Hours to Days	Since last backup Zero or near zero with Recovery Appliance
	GOLD: Fail over to secondary (Oracle Active Data Guard or Oracle GoldenGate)	Seconds	Zero to Near Zero
	PLATINUM: Active Data Guard Failover with Application Continuity	Zero Application Outage	Zero

Table 4-2 (Cont.) Unplanned Outage Matrix for MAA Reference Architectures and Multitenant Architectures

Event	Solutions by MAA Tier	Recovery Window (RTO)	Data Loss (RPO)
Performance Degradation	ALL: Oracle Enterprise Manager for monitoring and detection, Database Resource Management for Resource Limits and ongoing Performance Tuning	No downtime but degraded service	Zero

See Also:

High Availability Architectures (page 7-1) for tables summarizing the attainable recovery times for all of the types of unplanned downtime for each Oracle high availability reference architecture

5

Oracle Database High Availability Solutions for Planned Downtime

Planned downtime can be just as disruptive to operations as unplanned downtime. This is especially true for global enterprises that must support users in multiple time zones, or for those that must provide Internet access to customers 24 hours a day, 7 days a week.

In the past, planned downtime was necessary to perform the following activities:

- Periodic maintenance—such as patching or reconfiguring the system to update a database, application, operating system, middleware, or network
- New deployments—such as to perform major upgrades or new rollouts of the hardware, database, application, operating system, middleware, or network

This chapter contains the following topics:

5.1 High Availability Solutions for Migration

Table 5-1 (page 5-1) describes at a high level the high availability solutions for migration. Each solution is described in the sections following the table.

Table 5-1 High Availability Solutions for Migration

Migration Type	Oracle Recommended Solution	Solution Description	Outage Time
Migrate database to a different platform	Oracle Data Guard (page 3-2), Data Pump, Recovery Manager (page 3-17), and Oracle GoldenGate (page 3-12)	Platform Migration (page 5-2)	Seconds to minutes for Data Guard or Oracle GoldenGate Minutes to hours for Data Pump and Recovery Manager

Table 5-1 (Cont.) High Availability Solutions for Migration

Migration Type	Oracle Recommended Solution	Solution Description	Outage Time
Migrate database to a different character set	Oracle GoldenGate (page 3-12)	Database Migration to a Different Character Set (page 5-8)	Seconds to minutes
Migrate to pluggable databases or another pluggable database Remote cloning of pluggable databases	Data Pump, Recovery Manager (page 3-17), and Oracle GoldenGate (page 3-12) SQL*Plus	Migrating to Multitenant Architecture (page 5-9)	Seconds to minutes for GoldenGate Minutes to hours for Data Pump and Recovery Manager Minutes to hours for remote cloning of pluggable databases
Migrate storage	Oracle Automatic Storage Management (page 3-25) and Oracle Data Guard (page 3-2)	Migration to Oracle ASM Storage (page 5-9)	No downtime for Oracle ASM Seconds to minutes for Data Guard
Migrate database from a single-instance system to an Oracle RAC cluster	Oracle Data Guard (page 3-2)	Migrating a Database from a Single-Instance System to an Oracle RAC Cluster (page 5-11)	Seconds to minutes

5.1.1 Platform Migration

Migrating a database to a different platform is required when you move an existing database to a system that runs a different operating system than the current system. For example, database migration is required when moving a database from Microsoft Windows to Linux, or from AIX or HP-UX to Oracle Exadata Database Machine running Oracle Linux. Database migration options are highly dependent on the source database platform and source database version. Database migration to a different platform is accomplished with one of the following solutions:

- Data Guard heterogeneous physical standby
- Data Pump full transportable export/import
- Data Pump tablespace transportable export/import
- Recovery Manager cross-platform transport of a PDB using inconsistent backups
- Recovery Manager cross-platform transport of tablespaces in a PDB

The following features can be used in combination with the migration solutions previously described to reduce database migration downtime:

- Recovery Manager cross-platform inconsistent tablespace transportation

- Recovery Manager cross-platform transport of a PDB using inconsistent backups
- Oracle GoldenGate

5.1.1.1 Migrating a Database to Oracle Exadata Database Machine or Oracle SuperCluster

Database migration to Oracle Exadata Database Machine or Oracle SuperCluster uses the same methods as a database migration across platforms, as described in this section. The target platform for Oracle Exadata and Oracle SuperCluster is described in Table 5-2 (page 5-3).

Table 5-2 Platform Migration to an Engineered System

Engineered System	Database Platform
Oracle Exadata Database Machine	Oracle Linux x86 64-bit (little endian)
Oracle SuperCluster	Oracle Solaris SPARC (big endian)

5.1.1.2 Platform Migration Solutions

Table 5-3 (page 5-3) lists the recommended solutions to use for the database migration scenarios. Each solution is described in the sections following the table.

Table 5-3 Database Migration Scenarios and Solutions

Database Migration Scenario	Solution to Use
Migrate to a platform that is the same endian format	1. Recovery Manager Cross-Platform Transport of Tablespaces in a PDB 2. Heterogeneous Data Guard Configurations (page 5-4) 3. Recovery Manager Cross-Platform Transport of a PDB Using Inconsistent Backups 4. Recovery Manager Cross-Platform Transport of a Closed PDB 5. Data Pump Full Transportable Export/Import (page 5-4) if Data Guard Heterogeneous Physical Standby cannot be used 6. Data Pump tablespace transportable export/import if Data Pump full transportable export/import cannot be used 7. Oracle Golden Gate if Data Guard Heterogeneous Physical Standby cannot be used, and lower downtime than Data Pump can provide is required

Table 5-3 (Cont.) Database Migration Scenarios and Solutions

Database Migration Scenario	Solution to Use
Migrate to a platform that is a different endian format	1. Data Pump Full Transportable Export/Import (page 5-4) 2. Data Pump tablespace transportable export/import if Data Pump full transportable export/import cannot be used 3. Oracle Golden Gate if lower downtime than Data Pump can provide is required

5.1.1.2.1 Heterogeneous Data Guard Configurations

Data Guard supports running a physical standby database on a different platform than the primary system for a limited number of platform combinations (for example, Windows to Linux). Migration between platforms that support a heterogeneous primary/standby combination is accomplished with a simple Data Guard switchover operation. The following criteria must be met to use this method:

- The platform combination must be listed as supported in My Oracle Support Note 413484.1.
- The source database and target database must be the same Oracle Database release.

See Also:

My Oracle Support Note 413484.1 at `http://support.oracle.com/`

5.1.1.2.2 Data Pump Full Transportable Export/Import

You can use the full transportable export/import feature to copy an entire database from one platform to another. You can use Data Pump to produce an export dump file, transport the dump file and the data files for user-defined tablespaces to the target database if necessary, and then import the export dump file. Full transportable exports are supported from a source database running Oracle Database 11*g* release 2 (11.2.0.3) or later.

See "Transporting Data" in the *Oracle Database Administrator's Guide* for information about the general limitations of transporting data and limitations specific to full transportable export/import.

A full transportable export exports all objects and data necessary to create a complete copy of the database. A mix of data movement methods is used:

- Objects residing in transportable tablespaces have only their metadata unloaded into the dump file set; the data itself is moved when you copy the data files to the target database. The data files that must be copied are listed at the end of the log file for the export operation.
- Objects residing in non-transportable tablespaces (for example, SYSTEM and SYSAUX) have both their metadata and data unloaded into the dump file set, using direct path unload and external tables.

The length of time required to migrate a database to a new platform depends on the following factors:

- Data size
- Metadata size

The high-level steps to migrate a database are as follows:

1. Create a new, empty database on the target platform.
2. Stop the application (read-only access to the data is still permitted.)
3. Make the user tablespaces read only in the source database.
4. Perform full transportable export of the source database.
5. Transfer export dump file and data files for user tablespaces to the destination system.
6. Use RMAN to convert the data files to the endian format of the destination system (if necessary).
7. Perform full transportable import into the target database.
8. Make user tablespaces read/write in the target database.
9. Start the application, connecting to the target database.

To reduce migration downtime use Recovery Manager cross-platform inconsistent tablespace transportation in conjunction with Data Pump full transportable export/import.

See Also:

See Methods to Reduce Database Migration Downtime (page 5-6) for additional information.

5.1.1.3 Data Pump Tablespace Transportable Export/Import

You can use the tablespace transportable export/import feature to copy all user-defined tablespaces from a database on one platform to a database running on another. A tablespace transportable export exports only the metadata for the tables (and their dependent objects) within a specified set of user-defined tablespaces. The tablespace data files are copied in a separate operation. Then, a transportable tablespace import is performed to import the dump file containing the metadata and to specify the data files to use. Tablespace transportable exports are supported between different platforms for version 10.0 compatible or later source and target databases.

See "Transporting Data" in *Oracle Database Administrator's Guide* for information about the general limitations of transporting data and limitations specific to tablespace transportable export/import.

The length of time required to migrate a database to a new platform depends on the following factors:

- Data size

- Metadata size

The high-level steps are as follows:

1. Create a new, empty database on the target platform.
2. Stop the application (read-only access to the data is still permitted).
3. Import objects required for transport operations into the target database.
4. Make the user tablespaces read only in the source database.
5. Perform full transportable export of the source database.
6. Transfer export dump file and data files for user tablespaces to the destination system.
7. Use RMAN to convert the data files to the endian format of the destination system (if necessary).
8. Perform tablespace transportable import of all user tablespaces.
9. Export and import database objects that could not be transported.
10. Make user tablespaces read/write in the target database.
11. Start the application, connecting to the target database.

To reduce migration downtime use Recovery Manager cross-platform inconsistent tablespace transportation in conjunction with Data Pump tablespace transportable export/import.

See Also:

Methods to Reduce Database Migration Downtime (page 5-6)

5.1.1.4 Methods to Reduce Database Migration Downtime

The methods described in the following sections can be used in combination with the migration methods previously described to reduce database migration downtime:

- Recovery Manager Cross-Platform Inconsistent Tablespace Transportation (page 5-6)
- Reducing Migration Downtime with Oracle GoldenGate (page 5-8)

5.1.1.4.1 Recovery Manager Cross-Platform Inconsistent Tablespace Transportation

The downtime required to migrate a database using Data Pump full or tablespace transportable export/import is primarily determined by the following two factors:

- Data size
- Metadata size

To reduce migration downtime use Recovery Manager cross-platform inconsistent tablespace transportation in conjunction with Data Pump full or tablespace transportable export/import. Migration downtime is reduced by allowing most data to be moved while the source database remains online. When you use Recovery Manager cross-platform inconsistent tablespace transportation in conjunction with

Data Pump full or tablespace transportable export/import, the downtime required is primarily determined by the following:

- Data change rate
- Metadata size

RMAN enables you to transport the majority of the database to the target system while the database on the source system remains online by creating an inconsistent backup of the user-defined tablespaces on the source system and restoring it on the target system. Because the time for the initial backup/restore operation can be significant, the data files produced by the inconsistent backup can be rolled forward one or more times using a cross-platform incremental backup. To make the data files consistent in order to complete the transportation, you then apply a final cross-platform incremental backup, taken when the tablespaces are in read-only mode. The final step is to complete the migration using Data Pump full or tablespace transportable export/import.

The high-level steps to transport the database to the target system are as follows:

Phase 1: Prepare phase

1. Create an RMAN cross-platform inconsistent backup of all user-defined tablespaces.
2. Restore the cross-platform inconsistent backup on the target system. The target data files created on the target system are called foreign data files.

Phase 2: Roll forward phase

1. Create an RMAN cross-platform incremental backup of all user-defined tablespaces.
2. Recover the foreign data files on the target system by applying the cross-platform incremental backup.

   The roll forward phase is repeated as many times as necessary to catch foreign data files up to the source database.

Phase 3: Transport phase

1. Stop the application.
2. Make the user-defined tablespaces read-only.
3. Repeat Phase 2 (Roll forward phase) one final time.
4. Migrate the database using Data Pump full transportable export/import, or migrate the user-defined tablespaces using Data Pump tablespace transportable export/import.
5. Start the application, connecting to the target database.

See Also:

Oracle Database Backup and Recovery User's Guide

5.1.1.4.2 Reducing Migration Downtime with Oracle GoldenGate

Use Oracle GoldenGate to reduce migration downtime. Migration downtime is reduced by allowing the target database to be created and kept synchronized while the source database remains online. When you use Oracle GoldenGate the downtime required is the length of time it takes to reconnect the application to the target database.

The high-level steps are as follows:

1. Start an Oracle GoldenGate Extract group to extract ongoing data changes.
2. Create the target database using Data Pump full transportable export/import or Data Pump tablespace transportable export/import.
3. Start the Oracle GoldenGate Replicat group to re-synchronize rows that were changed while the target database was being created.
4. Stop the application.
5. Wait for Oracle GoldenGate Replicat to catch up and apply any remaining changes from the trail file.
6. Start the application, connecting to the target database.

See Also:

Oracle GoldenGate (page 3-12)

"Oracle GoldenGate Best Practices: Instantiation from an Oracle Source Database" at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1276058.1`

5.1.2 Database Migration to a Different Character Set

Use Oracle GoldenGate to reduce character set migration downtime. Character set migration downtime is reduced by allowing the target database to be created and kept synchronized while the source database remains online. When you use Oracle GoldenGate the downtime required is the length of time it takes to reconnect the application to the target database.

The high-level steps are as follows:

1. Create the empty target database with the desired character set.
2. Start a change-synchronization Extract group to extract ongoing data changes.
3. Perform a Data Pump full non-transportable export/import. The data is automatically converted to the new character set during the import process.
4. Start the change-synchronization Replicat group to resynchronize rows that were changed while the target database was being created.
5. Stop the application.
6. Start the application, connecting to the target database.

See Also:

Oracle GoldenGate (page 3-12)

5.1.3 Migrating to Multitenant Architecture

Migrating a non-container database (non-CDB), or an unplugged pluggable database (PDB), to a PDB in a target CDB is accomplished with one of the following solutions:

- `CREATE PLUGGABLE DATABASE` statement
- Data Pump full transportable export/import
- Data Pump tablespace transportable export/import

Table 5-4 Pluggable Database Migration Solutions

Solution	When to Use
`CREATE PLUGGABLE DATABASE` statement	Use when the non-CDB or unplugged PDB is Oracle Database 12*c* and it is the same endian format as the target CDB.
Data Pump full transportable export/import	Use when the non-CDB is Oracle Database 11*g* release 2 (11.2.0.3) or later, or the non-CDB is a different endian format than the CDB.
Data Pump tablespace transportable export/import	Use when the non-CDB version is earlier than Oracle Database 11*g* release 2 (11.2.0.3).
Remote cloning of a pluggable database	Use when you want to clone an existing PDB from one CDB into a different CDB.

The methods described in the following sections can be used in combination with the solutions previously described to reduce migration downtime:

- Recovery Manager Cross-Platform Inconsistent Tablespace Transportation (page 5-6)
- Reducing Migration Downtime with Oracle GoldenGate (page 5-8)

The use of these features to reduce downtime when migrating to multitenant architecture is the same as the use of these features to reduce downtime for database migration to a different platform.

See Also:

MAA white paper "High Availability Best Practices for Database Consolidation – The Foundation for Database-as-a-Service" at `http://www.oracle.com/technetwork/database/availability/maa-consolidation-2186395.pdf`

Oracle Database Administrator's Guide

5.1.4 Migration to Oracle ASM Storage

The following sections describe migration to Oracle ASM storage

- Migrate to Oracle ASM-Managed Storage Using Data Guard (page 5-10)
- Migrate to New Storage Using Oracle ASM Rebalance (page 5-10)
- Migrate to Oracle ASM by Relocating Online Data Files (page 5-10)

5.1.4.1 Migrate to Oracle ASM-Managed Storage Using Data Guard

If a database currently does not use Oracle ASM to manage storage, then you can migrate all or part of the database into Oracle ASM, thereby simplifying database administration. Use Data Guard to minimize downtime when migrating to Oracle ASM. The high-level steps are as follows:

1. Create a standby database using Oracle ASM storage
2. Perform a Data Guard switchover

See Also:

- My Oracle Support note 1617946.1 for information about standby creation
- *Oracle Automatic Storage Management Administrator's Guide* for information about performing Oracle ASM data migration with RMAN

5.1.4.2 Migrate to New Storage Using Oracle ASM Rebalance

If an existing storage device is already managed by Oracle ASM, and it will be replaced with new storage, and the new storage is connected to the existing database server or cluster, then use Oracle ASM to perform the storage migration. Oracle ASM enables you to add all disks from new storage and drop all disks from existing storage. Oracle ASM automatically rebalances and migrates data to the new storage while the database remains operational. Before removing the existing storage device, ensure that the rebalancing is complete.

The high-level steps are as follows:

1. Connect and configure the new storage on the existing system.
2. Add the new storage to Oracle ASM and drop the original storage from Oracle ASM using Oracle ASM commands.
3. Wait for the Oracle ASM rebalance operation that moves the data to the new storage to complete.
4. Disconnect the original storage device.

See Also:

Oracle Automatic Storage Management Administrator's Guide

5.1.4.3 Migrate to Oracle ASM by Relocating Online Data Files

See Renaming and Relocating Online Datafiles (page 5-11) for additional information about relocating data files to ASM with the `ALTER DATABASE MOVE DATAFILE` SQL statement.

5.1.5 Migrating a Database from a Single-Instance System to an Oracle RAC Cluster

You can use Data Guard or execute the conversion in place when migrating from a non-clustered system running single-instance Oracle Database to a clustered environment running Oracle RAC. Required downtime depends on the option you use.

See Also:

Oracle Real Application Clusters Administration and Deployment Guide for information about converting single-instance Oracle databases to Oracle RAC and Oracle RAC One Node

MAA white paper Rapid Oracle RAC Standby Deployment

5.2 Dynamic and Online Resource Provisioning

For system and database changes, use the dynamic resource provisioning features that are discussed in the following sections:

- Renaming and Relocating Online Datafiles (page 5-11)
- Dynamic Reconfiguration of the Database (page 5-12)
- Automatic Tuning of Memory Management (page 5-13)
- Automated Distribution of Data Files, Control Files, and Log Files (page 5-14)

5.2.1 Renaming and Relocating Online Datafiles

Every data file is either online (available) or offline (unavailable). You can alter the availability of individual data files or temporary files by taking them offline or bringing them online. Offline data files cannot be accessed until they are brought back online.

Starting in Oracle Database 12*c* Release 1 (12.1) you can use SQL to move an online data file from one physical file to another while the database is open and accessing the file.

You can use the `ALTER DATABASE MOVE DATAFILE` SQL statement to rename or relocate online datafiles. This statement enables you to rename or relocate a datafile while the database is open and users are accessing the data file.

When you rename or relocate online data files, the pointers to the data files, as recorded in the database control file, are changed. The files are also physically renamed or relocated at the operating system level.

You might rename or relocate online data files because you want to allow users to access the data files when you perform one of the following tasks:

- Move the data files from one type of storage to another
- Move data files that are accessed infrequently to lower cost storage
- Make a tablespace read-only and move its data files to write-once storage
- Move a database into Oracle Automatic Storage Management (Oracle ASM)

- Rename a data file to a more descriptive name

See Also:

Oracle Database Administrator's Guide to learn how to rename or relocate online data files.

5.2.2 Dynamic Reconfiguration of the Database

Oracle continues to broaden support for dynamic reconfiguration of the database, enabling it to adapt to changes in hardware demands without any service interruptions.

Oracle Clusterware online resource attribute modification allows certain attributes of a resource to be modified without the need to restart the resource for the change to take effect. Online resource attribute modification is available using SRVCTL and CRSCTL commands.

Oracle Database dynamically accommodates various changes to hardware and database configurations by providing the ability to:

- Add and remove processors from a symmetric multiprocessing (SMP) server
- Add and remove nodes and instances in an Oracle RAC environment
- Dynamically increase and decrease its shared memory allocation and automatically tune memory online using automatic shared memory management
- Add and remove database disks online without disturbing database activities using Oracle ASM
- Add and remove storage arrays or Exadata Cells online without disturbing database activities using Oracle ASM
- Change existing Exadata Elastic system by expanding with additional Exadata Database Servers, Exadata Storage Servers, or Exadata Racks without downtime
- Automatically rebalance the I/O load across the database storage using Oracle ASM
- Move data files online when adding or dropping disks using Oracle ASM, which automatically rebalances database storage whenever the storage configuration is changed
- Dynamically control database session resource consumption using Resource Manager consumer groups and plans
- Change almost all initialization parameters without shutting down the instance, by using either of the following SQL*Plus statements:
  - The `ALTER SESSION` statement changes the value of a parameter during a session.
  - The `ALTER SYSTEM` statement changes the value of a parameter in all sessions of an instance for the duration of the instance.

These capabilities provide no-cost system changes and capacity on-demand provisioning, both of which are fundamental requirements of enterprise grid computing.

See Also:

Oracle Database Administrator's Guide for information about platforms that support Automatic Memory Management

Oracle Exadata Database Machine Maintenance Guide for information about changing elastic configurations

5.2.3 Automatic Tuning of Memory Management

Two memory management initialization parameters, `MEMORY_TARGET` and `MEMORY_MAX_TARGET`, enable automatic management of the System Global Area (SGA), Program Global Area (PGA), and other memory required to run Oracle Database.

The `MEMORY_MAX_TARGET` parameter specifies the maximum value to which the `MEMORY_TARGET` can grow dynamically.

Table 5-5 MEMORY_MAX_TARGET and MEMORY_TARGET

if...	And...	Then...
You omit `MEMORY_MAX_TARGET`	You omit `MEMORY_TARGET`	The initialization parameters are left at their default values (0), and Oracle Database does not automatically tune memory.
You omit `MEMORY_MAX_TARGET`	Include a value for `MEMORY_TARGET`	The database automatically sets `MEMORY_MAX_TARGET` to the value of `MEMORY_TARGET`.
You omit `MEMORY_TARGET`	Include a value for `MEMORY_MAX_TARGET`	The `MEMORY_TARGET` parameter defaults to zero.

Oracle Database uses a noncentralized policy to free and acquire memory in each subcomponent of the SGA and the PGA. Oracle Database automatically tunes memory by prompting the operating system to transfer granules of memory from less needy to more needy components. The granularity of the memory transfer is dependent on the current free memory and the amount of memory the operating system requires to maintain a basic level of service.

Note:

Automatic memory management with the `MEMORY_TARGET` and `MEMORY_MAX_TARGET` initialization parameters is supported on Linux, Windows, Solaris, HP-UX, and AIX. See *Oracle Database Administrator's Guide* for more information about all supported platforms.

5.2.4 Automated Distribution of Data Files, Control Files, and Log Files

Oracle ASM automatically distributes data files, control files, and log files across all available disks. Database storage is rebalanced whenever the storage configuration changes, including adding and removing disks, Exadata Cells, or storage arrays. Oracle ASM provides redundancy through the mirroring of database files, and provides optimal performance by automatically striping database files across available disks.

See Also:

Oracle Database Concepts and *Oracle Automatic Storage Management Administrator's Guide* for more information about Oracle ASM

5.3 Online Reorganization and Redefinition

One way to enhance availability and manageability is to allow user access to the database during a data reorganization operation. The Online Reorganization and Redefinition feature in Oracle Database offers administrators significant flexibility to modify the physical attributes of a table and transform both data and table structure while allowing user access to the database. This capability improves data availability, query performance, response time, and disk space usage. All of these are important in a mission-critical environment and make the application upgrade process easier, safer, and faster.

The Online Reorganization and Redefinition architecture provides the following benefits:

- Online table reorganization and redefinition:
  - Change any physical attribute of the table online, including moving the table to a new location, partitioning the table, converting the table from one organization (such as heap-organized) to another (such as index-organized), and enabling data compression (Advanced Row Compression).
  - Change many logical attributes such as column names, types, and sizes. Columns can be added, deleted, or merged. However, you cannot modify the primary key of the table.
  - `REDEF_TABLE` procedure, which automates online table reorganization of a single table in one command (new in Oracle Database 12*c*).
  - Set an unused column online (new in Oracle Database 12*c*).
- Online index operations:
  - Create indexes online and analyze them simultaneously. You can also use online repair of the physical guess component of logical row IDs (used in secondary indexes and in the mapping table for index-organized tables).
  - Reorganize an index-organized table and secondary indexes online to eliminate the reorganization maintenance window. Secondary indexes support efficient use of block hints (physical guesses). You can also perform online repair of invalid physical guesses of logical row IDs stored in secondary indexes on an index-organized table.

  - Reorganize an index-organized table or table partition without rebuilding its secondary indexes, resulting in a short reorganization maintenance window.
  - New in Oracle Database 12*c*: drop index online, alter index visible/invisible, alter index unusable online, and drop constraint online.
- Maintain indexes during online moves and splits of partitioned tables
- Maintain indexes during online moves of non-partitioned tables
- Enable Basic Compression, Advanced Row Compression and Hybrid Columnar Compression for your partitions online if you have an Advanced Compression Option license.
- Online reorganization support for advanced queues, clustered tables, materialized views, and abstract data types (objects)
- Fast `ADD COLUMN` operations with default value (does not need to update all rows to a default value)
- Speedier application migration and testing with invisible indexes:
  - Speeds up migration with explicit hints, then drops when finished
  - Prevents premature use of newly created indexes
  - Tests effects of `DROP INDEX`, making the index visible if needed, thus there is no need for an index rebuild
- Online index builds with no pause to perform DML operations (no exclusive DML locks are required)
- Easier table DDL operations online (there is an option to wait for active DML operations instead of stopping)
- Redefinition of multiple partitions in a single redefinition session to reduce the completion time to redefine multiple partitions (new in Oracle Database 12*c*).
- Redefinition of tables that have Virtual Private Database (VPD) policies defined on them to eliminate downtime for redefining these tables (new in Oracle Database 12*c*).
- Improved `SYNC_INTERIM_TABLE` performance with optimized Materialized View Log processing (new in Oracle Database 12*c*).
- Improved resilience of `FINISH_REDEF_TABLE` with better lock management (new in Oracle Database 12*c*).

The ability to modify physical table attributes and transform both data and table structure has been available since the Oracle8*i* release. Table 5-6 (page 5-16) provides a comprehensive list of data reorganization capabilities.

Table 5-6 New Data Reorganization Capabilities by Release

Action	Oracle Database 9*i*	Oracle Database 10*g* Release 1	Oracle Database 10*g* Release 2	Oracle Database 11*g*	Oracle Database 12*c*
Online reorganization using the package `DBMS_REDEFINITION`	Modify table storage parameters Move the table to a different tablespace Add support for parallel queries Add or drop partitioning support Re-create the table to avoid fragmentation Change from a table to an index-organized table, or vice-versa Add or drop a column Transform a column using a function	Clones grants, constraints, and triggers Convert a `LONG` to a `LOB` Reorganize using a unique key Specify columns to order table by	Reorganize a single partition Advanced queue and clustered tables Table containing an ADT Retain and clone statistics Clone check and not null constraints Copies dependent objects for nested tables	Table with materialized view logs or materialized views No recompilation of dependent objects when redefinition does not logically affect objects	Redefinition of multiple partitions in a single redefinition session to reduce the completion time to redefine multiple partitions. Redefinition of tables that have Oracle Virtual Private Database policies defined on them to eliminate downtime for redefining these tables. Drop index online (create/rebuild index online in release 10*g* and 11*g*) Alter index visible / invisible; Alter index unusable online Drop constraint online (create constraint online in release 11*g*) Set unused column online (add column online in release 11*g*) Online, multi-partition redefinition in single session Online redefinition of tables with Oracle Virtual Private Database policies Single command redefinition with new

Table 5-6 (Cont.) New Data Reorganization Capabilities by Release

Action	Oracle Database 9*i*	Oracle Database 10*g* Release 1	Oracle Database 10*g* Release 2	Oracle Database 11*g*	Oracle Database 12*c*
					`REDEF_TABLE` procedure Edition-based redefinition enhancements
Online table operations using SQL					Online move partition Online split partition Online table move

Reclaiming unused space	Not applicable	Use the `SHRINK SPACE` clause on the following statements:
		`ALTER TABLE`
		`ALTER INDEX`
		`ALTER MATERIALIZED VIEW`
		`ALTER MATERIALIZED VIEW LOG`
			Not applicable	Not applicable	Not applicable

Index create online	`CREATE INDEX emp.ename.idx ON emp(ename) ONLINE;` • Parallel operations supported • Partitions supported • All index types except cluster	Not applicable	Not applicable	DML lock-free online index creation, allowing transparent creation with no dependency on workload	Not applicable

Table 5-6 (Cont.) New Data Reorganization Capabilities by Release

Action	Oracle Database 9*i*	Oracle Database 10*g* Release 1	Oracle Database 10*g* Release 2	Oracle Database 11*g*	Oracle Database 12*c*
Index coalesce online	`ALTER INDEX emp.ename_idx COALESCE;` • Parallel operations supported • Partitions supported • All index types	Not applicable	Not applicable	Not applicable	Not applicable
Index-organized table move online	`ALTER TABLE emp MOVE ONLINE;` • Parallel operations not supported • Partitions supported • Index-Organized Table only	Not applicable	Not applicable	Not applicable	Not applicable

5.4 Oracle High Availability Solutions for System and Software Maintenance

Oracle provides high availability solutions to prevent, tolerate, and reduce downtime for all types of planned maintenance. Table 5-7 (page 5-18) and Table 5-8 (page 5-20) describe the various Oracle high availability solutions for planned downtime, along with the outage time that can be attained with each solution.

In all cases, Oracle recommends that you extensively test all procedures before conducting planned maintenance operations. See the tables in High Availability Architectures (page 7-1) for a summary of the attainable recovery times for all types of planned downtime for each Oracle high availability architecture.

Table 5-7 Oracle High Availability Solutions for System and Software Maintenance

Maintenance Type	Oracle Recommended Solution	Solution Description	Outage Time
Operating system and hardware upgrades	Oracle Real Application Clusters and Oracle Clusterware (page 3-20), Oracle RAC One Node (page 3-24), or Data Guard Standby-First Patch Apply	Operating System Upgrades and Hardware Upgrades (page 5-21)	No downtime for Oracle RAC and Oracle RAC One Node. Seconds to minutes for Standby-First Patch Apply

Table 5-7 (Cont.) Oracle High Availability Solutions for System and Software Maintenance

Maintenance Type	Oracle Recommended Solution	Solution Description	Outage Time
Oracle interim patches or diagnostic patches	Oracle Real Application Clusters and Oracle Clusterware (page 3-20), Oracle RAC One Node (page 3-24), or Online Patching (page 5-22)	Online Patching (page 5-22)	No downtime Patches that cannot be applied by performing a rolling upgrade can be applied with the `MINIMIZE_DOWNTIME` option of the OPatch utility to reduce the availability impact of the patch application.
Oracle Database and Oracle Grid Infrastructure bundle patches, Patch Set Updates (PSU), Critical Patch Updates (CPU)	Data Guard Standby-First Patch Apply, Oracle Real Application Clusters and Oracle Clusterware (page 3-20), or Oracle RAC One Node (page 3-24)	System and Cluster Upgrades Using Data Guard (page 5-22)	Seconds to minutes with Standby-First Patch Apply No downtime for Oracle RAC and Oracle RAC One Node
Oracle Database and Oracle Grid Infrastructure Patch Set (for example, Oracle Database 12.1.0.1 to 12.1.0.2) and Major Upgrade (for example, Oracle Database 12.1 to 12.2)	Oracle Data Guard (page 3-2)	Database Rolling Upgrade with Data Guard (page 5-26)	Seconds to minutes
Upgrading Exadata storage	The Exadata PatchMgr utility	Rolling Upgrade of Exadata Storage Server Software (page 5-26)	No downtime
Application upgrades	Online Application Maintenance and Upgrades (page 5-32)	Online Application Maintenance and Upgrades (page 5-32)	No downtime

Table 5-8 Planned Maintenance Matrix for MAA Reference Architectures and Multitenant Architectures

Event	Solutions for Bronze, Silver, Gold, and Platinum Service Level Tiers	Expected Downtime
Migrations	See Also: MAA White Paper "High Availability Best Practices for Database Consolidation: The Foundation for Database-as-a-Service," section "Migration to Multitenant Architecture" at http://www.oracle.com/technetwork/database/availability/maa-consolidation-2186395.pdf	Varies
Dynamic and Online Resource Provisioning or Online reorganization and redefinition	ALL Tiers: Online Reorganization and Redefinition of select objects within each PDB See also: Dynamic and Online Resource Provisioning (page 5-11) and Online Reorganization and Redefinition (page 5-14)	Zero
Online Patches	ALL Tiers: Entire CDB can be online patched if relevant	Zero
Database and Grid Infrastructure Patches and One-off Patches	ALL Tiers: PDB can unplug and plug into a separate CDB with targeted software release SILVER: Entire CDB can leverage Oracle RAC One Node rolling upgrade if relevant GOLD/PLATINUM: Entire CDB can leverage Oracle RAC rolling upgrade if relevant. Application continuity will complement in the PLATINUM tier. GOLD: Entire CDB can leverage Data Guard standby-first patching and issue Data Guard switchover PLATINUM: Entire CDB can leverage Data Guard standby-first patching and issue Data Guard switchover and application continuity	Estimated seconds to hour with no datafile copy option Zero by relocating services Zero by relocating services Zero application outage Seconds to minutes Zero application outage
Database Patchsets	ALL Tiers: PDB can unplug and plug into a separate CDB with targeted software release GOLD/PLATINUM: Entire CDB can leverage Data Guard database rolling upgrade for patchsets and major database releases PLATINUM: CDB or PDB can fail over to secondary GoldenGate replica residing on the targeted software version	Estimated seconds to hour with no datafile copy option Seconds to Minutes Zero downtime
Application upgrades	PLATINUM: Edition-Based Redefinition requires developers to design to leverage this feature PLATINUM: PDB can switch over to GoldenGate replica with the targeted application changes See also: Online Application Maintenance and Upgrades (page 5-32)	Zero Zero to near zero downtime with moving services

See Also:

Oracle Data Guard Concepts and Administration for more information about using Data Guard with SQL Apply to upgrade an Oracle database

Rolling upgrade best practices white papers at `http://www.oracle.com/goto/maa`

5.4.1 Operating System Upgrades and Hardware Upgrades

Using Oracle RAC is the recommended solution for avoiding downtime during system and hardware upgrades. For a single-instance Oracle RAC database, you can use Oracle RAC One Node (page 3-24).

If you cannot perform the upgrade using Oracle RAC or Oracle RAC One Node, then the recommended solution is to use Data Guard and physical standby databases as described in System and Cluster Upgrades Using Data Guard (page 5-22). Alternatively, you can use cold cluster failover with Oracle Clusterware as described in Rolling Upgrade with Oracle Clusterware (page 5-25).

The following list provides a high-level overview of the steps when upgrading using Oracle RAC:

1. Perform the following prerequisite checks:
   - Ensure that the planned maintenance can be performed in a rolling manner from an operating system perspective.
   - Ensure that the database and clusterware versions are certified with the new system and hardware changes.
2. Stop the application service if the application service runs on more than one instance in the cluster. If the application service runs on only the instance being upgraded, then relocate the service to another node in the cluster.

   Stopping the application service implicitly redirects connections off of the destination instance when using fast application notification (FAN).
3. Shut down the destination instance or instances with the `IMMEDIATE` option.
4. Shut down and disable Oracle Clusterware.

   Disabling Oracle Clusterware prevents it from starting automatically.
5. Perform maintenance.
6. Enable and start Oracle Clusterware.

   This step implicitly starts the database instances.
7. Start the application service.

   This step implicitly redirects connections to the destination instance when using FAN.
8. Repeat all steps on the next node.

See your operating system-specific Oracle Real Application Clusters installation guide.

5.4.2 Online Patching

Typically, interim and diagnostic patches are applied to one node at a time in a rolling manner. During patch application to a software home, the software (for example, a database instance) running from the home is shut down. If, however, there is an urgent need for the patch to be installed and software cannot be shut down at the current time, then qualified interim and diagnostic patches can be applied online while software remains running.

The only time a patch should be applied in an online manner is when:

- The patch README indicates that it can be applied in an online manner.
- The patch needs to be applied urgently and database instances cannot be shut down to apply the regular (offline) version of the patch.

You can perform online patching with any Oracle database using the OPatch command-line utility.

Use the following considerations when performing online patching:

- Oracle provides qualified interim and diagnostic patches as combination patches, which contain both an online patch and an offline patch for the same bug fix.

  Thus, you can apply the online patch initially to avoid unplanned downtime. However, because online patches have memory overhead, you should roll back the online patch, and apply the offline patch during scheduled downtime.

- Applying an online patch increases memory consumption on the system because each Oracle process uses more memory from the Program Global Area (PGA) during the patch application. Consider memory requirements before you begin applying an online patch. Each online patch is unique, and the memory requirements are patch-specific. Apply the patch on your test system first so that you can assess the effect of the online patch on your production system and estimate any additional memory usage.

See Also:

"RDBMS Online Patching Aka Hot Patching" in My Oracle Support Note 761111.1 at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=761111.1`

Oracle Database Upgrade Guide for an overview of rolling upgrades and rolling patches

Oracle OPatch User's Guide for information about online patching and the OPatch utility

---

5.4.3 System and Cluster Upgrades Using Data Guard

Data Guard and physical standby databases are the recommended solution for performing system and cluster upgrades (including Oracle Grid Infrastructure upgrades) that you cannot upgrade using Oracle RAC rolling upgrades.

Data Guard is also recommended for migrations to Oracle ASM, Oracle RAC, 64-bit systems, Windows to Linux, or Linux to Windows, or the same processor architecture platforms. For example:

- Use Data Guard for system upgrades that cannot be upgraded using Oracle RAC rolling upgrades due to system restrictions.
- Use Data Guard when migrating to Oracle ASM, from a noncluster environment to Oracle RAC, to a different platform with the same endian format, or to a different platform with the same processor architecture. The time required to perform the switchover is the only downtime incurred. For more information, see "Data Guard Support for Heterogeneous Primary and Physical Standbys in Same Data Guard Configuration" in My Oracle Support Note 413484.1 at

  `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=413484.1`

In general, first upgrade the system or cluster where the physical standby database runs, and then perform a Data Guard switchover to the physical standby database. For cases where database software is upgraded, refer to Rolling Patch Installation with Data Guard (page 5-25).

5.4.3.1 Upgrading the Physical Standby Database

To upgrade the physical standby database and perform a switchover:

1. Upgrade the system or change the physical standby database system to your destination environment.

   For example, you can convert the standby database from a single-instance database to an Oracle RAC database by using Oracle ASM, without any effect on the primary database. Then, restart the standby database, ensure that it matches your destination environment, and wait for Redo Apply to finish applying all redo data to the standby database.

2. Perform a Data Guard switchover. Optimally, the switchover should take only seconds to minutes.
3. Shut down the original primary database (now the standby database).
4. Upgrade or make system changes to the original primary database.
5. Restart the upgraded database as a standby database and allow recovery to automatically synchronize the databases.
6. Optionally, perform a Data Guard switchover to return the standby database to the primary database role.

Note:

Conversion from 32-bit to 64-bit is automatic if you are applying an Oracle Database patch set or doing an Oracle Database upgrade at the same time. If you are upgrading only the operating system, then you may need to perform the additional post-upgrade steps that are described in the My Oracle Support Note 414043.1 at `http://support.oracle.com/`.

5.4.3.2 Best Practices for System and Cluster Upgrades

Consider the following best practices and guidelines for system and cluster upgrades and migrations:

- For fastest switchover, configure the standby database to use real-time apply and, if possible, ensure there are no archive log gaps and that the databases are close to being synchronized before beginning the switchover operation.
- Use Data Guard and physical standby databases to perform system and cluster upgrades if Oracle RAC rolling upgrade or online patching is not possible.

See Also:

Oracle Data Guard Concepts and Administration for more information about using Data Guard to perform system and cluster upgrades.

5.4.4 Patching and Rolling Upgrades With Oracle Real Application Clusters

Oracle patches to database software are usually applied to implement known fixes for software problems, or to apply diagnostic patches to gather information about a problem. Plan to perform upgrades and apply patches during a scheduled maintenance window. Use the options for rolling or non-rolling patch apply that work best with your business needs.

There are several types of patches including:

- **Interim patch**

  An interim patch is a bug fix made available to customers who cannot wait until the fix is included in a subsequent patch set release or database patch for Exadata. Installation of an interim patch is done on an as-needed basis, hence it is not a regularly scheduled planned maintenance event.

- **Bundle patch**

  A collection of patches that is issued between patch sets. A patch bundle is usually cumulative. Microsoft Windows bug fixes for Oracle Database are generally issued in a patch bundle (as opposed to an interim patch).

- **Patch Set**

  Patch sets contain primarily bug fixes; however, some minor new features and change in functionality may be included.

- **Patch Set Update (PSU)**

  A quarterly patch that contains the most critical fixes for the applicable product (including security fixes), enabling customers to apply one patch to avoid many problems.

- **Critical Patch Update (CPU)**

  A collection of high-priority fixes (usually for security issues) once a quarter. CPUs are cumulative with respect to prior security fixes but may contain other fixes in order to address patch conflicts with non-security patches (that is, reduce the need for merge requests).

- **Diagnostic patch**

  A patch created specifically to diagnose a problem and not to fix a bug.

5.4.4.1 Rolling Patch Installation with Oracle Real Application Clusters

To avoid downtime when applying Oracle database patches, perform rolling patch upgrades using Oracle RAC. You can apply approximately 90% of the patches using Oracle RAC. Oracle provides the capability to perform rolling patch upgrades with Oracle RAC with little or no database downtime using the OPatch command-line utility. If it is not possible to use Oracle RAC, then use Data Guard and physical standby databases.

An Oracle RAC rolling upgrade enables all but one of the instances of the Oracle RAC database to be available during the scheduled outage, further reducing the impact on the application downtime required for planned maintenance. The Oracle OPatch utility enables you to apply the patch successively to the different instances in an Oracle RAC database.

Performing a rolling upgrade is possible only for patches that are certified for rolling upgrades, which is indicated in the README.

See Also:

My Oracle Support note 1593712.1 at `http://support.oracle.com` for the steps required to gracefully apply rolling patches without application interruption.

System and Cluster Upgrades Using Data Guard (page 5-22) for information about using Data Guard for rolling patch upgrades

5.4.4.2 Rolling Patch Installation with Data Guard

If it is not possible to use Oracle RAC to apply updates in a rolling manner, then use Data Guard and physical standby databases. Data Guard Standby-First Patch Apply provides support for different patch set updates (PSUs), bundle patches, or interim patches between a primary database and its physical standby database for the purpose of applying and validating Oracle patches in rolling manner.

Check the README for the patch to determine if a target patch is certified as being a Data Guard Standby-First Installable.

See Also:

- My Oracle Support Note 1265700.1 at `http://support.oracle.com` for additional information about Oracle Data Guard Standby First Patch Apply.

5.4.5 Rolling Upgrade with Oracle Clusterware

Performing rolling upgrades of Oracle Clusterware is the recommended solution for avoiding downtime when upgrading Oracle Clusterware. For single-instance Oracle RAC databases, consider using Oracle RAC One Node.

Rolling upgrades avoid downtime and ensure continuous availability of Oracle Clusterware while the software is upgraded to the new version. When you upgrade to Oracle Clusterware 12*c*, Oracle Clusterware and Oracle ASM binaries are installed as a single binary called the grid infrastructure. You can upgrade Oracle Clusterware in a rolling manner from Oracle Clusterware 10*g* and Oracle Clusterware 11*g*.

You can perform all upgrades to Oracle Clusterware in a rolling manner.

See Also:

Your operating system-specific Oracle Clusterware or Oracle Real Application Clusters installation guide at https://docs.oracle.com/database

5.4.6 Rolling Upgrade with Oracle Automatic Storage Management

Performing rolling upgrades is the recommended solution for upgrading Oracle ASM. You can perform all upgrades starting with Oracle Database 11*g* (and later releases) in a rolling manner.

When you upgrade to Oracle Clusterware 12*c*, Oracle Clusterware and Oracle ASM binaries are installed as a single binary called the grid infrastructure. You can only upgrade Oracle ASM in a rolling manner from Oracle Database 11*g* release 1 (11.1).

See Also:

Oracle Automatic Storage Management Administrator's Guide.

5.4.7 Rolling Upgrade of Exadata Storage Server Software

During a rolling Exadata Storage Server Software upgrade, storage servers are patched one at a time until all of the servers are updated. Rolling patching takes advantage of Oracle ASM redundancy and automatic disk resynchronization to allow databases to continue to operate during patching. Rolling Exadata Storage Server Software upgrade orchestration is managed by the PatchMgr utility provided with the Exadata Storage Server Software.

See Also:

- My Oracle Support Note 888828.1 at `http://support.oracle.com/` that includes:
- The Oracle Exadata Storage Server website at `http://www.oracle.com/exadata`

5.4.8 Database Rolling Upgrade with Data Guard

Data Guard using SQL Apply is a recommended solution for performing patch set and database upgrades with minimal downtime. If the source database is using data types not natively supported by SQL Apply, you can use Extended Datatype Support (EDS) to accommodate several more advanced data types.

If the source database is using a software version not supported by SQL Apply rolling upgrade (earlier than Oracle Database release 10.1.0.3), or using EDS cannot sufficiently resolve SQL Apply data type conflicts, then consider using Rolling Upgrades using Oracle Active Data Guard, Database Upgrade Assistant (DBUA), transportable tablespace, or Oracle GoldenGate.

- Rolling Upgrades using Oracle Active Data Guard use a Data Guard physical standby database and the SQL Apply process.

- DBUA provides a graphical user interface (GUI) utility that guides you through the upgrade process and is the simplest and recommended method of upgrading a database. However, if the time it takes DBUA to upgrade a database does not fit in the defined maintenance window, then consider using the transportable tablespace feature to perform a database upgrade in less than 1 hour.
- Transportable tablespace is the solution if you cannot use SQL Apply but the maintenance window requires downtime to be less than 1 hour in duration, and the database being upgraded has a small number of simple schemas and data files that do not need to be transferred as part of the transport process (such as when the data files will be used in place).
- Oracle GoldenGate provides the most flexibility when performing database upgrades and requiring additional data type support.

DBUA incurs downtime. The amount of downtime is dependent on a number of factors.

Do not use Oracle RAC to perform rolling upgrades of patch sets. See your operating system-specific Oracle Real Application Clusters installation guide.

Oracle Data Guard broker supports Oracle Active Data Guard rolling upgrade in Oracle Database 12*c* Release 2. Oracle Active Data Guard rolling upgrade was introduced in Oracle Database 12*c* Release 1. It simplifies the execution of the transient logical database rolling upgrade process by automating many manual steps in a simple PL/SQL package, `DBMS_ROLLING`. In addition to making database rolling upgrades simpler, the automated process is much more reliable. Oracle Data Guard broker can now direct Oracle Active Data Guard rolling upgrades from the DGMGRL command-line interface. Data Guard Broker support also adds substantial simplification to the rolling upgrade process by transparently handling redo transport destination settings and other tasks.

See Also:

Performing Database Upgrades Using Data Guard and Physical Standby Databases (page 5-27) describes Rolling Upgrades using Oracle Active Data Guard

Performing Database Upgrades Using Transportable Tablespace (page 5-28) describes the transportable tablespace solution

Performing Database Upgrades Using Oracle GoldenGate (page 5-29)

Oracle Database High Availability Best Practices for help choosing the database upgrade method appropriate for your configuration, and for additional considerations when choosing DBUA as an upgrade option

Oracle Database Upgrade Guide for information about using DBUA to upgrade Oracle Database software

5.4.8.1 Performing Database Upgrades Using Data Guard and Physical Standby Databases

Rolling Upgrades using Oracle Active Data Guard provides new PL/SQL packages that automate much of the process of performing a database rolling upgrade (to a later Oracle Database release or to a new patch set, or when performing other database maintenance) using a physical standby database. You input an upgrade plan and

PL/SQL packages automate three phases of the upgrade according to that plan: start, switchover, and finish.

During the upgrade, SQL Apply is used to synchronize the standby across versions, however, when the upgrade is complete, the Data Guard configuration is returned its original state of a primary database and a physical standby database.

Data protection is maintained during the Data Guard database rolling upgrade process by enabling the standby database that is the target of the upgrade to continue receiving primary database redo while the standby database is open in upgrade mode.

If errors are encountered during the process, then you can choose to either correct the errors and resume the upgrade or fall back to the original state of the configuration. This is supported for database rolling upgrades from Oracle Database 12*c* release 1 (12.1) onward.

The Oracle Database 12*c* release includes additional native redo-based replication for Data Guard SQL Apply to support database rolling upgrades (**transient logical standby**). Supported data types include Oracle Securefile, XML, Database File System (DBFS), XDB, Oracle Spatial, Oracle Text and Oracle Multimedia.

Data Guard broker also supports database rolling upgrades.

Starting with Oracle Database 12*c*, Oracle Enterprise Manager Cloud Control (Cloud Control) provides options to perform a rolling upgrade of databases in a Data Guard configuration. The procedures are described in online help within Cloud Control.

See Also:

Appendix C in *Oracle Data Guard Concepts and Administration* for a full list of supported data types

---

5.4.8.2 Performing Database Upgrades Using Data Pump Full Transportable Export/Import

You can use full transportable export/import to upgrade a database from release 11.2.0.3 or later to Oracle Database 12*c*. To do so, install Oracle Database 12*c* and create an empty database. Next, use full transportable export/import to transport the release 11.2.0.3 database into the Oracle Database 12*c* database.

See Also:

Data Pump Full Transportable Export/Import (page 5-4) for the high-level steps.

Oracle Database Administrator's Guide for information about the general limitations of transporting data and limitations specific to full transportable export/import

---

5.4.8.3 Performing Database Upgrades Using Transportable Tablespace

If you cannot use SQL Apply because of data type conflicts, and testing shows that upgrading with DBUA cannot meet uptime requirements, then consider using the transportable tablespace solution to upgrade your database.

To use the transportable tablespace feature to upgrade an Oracle database:

1. Install Oracle Database software on the destination system and perform the initial steps on the source database to prepare for the transport process.
2. Prepare the source and destination databases:
   a. Gather information from the source database.
   b. Create the destination database with Database Configuration Assistant (DBCA).
   c. Prepare the destination database for Oracle Data Pump usage and to accept the tablespaces being transported.
3. Transport the user tablespaces:
   a. Ready the source database for transport by disconnecting users and restricting access to the source database, making all user tablespaces `READ ONLY`, and capturing sequence starting values from the source database.
   b. Transport the user tablespaces.
4. Verify that the destination database is complete and functional, and then back up the destination database.

Consider the following information when using the transportable tablespace feature:

- The transportable tablespace feature is an option for performing a database upgrade in less than 1 hour for databases that have simple schemas and where the data files do not need to be transferred as part of the transport process (such as when the data files will be used in place). See the MAA white paper "Database Upgrade Using Transportable Tablespace" available on the MAA web site at

  `http://www.oracle.com/goto/maa`

- Using the transportable tablespace feature reduces database upgrade time by moving all user tablespaces from a database running an earlier software release to an empty destination database running a current software release. With transportable tablespace, tablespace data files are plugged in to the database by copying the data files to the destination database, then importing the object metadata into the destination database.

5.4.8.4 Performing Database Upgrades Using Oracle GoldenGate

Use Oracle GoldenGate to reduce database upgrade downtime. Database upgrade downtime is reduced by allowing the target database to be upgraded to the new version and kept synchronized while the source database remains online running the current version. When you use Oracle GoldenGate the downtime required is the length of time it takes to reconnect the application to the target database.

The high-level steps are:

1. Start a change-synchronization Extract group to extract ongoing data changes.
2. Create a duplicate target database. The ideal duplicate target database will begin as a physical standby database that is up-to-date.
3. Activate and upgrade the target database to the target version (or perform your maintenance action as described in Table 7–6).
4. Start the change-synchronization Replicat group to resynchronize rows that were changed while the target database was being created and upgraded.

5. Stop the application.
6. Start the application, connecting to the target database.

See Also:

Oracle GoldenGate documentation for complete information about performing an online database upgrade at `http://www.oracle.com/technetwork/middleware/goldengate/overview/index.html`

"Zero-Downtime Database Upgrades Using Oracle GoldenGate" at `http://www.oracle.com/technetwork/middleware/goldengate/overview/ggzerodowntimedatabaseupgrades-174928.pdf`

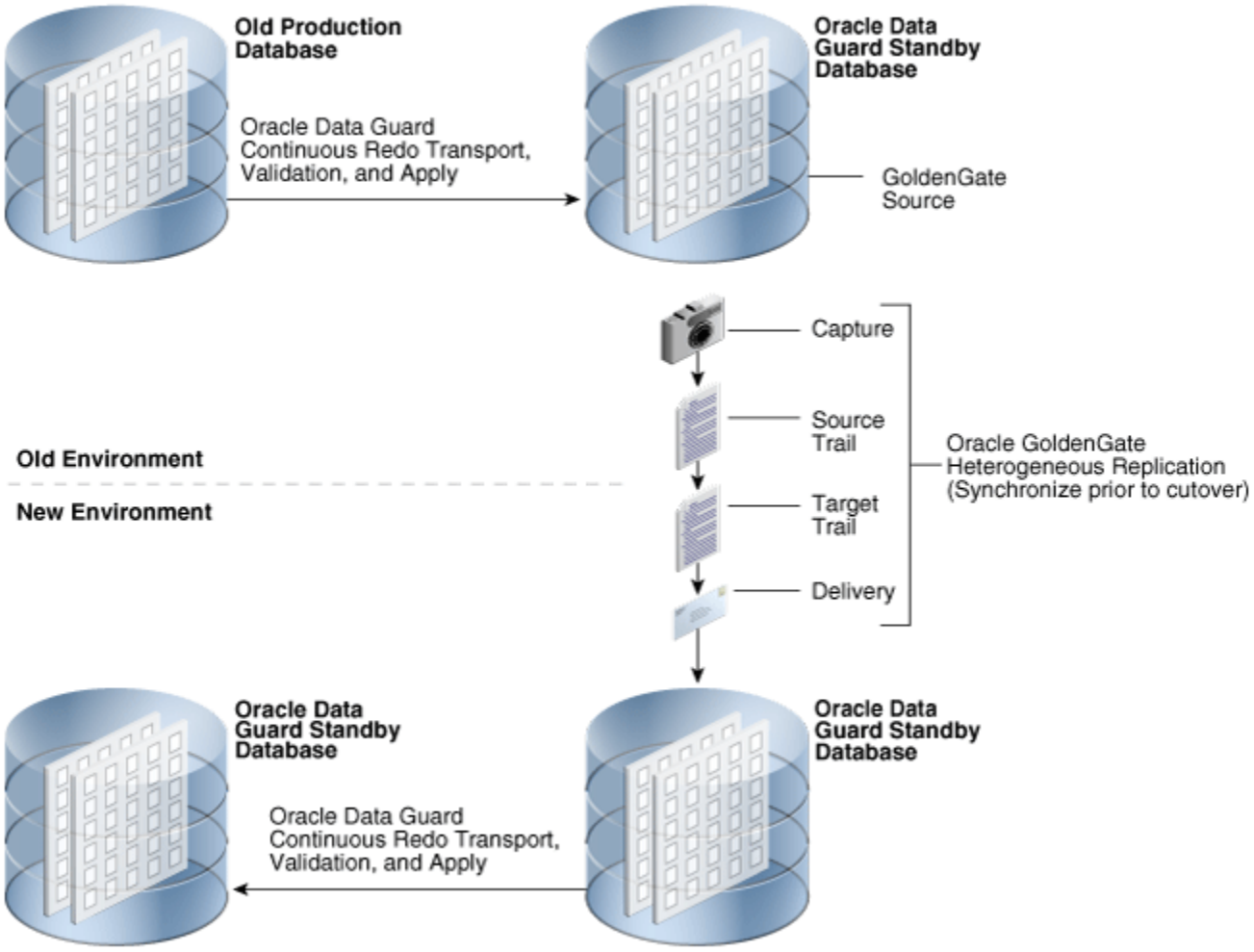
Oracle Database Backup and Recovery User's Guide to learn about duplicating a database

---

5.4.8.5 Performing Database Upgrades Using Oracle GoldenGate and Data Guard

The configuration in Figure 5-1 (page 5-31) shows how to configure Oracle GoldenGate and Data Guard to minimize downtime and risk for planned outages, such as for any upgrades and migrations that are not supported by a Data Guard database rolling upgrade. For example, this might include migrating to a different hardware architecture and operating system, or performing application upgrades that modify database objects. In this configuration, the physical standby databases provide disaster protection to prevent downtime or data loss before, during, and after the migration. This configuration also avoids any performance impact or operational risk by isolating the production database from any work required to perform the migration.

Figure 5-1 Oracle GoldenGate Configuration for Minimizing Planned Downtime



Oracle GoldenGate replication from the standby database (in the top right of Figure 5-1 (page 5-31)), to the new production database (bottom right), requires Oracle GoldenGate Archive Log Mode. If the requirements for Archive Log Mode cannot be met, then replicate directly from the original production database (represented by the database in the top left corner).

It is possible to Extract from the standby (top left in Figure 5-1 (page 5-31)) but Classic Capture is required along with ALO mode. See Oracle GoldenGate documentation for more information about ALO against a standby database.

`http://docs.oracle.com/goldengate/1212/gg-winux/GIORA/classic_capture.htm#A1740233`

These requirements are achieved by creating a parallel environment on the new platform. Depending upon the type of migration planned, instantiating the new primary database may be as simple as restoring a backup of the existing standby database. For more complex migrations it may be necessary to use other Oracle technologies to instantiate the new primary database, such as Oracle Transportable Technologies or Oracle Data Pump. After instantiated, any additional changes are then implemented on what will become the new production system. When all of the changes are implemented, a new physical standby database is created to provide continuous data protection after cutover. Oracle GoldenGate heterogeneous replication (previously configured), is then used to synchronize the new production

system with all transactions that occurred on the old system while the new environment was being implemented. When synchronization is complete, production is ready for cutover to the new environment. There is also the option of using Oracle GoldenGate heterogeneous replication after the cutover to keep the old environment synchronized with the new production system for a period of time, to provide a fast fall back option if any unanticipated problems arise.

5.5 Online Application Maintenance and Upgrades

For application changes, use the features described in the following list that can significantly reduce (or eliminate) the application downtime required to make changes to an application's database objects:

5.5.1 Edition-Based Redefinition

Edition-based redefinition (EBR) lets you upgrade the database component of an application while it is in use, thereby minimizing or eliminating downtime.

To upgrade an application while it is in use, you copy the database objects that comprise the application and redefine the copied objects in isolation. Your changes do not affect users of the application—they continue to run the unchanged application. When you are sure that your changes are correct, you make the upgraded application available to all users.

The following sections describe the Editions (page 5-32), Editioning Views (page 5-33), and Crossedition Triggers (page 5-33) features of edition-based redefinition.

For more information, see *Oracle Database Development Guide*.

5.5.1.1 Editions

Editions are nonschema objects; as such, they do not have owners. Editions are created in a single namespace, and multiple editions can coexist in the database. The edition feature enables you to copy database objects and redefine the copied objects in isolation.

The database must have at least one edition. Every newly created or upgraded Oracle Database starts with one edition named ora$base.

Editions provide a privacy mechanism for installing new code and for making data changes so that the running production application does not see the changes. When all the required changes are made in private, they are published in a single operation. Cutover depends simply on which edition a session uses.

5.5.1.2 Editioning Views

If you change the structure of one or more tables, you must also use the editioning view feature to insulate application code from changes made to the underlying table during online application upgrade. Tables are not editionable.

Columns are added to the underlying table and a new editioning view is created in the postupgrade edition to expose and to populate them.

Triggers may be created on an editioning view and its columns may be used in SQL hints. The defining subquery of an editioning view may only project or define aliases for selected columns. The `SELECT` list is used to project a subset of the table's columns and, typically, to rename them. It, therefore, defines a mapping of physical columns to logical columns.

5.5.1.3 Crossedition Triggers

Crossedition triggers are used as part of edition-based redefinition to keep the data in the preupgrade and postupgrade editions in step with each other. The preupgrade application remains in use concurrently while changes are applied, redefining the preupgrade edition to a postupgrade edition.

If users must be able to change data in the tables while you are changing the table structure, you use *forward* crossedition triggers. If you make the upgraded application available to some users while others continue to use the older version of the application, you also use *reverse* crossedition triggers. Crossedition triggers are not a permanent part of the application because you drop or disable them after you have made the upgraded application available to all users.

5.5.2 Oracle GoldenGate for Rolling Upgrades

Consider using Oracle GoldenGate for fast rolling upgrades. However, although Oracle GoldenGate upgrades might incur little or no database downtime, your ability to configure this solution requires some operational investment.

See Also:

Oracle GoldenGate (page 3-12) and the Oracle GoldenGate documentation

5.5.3 DDL with the WAIT Option

Data definition language (DDL) commands require exclusive locks on internal structures. If DDL commands are issued, then these locks may not be available causing the statement to immediately fail even though the DDL might have succeeded less than a second later. Specifying DDL commands with the `WAIT` option (the new default) resolves this issue. You specify the wait time instance-wide (in the initialization parameter file) and modify the wait time on a session level.

Specifying DDL commands with the `WAIT` option provides more flexibility to define grace periods for such commands to succeed instead of raising an error right away, thus requiring additional application logic to handle such errors.

5.5.4 ENABLE, DISABLE, and FOLLOWS Clauses for CREATE TRIGGER

The states (`ENABLE` and `DISABLE`) and ordering (`FOLLOWS`) are triggers to control the firing of triggers. These additional states allow greater administrative control for triggers. You can use the `CREATE TRIGGER` statement in a disabled state to validate successful compilation before enabling. In addition, the trigger order can be controlled with the `FOLLOWS` clause.

5.5.5 Enhanced ADD COLUMN Functionality

Default values of columns are maintained in the data dictionary for columns specified as `NOT NULL`.

Adding new columns with `DEFAULT` values and the `NOT NULL` constraint no longer requires the default value to be stored in all existing records. This enhancement not only enables a schema modification in less than a second and works independently of the existing data volume, but it also consumes no space.

5.5.6 Finer-Grained Dependencies

Prior to Oracle Database 11*g*, metadata recorded mutual dependencies between objects with the granularity of the whole object. (For example, PL/SQL unit P depends on PL/SQL unit Q, or view V depends on table T.) In such cases, the dependent objects were sometimes needlessly invalidated. For example, if view V depends only on columns C1, C2, and C3 in table T and a new column, C99, is added, the validity of view V is not logically affected. Nevertheless, in earlier releases, V was invalidated by the addition of column C99.

Beginning with Oracle Database 11*g* release 1 (11.1), dependency metadata is recorded at a finer level of granularity, so that the addition of C99 does not invalidate view V. Similarly, if procedure P depends only on elements E1 and E2 in package PKG, then if element E99 is added to PKG, procedure P is not invalidated. (In Oracle Database 10*g*, this change to PKG would invalidate procedure P.)

By reducing the consequential invalidation of dependent objects in response to changes in the objects they depend upon, you can increase application availability. The benefit occurs both in the development environment and when an active application is parsed or upgraded. The benefit occurs when an Oracle Database patch set is applied because changes to schema objects must be compatible.

5.5.7 Invisible Indexes

An invisible index provides an alternative to making an index unusable or even to dropping the index. An invisible index is maintained for any DML operation but is not used by the optimizer unless you explicitly specify the index with a hint.

Applications often require modification even when the complete application cannot be taken offline. Invisible indexes enable you to use temporary index structures for certain operations or modules of an application without affecting the overall application. Furthermore, you can use invisible indexes to test the removal of an index without dropping it right away, thus enabling a grace period for testing in production environments.

5.5.8 Invisible Columns

An invisible column is a user-specified column whose values are only visible when the column is explicitly specified by name. You can add an invisible column to a table without affecting existing applications, and make the column visible if necessary.

You might use invisible columns if you want to make changes to a table without disrupting applications that use the table. After you add an invisible column to a table, queries and other operations that must access the invisible column must refer to the column explicitly by name. When you migrate the application to account for the invisible columns, you can make the invisible columns visible.

5.5.9 Multiple Indexes on the Same Set of Columns

In Oracle Database 12*c*, both B-tree and bitmap indexes can be created on the same set of columns. This feature enables an index to be created on the same set of columns as an existing index as long as some characteristic is different. This enables the type of an index to be changed in a patch edition while not disrupting an application. Only one of the multiple indexes can be a visible index at any time.

See Also:

"Creating Multiple Indexes on the Same Set of Columns" in *Oracle Database Administrator's Guide*

5.5.10 Dependent PL/SQL Recompilation After Online Table Redefinition

This feature minimizes the need to recompile dependent PL/SQL packages after an online table redefinition. If the redefinition does not logically affect the PL/SQL packages, recompilation is not needed. This optimization is turned on by default.

If recompilation is needed, this feature reduces the time and effort to manually recompile a dependent PL/SQL package after an online table redefinition. The recompilation also includes views, synonyms, and other table-dependent objects (with the exception of triggers) that are not logically affected by the redefinition.

See Also:

Oracle Database Administrator's Guide for more information about redefining tables online

6

Operational Prerequisites to Maximizing Availability

Use operational best practices to provide a successful MAA implementation.

This chapter contains the following topics:

6.1 Understand Availability and Performance SLAs

Understand and document your high availability and performance service-level agreements (SLAs):

- Understand the attributes of High Availability and various causes of downtime as described in Overview of High Availability (page 1-1).
- Get agreement from line of business, upper management, and technical teams on HA and performance service level agreements as described in High Availability Requirements (page 2-1), and A Methodology for Documenting High Availability Requirements (page 2-2).

6.2 Implement and Validate a High Availability Architecture That Meets Your SLAs

Once you have agreement on your high availability and performance service level requirements, map requirements to one of Oracle's standard and validated architectures as described in High Availability and Data Protection – Getting From

Requirements to Architecture (page 2-1). Evaluate Outage and Planned Maintenance matrices relevant for your MAA referenced architecture similar to those found in Oracle Database High Availability Solutions for Unplanned Downtime (page 4-1), and Oracle Database High Availability Solutions for Planned Downtime (page 5-1). For more details about your chosen MAA reference architecture, refer to High Availability Architectures (page 7-1).

See Also:

High Availability Architectures (page 7-1)

MAA white paper "High Availability Best Practices for Database Consolidation: The Foundation for Database-as-a-Service" at `http://www.oracle.com/technetwork/database/availability/maa-consolidation-2186395.pdf`

6.3 Establish Test Practices and Environment

Validate and automate repair operations to ensure that you meet your target high availability service-level agreements (SLAs). You should validate the backup, restore, and recovery operations and periodically evaluate all repair operations for various types of possible outages.

If you use Data Guard for disaster recovery and data protection, Oracle recommends that you perform periodic switchover operations or conduct full application and database failover tests. Also, periodically execute Application and Data Guard switchovers to fully validate all role transition procedures.

A good test environment and proper test practices are essential prerequisites in achieving the highest stability and availability in your production environment. By validating every change in your test environment thoroughly, you can proactively detect, prevent and avoid problems before applying the same change on production.

These practices involve the following:

- Configuring the Test System and QA Environments (page 6-2)
- Performing Preproduction Validation Steps (page 6-4)

See Also:

Table 4-1 (page 4-1)

Table 4-2 (page 4-6)

Table 5-7 (page 5-18)

Table 5-8 (page 5-20)

6.3.1 Configuring the Test System and QA Environments

The test system should be a replica of the production MAA environment (for example, using the MAA Gold tier). There will be trade offs if the test system is not identical to the MAA service-level driven standard reference architecture that you chose. It's recommended to execute functional, performance and availability tests with a workload that mimics production. Evaluate if availability and performance SLAs are

maintained after each change and ensure that clear fallback or repair procedures are in place if things go awry while applying the change on the production environment.

With a properly configured test system, many problems can be avoided because changes are validated with an equivalent production and standby database configuration containing a full data set and using a workload framework to mimic production (for example, using Oracle Real Application Testing).

Do not try to reduce costs by eliminating the test system because that decision ultimately affects the stability and the availability of your production applications. Using only a subset of system resources for testing and QA has the tradeoffs shown in Table 6-1, which is an example of the MAA Gold tier.

Table 6-1 Tradeoffs for Different Test and QA Environments

Test Environment	Benefits and Tradeoffs
Full Replica of Production and Standby Systems	Validate all patches and software changes. Validate all functional tests. Full performance validation at production scale. Full high availability validation.
Full Replica of Production Systems	Validate all patches and software changes. Validate all functional tests. Full performance validation at production scale. Full high availability validation minus the standby system. No functional, performance, high availability and disaster recovery validation with standby database.
Standby System	Validate most patches and software changes. Validate all functional tests. Full performance validation if using Data Guard Snapshot Standby but this can extend recovery time if a failover is required. Role transition validation. Resource management and scheduling is required if standby and test databases exist on the same system.
Shared System Resource	Validate most patches and software changes. Validate all functional tests. This environment may be suitable for performance testing if enough system resources can be allocated to mimic production. Typically, however, the environment includes a subset of production system resources, compromising performance validation. Resource management and scheduling is required.
Smaller or Subset of the system resources	Validate all patches and software changes. Validate all functional tests. No performance testing at production scale. Limited full-scale high availability evaluations.

Table 6-1 (Cont.) Tradeoffs for Different Test and QA Environments

Test Environment	Benefits and Tradeoffs
Different hardware or platform system resources but same operating system	Validate most patches and software changes. Limited firmware patching test. Validate all functional tests unless limited by new hardware features. Limited production scale performance tests. Limited full-scale high availability evaluations.

See Also:

Understand Availability and Performance SLAs (page 6-1)

Oracle Database Testing Guide

6.3.2 Performing Preproduction Validation Steps

Pre-production validation and testing of hardware, software, database, application or any changes is an important way to maintain stability. The high-level pre-production validation steps are:

1. Review the patch or upgrade documentation or any document relevant to that change. Evaluate the possibility of performing a rolling upgrade if your SLAs require zero or minimal downtime. Evaluate any rolling upgrade opportunities to minimize or eliminate planned downtime. Evaluate whether the patch or the change qualifies for Standby-First Patching.

   Note:

   Standby-First Patch enables you to apply a patch initially to a physical standby database while the primary database remains at the previous software release (this applies to certain types of patches and does not apply to Oracle patch sets and major release upgrades; use the Data Guard transient logical standby method for patch sets and major releases). Once you are satisfied with the change, then perform a switchover to the standby database. The fallback is to switchback if required. Alternatively, you can proceed to the following step and apply the change to your production environment. For more information, see "Oracle Patch Assurance - Data Guard Standby-First Patch Apply" in My Oracle Support Note 1265700.1 at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1265700.1`

2. Validate the application in a test environment and ensure the change meets or exceeds your functionality, performance, and availability requirements. Automate the procedure and be sure to also document and test a fallback procedure. This requires comparing metrics captured before and after patch application on the test and against metrics captured on the production system. Real Application Testing may be used to capture the workload on the production system and replay it on the test system. AWR and SQL Performance Analyzer may be used to assess performance improvement or regression resulting from the patch.

Validate the new software on a test system that mimics your production environment, and ensure the change meets or exceeds your functionality, performance, and availability requirements. Automate the patch or upgrade procedure and ensure fallback. Being thorough during this step eliminates most critical issues during and after the patch or upgrade.

3. Use Oracle Real Application Testing and test data management features to comprehensively validate your application while also complying with any security restrictions your line of business may have. Oracle Real Application Testing (a separate database option) enables you to perform real-world testing of Oracle Database. By capturing production workloads and assessing the impact of system changes on these workloads before production deployment, Oracle Real Application Testing minimizes the risk of instabilities associated with system changes. SQL Performance Analyzer and Database Replay are key components of Oracle Real Application Testing. Depending on the nature and impact of the system change being tested, and on the type of system on which the test will be performed, you can use either or both components to perform your testing.

   When performing real-world testing there is a risk of exposing sensitive data to non-production users in a test environment. The test data management features of Oracle Database help to minimize this risk by enabling you to perform data masking and data subsetting on the test data.

4. If applicable, perform final pre-production validation of all changes on a Data Guard standby database before applying them to production. Apply the change in a Data Guard environment, if applicable.

5. Apply the change in your production environment.

See Also:

Data Guard Redo Apply and Standby-First Patching (page 3-9) and Data Guard Transient Logical Rolling Upgrades (page 3-9) for more information about Data Guard standby-first patch apply and transient logical standby method

Oracle Database Testing Guide

Oracle Data Guard Concepts and Administration for complete information about Converting a Physical Standby Database into a Snapshot Standby Database

Oracle Data Guard Concepts and Administration for more information about Performing a Rolling Upgrade With an Existing Physical Standby Database

Oracle GoldenGate For Windows and UNIX Administrator's Guide for more information about Oracle GoldenGate

The MAA white paper, "Oracle Database Rolling Upgrades: Using a Data Guard Physical Standby Database", from the MAA Best Practices area for Oracle Database at `http://www.oracle.com/goto/maa`

See "Oracle Patch Assurance - Data Guard Standby-First Patch Apply" in My Oracle Support Note 1265700.1 at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1265700.1`

6.4 Set Up and Use Security Best Practices

Corporate data can be at grave risk if placed on a system or database that does not have proper security measures in place. A well-defined security policy can help protect your systems from unwanted access and protect sensitive corporate information from sabotage. Proper data protection reduces the chance of outages due to security breaches.

See Also:

Oracle Database Security Guide.

6.5 Establish Change Control Procedures

Institute procedures that manage and control changes as a way to maintain the stability of the system and to ensure that no changes are incorporated in the primary database unless they have been rigorously evaluated on your test systems, or any one of the base architectures in the MAA service-level tiers.

Review the changes and get feedback and approval from your change management team, which should include representatives for any component that affects the business requirements, functionality, performance, and availability of your system. For example, the team can include representatives for end-users, applications, databases, networks, and systems.

6.6 Apply Recommended Patches and Software Periodically

By periodically testing and applying the latest recommended patches and software versions, you ensure that your system has the latest security and software fixes required to maintain stability and avoid many known issues. Remember to validate all updates and changes on a test system before performing the upgrade on the production system.

Furthermore, Oracle health check tools such as `orachk` (supporting Non-Engineered Systems and Oracle Database Appliance) and `exachk` (supporting Engineered Systems such as Oracle Exadata Database Machine, Exalogic, Zero Data Loss Recovery Appliance, and Big Data Appliance) provide Oracle software upgrade advice, critical software update recommendations, and patching and upgrading pre-checks, along with its system and database health checks and MAA recommendations.

See Also:

"Oracle Recommended Patches -- Oracle Database" in My Oracle Support Note 756671.1 at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=756671.1

"Exadata Database Machine and Exadata Storage Server Supported Versions" in My Oracle Support Note 888828.1 at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=888828.1

"ORAchk - Health Checks for the Oracle Stack" in My Oracle Support Note 1268927.2 at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1268927.2

"Oracle Exadata Database Machine exachk or HealthCheck" in My Oracle Support Note 1070954.1 at https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1070954.1

6.7 Execute Disaster Recovery Validation

Disaster recovery validation is required to ensure that you meet your disaster recovery service level requirements such as RTO and RPO.

Whether you have a standby database, Oracle GoldenGate replica, or leverage database backups from Zero Data Loss Recovery Appliance (Recovery Appliance), ZFS Storage, or another third party, it is important to ensure that the operations and database administration teams are well prepared to fail over or restore the database and application any time the primary database is down or underperforming, according to a predetermined threshold. By reacting and executing efficiently, which includes detection and making the decision to fail over or restore, overall down time can be reduced significantly.

If you use Data Guard or Oracle GoldenGate for high availability, disaster recovery, and data protection, Oracle recommends that you perform regular application and database switchover operations every three to six months, or conduct full application and database failover tests.

Periodic RMAN cross checks, RMAN backup validations, and complete database restore and recovery are required to validate your disaster recovery solution using your existing backup solution. With Recovery Appliance there are inherent backup checks and validations done automatically within the appliance, but periodic restore and recovery tests are still recommended.

See Also:

Oracle Database High Availability Best Practices for more information about configuring Oracle Data Guard and role transition best practices

Oracle Data Guard Concepts and Administration for information about role transitions

Oracle Data Guard Broker for information about switchover and failover operations

Zero Data Loss Recovery Appliance Administrator's Guide

6.8 Establish Escalation Management Procedures

Establish escalation management procedures so repair is not hindered. Most repair solutions, when conducted properly are automatic and transparent with the MAA solution. The challenges occur when the primary database or system is not meeting availability or performance SLAs and failover procedures are not automatic as in the case with some Data Guard failover scenarios. Downtime can be prolonged if proper escalation policies are not followed and decisions are not made quickly.

If availability is the top priority, execute repair and failover operations first and then proceed with gathering logs and information for Root Cause Analysis (RCA) after the application service has been reestablished. For simple data gathering, use the Trace File Analyzer Collector (TFA).

See Also:

Table 4-1 (page 4-1)

Table 4-2 (page 4-6)

Table 5-7 (page 5-18)

Table 5-8 (page 5-20)

For more information about MAA outage and repair, check the MAA web page on the Oracle Technology Network (OTN) at `http://www.oracle.com/goto/maa`

My Oracle Support note 1513912.2 “TFA Collector - Tool for Enhanced Diagnostic Gathering” at 1513912.2

6.9 Configure Monitoring and Service Request Infrastructure for High Availability

To maintain your High Availability environment, you should configure the monitoring infrastructure that can detect and react to performance and high availability related thresholds before any downtime has occurred. Also, where available, Oracle can detect failures, dispatch field engineers, and replace failed hardware components such as disks, flash cards, fans, or power supplies without customer involvement.

6.9.1 Execute Database Health Checks Periodically

Oracle database health checks are designed to evaluate your hardware and software configuration and MAA compliance to best practices. All of the Oracle health check tools will evaluate Oracle Grid Infrastructure, Oracle Database, and provide an automated MAA scorecard or review that highlights when key architectural and configuration settings are not enabled for tolerance of failures or fast recovery. For Oracle's engineered systems such as Exadata Database Machine, there may be hundreds of additional software, fault and configuration checks.

Oracle recommends periodically (for example, monthly for Exadata Database Machine) downloading the latest database health check, executing the health check, and addressing the key FAILURES, WARNINGS, and INFO messages. Use `exachk` for Engineered Systems such as Oracle Exadata Database Machine, Exalogic, Zero

Data Loss Recovery Appliance, and Big Data Appliance, and use `orachk` for Non-Engineered Systems and Oracle Database Appliance.

Furthermore, it is recommended that you execute the health check prior to and after any planned maintenance activity.

- Evaluate existing or new critical health check alerts prior to planned maintenance window
- Evaluate adding any new recommendations to the planned maintenance window after testing
- Evaluate existing software or critical software recommendations

See Also:

My Oracle Support Note 1268927.2 "ORAchk - Health Checks for the Oracle Stack" at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1268927.2`

My Oracle Support Note 1070954.1 "Oracle Exadata Database Machine exachk or HealthCheck" at `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1070954.1`

6.9.2 Configure Oracle Enterprise Manager Monitoring Infrastructure for High Availability

You should configure and use Enterprise Manager and the monitoring infrastructure that detects and reacts to performance and high availability related thresholds to avoid potential downtime. The monitoring infrastructure assists you with monitoring for High Availability and enables you to do the following:

- Monitor system, network, application, database and storage statistics
- Monitor performance and service statistics
- Create performance and high availability thresholds as early warning indicators of system or application problems
- Provide performance and availability advice
- Established alerts and tools and database performance
- Receive alerts for engineered systems hardware faults

See Also:

Oracle Database High Availability Best Practices for information about monitoring for high availability

Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide for information about detecting and reacting to potential problems and failures

The MAA Best Practices area for Enterprise Manager at http://www.oracle.com/goto/maa for Enterprise Manager and Exadata manageability best practices

6.9.3 Configure Automatic Service Request Infrastructure

In addition to monitoring infrastructure with Enterprise Manager in the Oracle high availability environment where available, Oracle can detect failures, dispatch field engineers, and replace failing hardware without customer involvement. For example, Oracle Automatic Service Request (ASR) is a secure, scalable, customer-installable software solution available as a feature. The software resolves problems faster by using auto-case generation for Oracle's Solaris server and storage systems when specific hardware faults occur.

See Also:

See "Oracle Automatic Service Request" in My Oracle Support Note 1185493.1 at

`https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=1185493.1`

6.10 Check the Latest MAA Best Practices

MAA solutions and best practices continue to be developed and published on the Oracle Technology Network (OTN) at http://www.oracle.com/goto/maa.

For Oracle Database MAA best practices, refer to the Oracle Databases, Exadata Database Machine, Zero Data Loss Recovery Appliance, and Oracle Cloud pages.

The MAA solution encompasses the full stack of Oracle technologies, so you can find MAA best practices for Oracle Fusion Middleware, Oracle Fusion Applications, Oracle Applications Unlimited, Oracle Exalytics, Oracle Exalogic, Oracle VM, and Oracle Enterprise Manager Cloud Control on the MAA pages.

7

High Availability Architectures

Oracle MAA provides best practice recommendations for the design, implementation, and operation of high availability architectures for the Oracle Database residing On-Premise, in the Oracle Public Cloud, or a hybrid of both On-Premise and Oracle Public Cloud.

It includes the following sections:

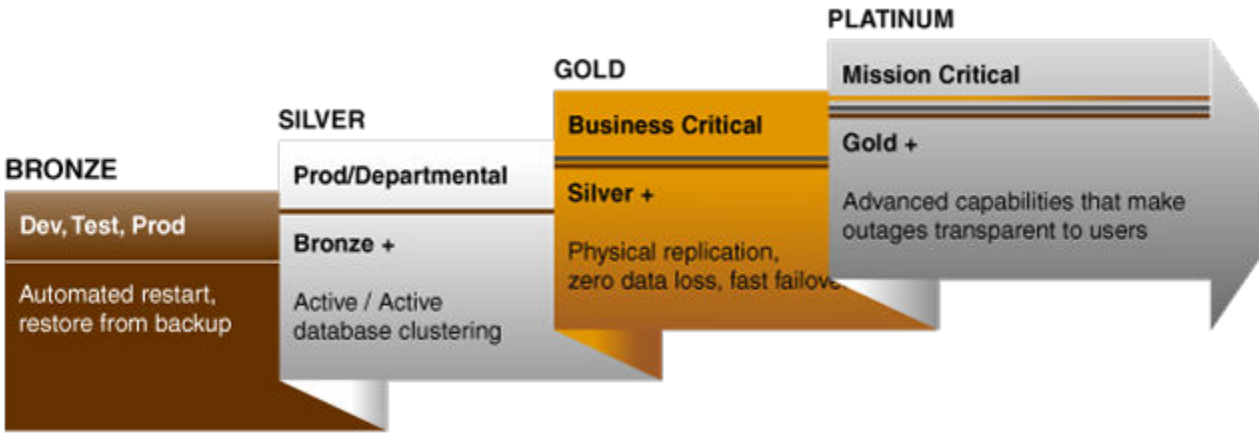
7.1 Introduction to MAA Reference Architectures

Each MAA reference architecture, or high availability tier, utilizes an optimal set of Oracle capabilities that, when deployed together, reliably achieve a given service level for high availability and data protection.

The Oracle Maximum Availability Architecture offers a choice of architecture patterns for high availability and scalability:

- A set of standard reference architectures, Bronze, Silver, Gold, and Platinum, that provide application transparent scalability (with Oracle RAC), data protection, high availability, and disaster recovery for the Oracle Database. The figure below illustrates the technologies used by each tier.
- A special-purpose reference architecture that uses Oracle Sharding for linear scalability with complete fault isolation. The Oracle Sharding MAA reference architecture, introduced in Oracle Database 12*c* Release 2, is a separate MAA reference architecture that is only applicable to shard-ready applications. The Oracle Sharding reference architecture uses these same standard Bronze, Silver, Gold, and Platinum reference architectures as building blocks to provide shard-level high availability, given that each shard is a standalone Oracle Database.

Figure 7-1 Oracle MAA Reference Architectures

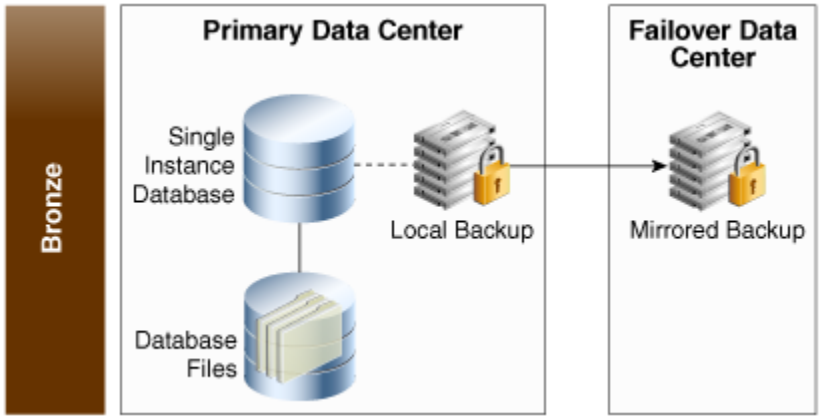


Each of the architectures described above are implemented using the operational and configuration best practices described in Operational Prerequisites to Maximizing Availability (page 6-1), additional best practices are provided in MAA technical white papers on the Oracle Maximum Availability Architecture web site, and in *Oracle Database High Availability Best Practices*. See Roadmap to Implementing the Maximum Availability Architecture (page 1-7), for more information that will help you navigate MAA best practices documentation.

7.2 The Bronze Tier – A Single Instance HA Architecture

The Bronze tier provides basic database service at the lowest possible cost. A reduced level of HA and data protection is accepted in exchange for reduced cost and implementation complexity. The following figure provides an overview of the Bronze tier.

Figure 7-2 Bronze Tier – Single Instance HA Architecture



Bronze uses a single instance Oracle Database; there is no clustering technology used for automatic failover if there is an outage of the server on which the Oracle Database instance is running. When a server becomes unusable or the database unrecoverable, RTO is a function of how quickly a replacement system can be provisioned or a backup restored. In a worst case scenario of a complete site outage there will be additional time required to perform these tasks at a secondary location, and in some cases this can take days.

Oracle Recovery Manager (RMAN) is used to perform regular backups of the Oracle Database. The RPO, if there is an unrecoverable outage, is equal to the data generated since the last backup was taken. Copies of database backups are also retained at a remote location or on the Cloud for the dual purpose of archival and disaster recovery should a disaster strike the primary data center.

The Bronze tier is comprised of the major components described in the following topics:

7.2.1 Oracle Database HA and Data Protection

Bronze utilizes HA and data protection capabilities that are included with the Oracle Database Enterprise Edition at no additional cost.

- Oracle Restart automatically restarts the database, the listener, and other Oracle components after a hardware or software failure, or whenever a database host computer restarts.
- Oracle corruption protection checks for physical corruption and logical intra-block corruptions. In-memory corruptions are detected and prevented from being written to disk, and in many cases can be repaired automatically. For more details see Preventing, Detecting, and Repairing Block Corruption for the Oracle Database.
- Automatic Storage Management (ASM) is an Oracle-integrated file system and volume manager that includes local mirroring to protect against disk failure.
- Oracle Flashback Technologies provide fast error correction at a level of granularity that is appropriate to repair an individual transaction, a table, or the full database.
- Oracle Recovery Manager (RMAN) enables low-cost, reliable backup and recovery optimized for the Oracle Database.
- Online maintenance includes online redefinition and reorganization for database maintenance, online file movement, and online patching.

7.2.2 Database Consolidation in the Bronze Tier

Databases deployed in the Bronze tier include development and test databases and databases supporting smaller work group and departmental applications that are often the first candidates for database consolidation and for deployment as Database as a Service (DBaaS).

Oracle Multitenant is the MAA best practice for database consolidation and virtualization from Oracle Database 12*c* onward. Other consolidation options include:

- Operating System Virtualization - Virtual Machines
- Schema Consolidation

- Consolidation of multiple discrete databases onto a single physical machine or cluster using Oracle RAC

See Also:

MAA white paper High Availability Best Practices for Database Consolidation for a complete discussion of the trade-offs between Oracle Multitenant and other consolidation approaches

Oracle Database Administrator's Guide for information about managing a multitenant environment

7.2.3 Life Cycle Management and DBaaS

Oracle Enterprise Manager Cloud Control enables self service deployment of IT resources for business users along with resource pooling models that cater to various multitenant architectures. These capabilities are required for implementing Database as a Service (DBaaS), a paradigm in which end users (Database Administrators, Application Developers, Quality Assurance Engineers, Project Leads, and so on) can request database services, consume it for the lifetime of the project, and then have them automatically de-provisioned and returned to the resource pool. Cloud Control Database as a Service (DBaaS) provides:

- A shared, consolidated platform on which to provision database services
- A self-service model for provisioning those resources
- Elasticity to scale out and scale back database resources
- Chargeback based on database usage

7.2.4 Oracle Engineered Systems

Oracle Engineered Systems are an efficient deployment option for database consolidation and DBaaS at all tiers. Oracle Engineered Systems reduce lifecycle cost by standardizing on a pre-integrated and optimized platform for Oracle Database, with hardware and software supported by Oracle. Oracle Engineered Systems include:

- Oracle Virtual Compute Appliance radically simplifies the way customers install, deploy, and manage virtual infrastructures for any Linux, Oracle Solaris, or Microsoft Windows application.
- Oracle Database Appliance is a complete low cost package of software, server, storage, and networking engineered for simplicity, saving time and money by simplifying deployment, maintenance, and support of database and application workloads. The Oracle Database Appliance supports both physical and virtual deployments.
- Oracle Exadata Database Machine is the highest performing, most scalable, and most available platform for running Oracle Database. Oracle Exadata Database Machine runs all types of database workloads including Online Transaction Processing (OLTP), Data Warehousing (DW), and consolidation of mixed workloads, and it is the ideal foundation for database consolidation.
- Oracle SuperCluster engineered systems are ideal for consolidating databases and applications, private cloud deployments, and Oracle software on a single, general

purpose platform. Oracle SuperCluster uses the world's fastest processors based on SPARC architecture and Exadata storage.

- Oracle ZFS Storage Appliance provides immediate space, management, and cost benefits for customers using network-attached storage (NAS). Oracle ZFS includes a rich software suite for managing, monitoring, troubleshooting, snaps, clones, replication, and advanced data services that are a natural complement to all Oracle Engineered Systems.
- Zero Data Loss Recovery Appliance (Recovery Appliance) is an Engineered System designed to dramatically reduce data loss and backup overhead for all Oracle databases in the enterprise. Integrated with Recovery Manager (RMAN), the Recovery Appliance enables a centralized, incremental-forever backup strategy for large numbers of databases, using cloud-scale, fault-tolerant hardware and storage. The Recovery Appliance continuously validates backups for recoverability and offloads the continuous compression, deduplication and validation from the database servers.

  Recovery Appliance is Oracle's strategic backup and recovery system that is designed to protect all Oracle databases within a data center or in your private cloud. Recovery Appliance can also replicate to a different Recovery Appliance in a different site for disaster recovery.

7.2.5 Bronze Summary: Data Protection, RTO, and RPO

Table 7-1 (page 7-5) summarizes the data protection capabilities of the Bronze tier. The first column of Table 7-1 (page 7-5) indicates when validations for physical and logical corruption are performed.

- Manual checks are initiated by the administrator or at regular intervals by a scheduled job that performs periodic checks.
- Runtime checks are automatically executed on a continuous basis by background processes while the database is open.
- Background checks are run on a regularly scheduled interval, but only during periods when resources would otherwise be idle.

Each check is unique to Oracle Database using specific knowledge of Oracle data block and redo structures.

Table 7-1 Bronze Tier Data Protection

Type	Capability	Physical Block Corruption	Logical Block Corruption
Manual	Dbverify, Analyze	Physical block checks	Logical checks for intra-block and inter-object consistency
Manual	RMAN	Physical block checks during backup and restore	Intra-block logical checks
Runtime	Database	In-memory block and redo checksum	In-memory intra block logical checks

Table 7-1 (Cont.) Bronze Tier Data Protection

Type	Capability	Physical Block Corruption	Logical Block Corruption
Runtime	ASM and Exadata	Automatic corruption detection and repair using local extent pairs	
Runtime	Exadata, SuperCluster, and Recovery Appliance	HARD checks on write	HARD checks on write
Background	Exadata, SuperCluster, and Recovery Appliance	Automatic HARD Disk Scrub and Repair[1]	n/a
Background	Recovery Appliance	Complete backup validation including control file, data file backups, and redo	n/a

[1] Available with Exadata 11.2.3.3 and later and Oracle Database 11*g* Release 2 (11.2.0.4) and later.

Note that HARD validation and the Automatic Hard Disk Scrub and Repair (the last two rows of Table 2) are unique to Exadata storage. HARD validation ensures that Oracle Database does not write physically corrupt blocks to disk. Automatic Hard Disk Scrub and Repair inspects and repairs hard disks with damaged or worn out disk sectors (cluster of storage) or other physical or logical defects periodically when there are idle resources. Exadata sends a request to ASM to repair the bad sectors by reading the data from another mirror copy. By default the hard disk scrub runs every two weeks.

Table 3 summarizes RTO and RPO for the Bronze tier for various unplanned and planned outages.

Table 7-2 Bronze Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Unplanned	Database instance failure	Minutes	Zero
Unplanned	Recoverable server failure	Minutes to an hour	Zero
Unplanned	Data corruptions, unrecoverable server failure, database failures or site failures	Hours to days	Since last backup or Zero to Near Zero with Recovery Appliance
Planned	Online File Move, Online Reorganization and Redefinition, Online Patching	Zero	Zero

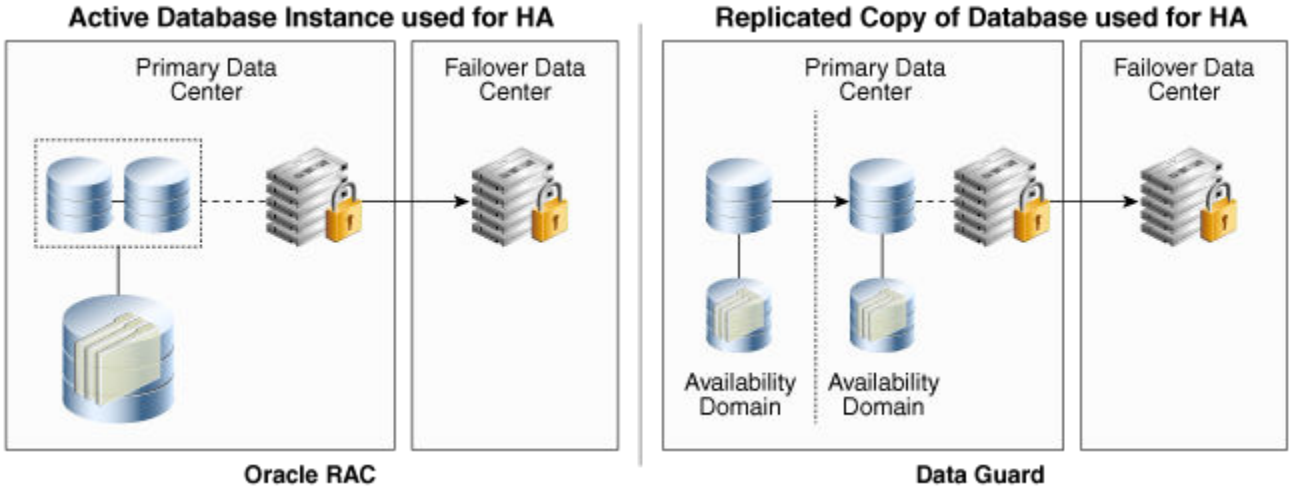
Table 7-2 (Cont.) Bronze Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Planned	Hardware or operating system maintenance and database patches that cannot be done online	Minutes to hours	Zero
Planned	Database upgrades: patch sets and full database releases	Minutes to hours	Zero
Planned	Platform migrations	Hours to a day	Zero
Planned	Application upgrades that modify back-end database objects	Hours to days	Zero

7.3 The Silver Tier - High Availability with Automatic Failover

The Silver tier builds upon Bronze by incorporating clustering technology for improved availability for both unplanned outages and planned maintenance. Silver uses Oracle RAC or Oracle RAC One Node for HA within a data center by providing automatic failover should there be an unrecoverable outage of a database instance or a complete failure of the server on which it runs. Oracle RAC also delivers substantial benefit for eliminating many types of planned downtime by performing maintenance in a rolling manner across Oracle RAC nodes. The following figure provides an overview of the Silver tier.

Figure 7-3 Silver Tier – High Availability with Automatic Failover



Silver includes the HA components described in the following sections.

- Oracle RAC (page 7-8)
- Oracle RAC One Node (page 7-9)

- Silver Tier Summary: Data Protection, RTO, and RPO (page 7-9)

7.3.1 Oracle RAC

Oracle RAC improves application availability within a data center should there be an outage of a database instance or of the server on which it runs. Server failover with Oracle RAC is instantaneous. There is a very brief brownout before service is resumed on surviving instances and users from the down instance are able to reconnect. Downtime is also eliminated for planned maintenance tasks that can be performed in a rolling manner across Oracle RAC nodes. Users complete their work and terminate their sessions on the node where maintenance is to be performed. When they reconnect they are directed to a database instance already running on another node.

A quick review of how Oracle RAC works helps to understand its benefits. There are two components: Oracle Database instances and the Oracle Database itself.

- A database instance is defined as a set of server processes and memory structures running on a single node (or server) which make a particular database available to clients.
- The database is a particular set of shared files (data files, index files, control files, and initialization files) that reside on persistent storage, and together can be opened and used to read and write data.
- Oracle RAC uses an active-active architecture that enables multiple database instances, each running on different nodes, to simultaneously read and write to the same database.

The active-active architecture of Oracle RAC provides a number of advantages:

- Improved high availability: If a server or database instance fails, connections to surviving instances are not affected; connections to the failed instance are quickly failed over to surviving instances that are already running and open on other servers in the cluster.
- Scalability: Oracle RAC is ideal for high volume applications or consolidated environments where scalability and the ability to dynamically add or reprioritize capacity across more than a single server are required. An individual database may have instances running on one or more nodes of a cluster. Similarly, a database service may be available on one or more database instances. Additional nodes, database instances, and database services can be provisioned online. The ability to easily distribute workload across the cluster makes Oracle RAC the ideal complement for Oracle Multitenant.
- Reliable performance: Oracle Quality of Service (QoS) can be used to allocate capacity for high priority database services to deliver consistent high performance in database consolidated environments. Capacity can be dynamically shifted between workloads to quickly respond to changing requirements.
- HA during planned maintenance: High availability is maintained by implementing changes in a rolling manner across Oracle RAC nodes. This includes hardware, OS, or network maintenance that requires a server to be taken offline; software maintenance to patch the Oracle Grid Infrastructure or database; or if a database instance needs to be moved to another server to increase capacity or balance the workload.

Oracle RAC is the MAA best practice for server HA.

7.3.2 Oracle RAC One Node

Oracle RAC One Node provides an option to Oracle RAC in the Silver tier when server HA is a requirement, but scalability and instant failover are not. An Oracle RAC One Node license is one-half the cost of Oracle RAC, providing a lower cost alternative if an RTO of minutes is sufficient for managing server failures.

Oracle RAC One Node is an active-passive failover technology. It is built upon an infrastructure that is identical to Oracle RAC, but in the case of Oracle RAC One Node there is only one database instance open at a time during normal operation. This can reduce memory requirements significantly, especially when consolidating a large number of databases. If the server hosting the open instance fails, Oracle RAC One Node automatically starts a new database instance on a second node to quickly resume service.

Oracle RAC One Node provides several advantages over alternative active-passive clustering technologies. In an Oracle RAC One Node configuration, Oracle Database HA Services, Grid Infrastructure, and database listeners are always running on the second node. At failover time only the database instance and database services need to start, reducing the time required to resume service, and enabling service to resume in minutes.

Oracle RAC One Node also provides the same advantages for planned maintenance as Oracle RAC. Oracle RAC One Node allows two active database instances during periods of planned maintenance to allow graceful migration of users from one node to another with zero downtime. Maintenance is performed in a rolling manner across nodes while database services remain available to users at all times.

7.3.3 Silver Tier Summary: Data Protection, RTO, and RPO

There is no change in the level of data protection compared to what is offered by the Bronze tier. All of the improvements that Silver offers compared to Bronze are related to RTO for server outages and for several frequently executed types of planned maintenance. Table 7-3 (page 7-9) summarizes RTO and RPO enabled by the Silver tier. Areas of improvement compared to Bronze are in parentheses.

Table 7-3 Silver Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Unplanned	Database instance failure	Seconds if Oracle RAC (instead of minutes)	Zero
Unplanned	Recoverable Server failure	Seconds if Oracle RAC (instead of minutes to an hour) Minutes if Oracle RAC One Node (instead of minutes to an hour)	Zero Zero
Unplanned	Data corruptions, unrecoverable server failure, database or site failures	Hours to days	Since last backup, or zero or near zero with Recovery Appliance

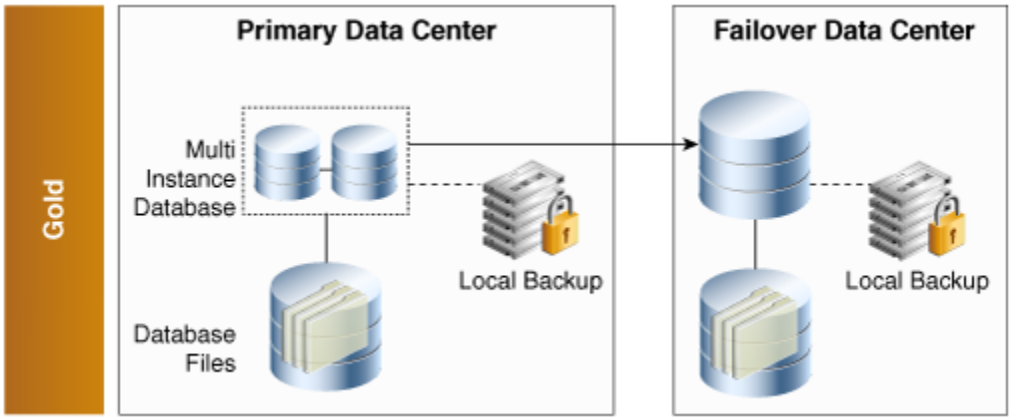
Table 7-3 (Cont.) Silver Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Planned	Online File Move, Online Reorganization and Redefinition, Online Patching	Zero	Zero
Planned	Hardware or operating system maintenance and database patches that cannot be done online but are qualified for Oracle RAC rolling install	Zero (instead of minutes to hours)	Zero
Planned	Database upgrades: patch sets and full database releases	Minutes to hours	Zero
Planned	Platform migrations	Hours to a day	Zero
Planned	Application upgrades that modify back-end database objects	Hours to days	Zero

7.4 The Gold Tier - Comprehensive High Availability and Disaster Recovery

The Gold tier builds upon Silver by using database replication technology to eliminate single point of failure and provide a much higher level of data protection and HA from all types of unplanned outages including data corruptions, database failures, and site failures. The existence of a replicated copy also provides substantial advantages for reducing downtime during periods of planned maintenance. RTO is reduced to seconds or minutes with an accompanying RPO of zero or near zero depending upon configuration. An overview of the Gold tier is shown in the following figure.

Figure 7-4 Gold Tier – Comprehensive HA and DR



Note that Gold uses Oracle RAC as the standard for server high availability instead of the lesser option of Oracle RAC One Node that is available for Silver.

The Gold tier adds advanced high availability components to achieve improved service levels described in the following sections.

7.4.1 Oracle Active Data Guard - Real Time Data Protection and Availability

Oracle Active Data Guard maintains one or more synchronized physical replicas (standby databases) at a remote location that are used to eliminate single point of failure for a production database (the primary database). Capabilities that Oracle Active Data Guard adds to the Gold tier include:

- Choice of zero or near-zero data loss potential. Oracle Active Data Guard performs real-time replication of changes from a primary to a standby database. Changes are transmitted directly from the log buffer of the primary to minimize propagation delay and overhead, and to completely isolate replication from corruptions that can occur in the I/O stack of a production database.

  Administrators can choose synchronous transport with Maximum Availability for a guarantee of zero data loss. Alternatively they can choose asynchronous transport and Maximum Performance for near-zero data loss. Maximum Performance can achieve sub-second data loss exposure when provided sufficient network bandwidth to accommodate transport volume.

  Data Guard is the Oracle replication technology that provides zero data loss protection.

- An Oracle Active Data Guard standby database can quickly take over production and restore service if there is a database or site outage that impacts the availability of the primary database. The Oracle Database is always running, it does not need to be restarted to transition to the primary role, and role transitions can complete in less than 60 seconds, even on heavily loaded systems.

  Gold utilizes Data Guard Fast-Start Failover to automate database failover. This accelerates recovery time by eliminating the delay required for an administrator to be notified and respond to an outage. Fast Start Failover uses role-specific database services and the Oracle client notification framework to ensure that applications quickly drop their connections to a failed primary database and automatically reconnect to the new primary. Role transitions can also be executed manually using either a command line interface or Oracle Enterprise Manager.

- Transparent replication. Oracle Active Data Guard performs complete, one-way physical replication of an Oracle Database with the following characteristics: high performance, simple to manage, support for all data types, applications, and workloads such as DML, DDL, OLTP, batch processing, data warehouse, and consolidated databases. Oracle Active Data Guard is closely integrated with Oracle RAC, ASM, RMAN and Oracle Flashback technologies.

- Production offload for high return on investment (ROI). Oracle Active Data Guard standby databases can be opened read-only while replication is active, and they can be used to offload ad-hoc queries and reporting workloads from the production database. The offload increases ROI in standby systems and improves performance for all workloads by utilizing capacity that would otherwise be idle. It also

provides continuous application validation because the standby systems are ready to support production workloads.

- Backup offload. Primary and standby systems are exact physical replicas, enabling backups to be offloaded from the primary to the standby database. A backup taken at the standby can be used to restore either the primary or standby database. This provides administrators with flexible recovery options without burdening production systems with the overhead of performing backups.
- Reduced downtime for planned maintenance. Standby databases can be used to upgrade to new Oracle Patch Sets (for example, patch release 11.2.0.2 to 11.2.0.4) or new Oracle releases (for example, release 11.2 to 12.1) in a rolling manner by implementing the upgrade at the standby first then switching production to the new version. Total downtime is limited to the time required to switch a standby database to the primary production role after maintenance has been completed.
- An Oracle Active Data Guard standby performs continuous Oracle validation to ensure that corruption is not propagated from the source database. It detects physical and logical intra-block corruptions that can occur independently at either primary or standby databases. It is also unique in enabling run-time detection of silent lost-write corruptions (lost or stray writes that are acknowledged by the I/O subsystem as successful). For more details see My Oracle Support Note 1302539.1 - Best Practices for Corruption Detection, Prevention, and Automatic Repair.
- Automatic block repair. Oracle Active Data Guard automatically repairs block-level corruption caused by intermittent random I/O errors that can occur independently at either primary or standby databases. It does this by retrieving a good copy of the block from the opposite database. No application changes are required and the repair is transparent to the user.

The points above explain how the Gold tier utilizes Oracle replication technology to maintain a synchronized copy, rather than using storage remote mirroring products (for example, SRDF, Hitachi TrueCopy, and so on) For a more in-depth discussion of the differences see Oracle Active Data Guard vs. Storage Remote Mirroring.

7.4.2 Oracle GoldenGate

Oracle GoldenGate provides the option of logical replication to maintain a synchronized copy (target database) of the production database (source database). Logical replication is a more complex process than physical replication but provides greater flexibility to handle different replication scenarios and heterogeneous platforms.

- From a data distribution perspective, logical replication is designed to efficiently replicate subsets of a source database to distribute data to other target databases. It can also be used to consolidate data into a single target database (for example, an Operational Data Store) from multiple source databases.
- From a high availability perspective, logical replication can be used to maintain a complete replica of a source database for high availability or disaster protection that is ready for immediate failover should the source database become unavailable. Oracle GoldenGate uses a logical replication process. It reads changes from disk at a source database, transforms the data into a platform independent file format, transmits the file to a target database, then transforms the data into SQL (updates, inserts, and deletes) native to the target database. The target database contains the same data, but is a different database from the source (for example, backups are not interchangeable).

- Oracle GoldenGate logical replication provides increased flexibility to perform maintenance and migrations in a rolling manner that is not possible using Data Guard physical replication. For example, Oracle GoldenGate enables replication of a source running on a big-endian platform and target running on a little-endian platform (cross-endian replication). This makes it possible to execute platform migrations with the additional advantage of being able to reversing the replication for fast fallback to the prior version after cutover.

Oracle GoldenGate logical replication is a more sophisticated process that has a number of prerequisites that do not apply to Data Guard physical replication. In return for these prerequisites Oracle GoldenGate provides unique capabilities to address advanced replication requirements. Refer to MAA Best Practices: Oracle Active Data Guard and Oracle GoldenGate for additional insights on the tradeoffs of each replication technology and requirements that may favor the use of one versus the other, or the use of both technologies in a complementary manner.

7.4.3 Oracle Site Guard

Oracle Site Guard enables administrators to orchestrate switchover (a planned event) and failover (in response to an unplanned outage) of their Oracle environment, multiple databases, and applications, between a production site and a remote disaster recovery site. Oracle Site Guard is licensed as part of the Oracle Enterprise Manager WebLogic Server Management Pack Enterprise Edition and the Oracle Database Lifecycle Management Pack.

Oracle Site Guard offers the following benefits:

- Reduction of errors due to prepared response to site failure. Oracle Site Guard reduces the possibility of human error in case of disasters. Recovery strategies are mapped out, tested, and rehearsed in prepared responses within the application. Once an administrator initiates a Site Guard operation for disaster recovery, human intervention is not required.
- Coordination across multiple applications, databases, and various replication technologies. Oracle Site Guard automatically handles dependencies between different targets while starting or stopping a site. Site Guard integrates with Oracle Active Data Guard to coordinate multiple concurrent database failovers. Site Guard also provides an easy mechanism to integrate with any storage remote mirroring product. It integrates with storage appliances to perform switchover or failover by using callouts to any user-specified storage role reversal scripts in the operation workflow.
- Faster recovery time. Oracle Site Guard automation minimizes the manual coordination of recovery activities. This accelerates recovery time even compared to the case where all manual efforts are executed successfully. Site Guard also avoids time consuming resolution of human error that often accompanies manual implementation of complex procedures.

7.4.4 Gold Summary: Data Protection, RTO, and RPO

Table 7-4 (page 7-14) summarizes the data protection offered by the Gold tier.

Table 7-4 Gold Tier Data Protection

Type	Capability	Physical Block Corruption	Logical Block Corruption
Manual	Dbverify, Analyze	Physical block checks	Logical checks for intra-block and inter-object consistency
Manual	RMAN	Physical block checks during backup and restore	Intra-block logical checks
Runtime	Oracle Active Data Guard	Physical block checking at standby Strong isolation between primary and standby eliminates single point of failure Automatic repair of physical corruptions Automatic database failover	Detect lost write corruption, auto shutdown and failover Intra-block logical checks at standby
Runtime	Database	In-memory block and redo checksum	In-memory intra block logical checks
Runtime	ASM and Exadata	Automatic corruption detection and repair using local extent pairs	
Runtime	Exadata, SuperCluster, and Recovery Appliance	HARD checks on write	HARD checks on write
Background	Exadata, SuperCluster, and Recovery Appliance	Automatic Hard Disk Scrub and Repair	n/a
Background	Recovery Appliance	Complete backup validation including control file, data file backups, and redo	n/a

Table 7-5 (page 7-14) summarizes RTO and RPO for the Gold tier. Recovery time and data loss potential are dramatically reduced in the Gold tier compared to Silver. Areas of improvement compared to the Silver tier are in parentheses.

Table 7-5 Gold Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Unplanned	Database instance failure	Seconds	Zero

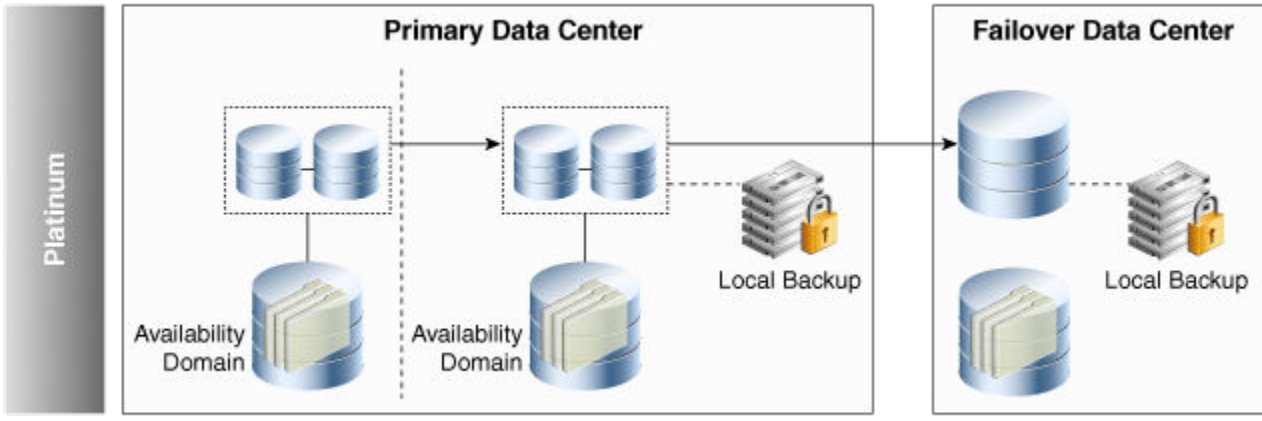
Table 7-5 (Cont.) Gold Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Unplanned	Recoverable server failure	Seconds	Zero
Unplanned	Data corruptions, unrecoverable server failure, database failures or site failures	Zero to minutes (instead of hours to days)	Near-zero if using ASYNC (instead of since last backup) Zero if using Data Guard synchronous transport (instead of since last backup)
Planned	Online File Move, Online Reorganization and Redefinition, Online Patching	Zero	Zero
Planned	Hardware or operating system maintenance and database patches that cannot be done online	Zero	Zero
Planned	Database upgrades: patch sets and full database releases	Seconds (instead of minutes to hours)	Zero
Planned	Platform migrations	Seconds (instead of hours to a day)	Zero
Planned	Application upgrades that modify back-end database objects	Hours to days	Zero

7.5 The Platinum Tier - Zero Outage for Platinum Ready Applications

The Platinum tier builds upon Gold to provide the highest level of HA and data protection for applications that have zero tolerance for outages or data loss. Platinum introduces several new Oracle Database 12*c* capabilities as well as previously available products that have been enhanced with the latest release. Platinum masks the impact of outages to applications and users, ensuring that even in-flight transactions are preserved following recoverable failures. It enables zero downtime maintenance, migrations, and application upgrades. It guarantees zero data loss in the event of failure of the primary database for any reason, regardless of the distance between sites. Finally, Platinum automatically manages the availability of database services and workload load balancing across database replicas in multiple sites. An overview of the Platinum tier is provided in the following figure.

Figure 7-5 Platinum Tier – Zero Outage



Some applications will require a level of modification to achieve zero application outage using the capabilities provided by the Platinum tier. This explains why Platinum is described as providing zero application outage for Platinum-Ready Applications. Note that no application modifications are necessary in order to achieve zero data loss.

The Platinum tier enables the HA capabilities described in the following sections.

- Application Continuity (page 7-16)
- Oracle Active Data Guard Far Sync (page 7-16)

  Oracle GoldenGate Zero Downtime Maintenance and Active-Active Replication (page 7-17)
- Edition Based Redefinition (page 7-18)
- Global Data Services (page 7-18)
- Platinum Summary: Data Protection, RTO, and RPO (page 7-19)

7.5.1 Application Continuity

Application Continuity protects applications from database session failures due to instance, server, storage, network, or any other related component, and even complete database failure. Application Continuity re-plays affected "in-flight" requests so that the failure appears to the application as a slightly delayed execution, masking the outage to the user.

If an entire Oracle RAC cluster fails, making the database unavailable, Application Continuity will replay the session including the transaction, following an Oracle Active Data Guard failover. Use of Application Continuity with a standby database requires Data Guard Maximum Availability mode (zero data loss) and Data Guard Fast Start Failover (automatic database failover).

While in many cases there is some modification to existing application code required to use Application Continuity, it simplifies development of new applications by transparently handling recoverable failures.

7.5.2 Oracle Active Data Guard Far Sync

Oracle Active Data Guard is the only Oracle-aware replication technology that offers zero data loss failover for Oracle Database. Zero data loss is achieved using

synchronous transport with Data Guard Maximum Availability mode. Network latency between primary and standby sites will affect database performance when synchronous transport is used. As distance between site increases, so will latency and its impact on database performance. Because primary and secondary data centers are often separated by long distances, zero data loss failover is impractical to implement for many databases.

Oracle Active Data Guard Far Sync with Oracle Database 12c eliminates prior limitations by enabling zero data loss failover even when primary and standby databases are hundreds or thousands of miles apart, without impacting primary database performance. It achieves this by using a light-weight forwarding mechanism that is simple to deploy and transparent to Oracle Active Data Guard failover or switchover operations. Far Sync, when used in combination with the Oracle Advanced Compression Option, also enables off-host transport compression to conserve network bandwidth.

By combining Far Sync with Data Guard Fast-Start-Failover (automatic database failover), Application Continuity can mask outages for in-flight transactions regardless of the distance between primary and stand by sites. Far Sync, therefore, enables two critical enhancements offered by the Platinum tier: zero data loss failover for any database and the ability to use Application Continuity regardless of the distance between sites. There are no application modifications required to take advantage of Far Sync.

7.5.3 Oracle GoldenGate Zero Downtime Maintenance and Active-Active Replication

The Platinum tier uses Oracle GoldenGate's advanced replication capabilities to implement zero downtime maintenance and migrations using bi-directional replication. In such a scenario:

- Maintenance is first implemented at a target database.
- Source and target are synchronized across versions using Oracle GoldenGate logical replication. This handles cross-endian platform migrations. It also handles complex application upgrades that modify back-end objects where the replication mechanism must be able to transform data from old to new versions and vice versa.
- Once the new version or platform is synchronized and stable, the bi-directional replication enables users to be gradually migrated to the new platform as they terminate sessions on the previous version and reconnect, providing a zero downtime experience. Oracle GoldenGate bi-directional replication keeps old and new versions in sync during the migration process. This also provides for a quick fall back option should any unanticipated issues arise with the new version as load is added.

Active-active bi-directional replication can also be used to increase availability service levels where a continuous read-write connection to multiple copies of the same data is required.

Bi-directional replication is not application transparent. It requires conflict detection and resolution when changes are made to the same record at the same time in multiple databases. It also requires careful consideration of the impact of different failure states and replication lag. When GoldenGate bi-directional replication is used for application upgrades that modify back-end database objects, developer-level knowledge of the database objects modified or added by the new release is required in order to enable GoldenGate to replicate across versions. Implementing cross-version mapping is required for every new release of the application.

As GoldenGate replication is by definition an asynchronous process, it is not able to provide zero data loss protection. For this reason the Platinum tier does not use Oracle GoldenGate to replicate between sites when the remote replica must provide zero data loss protection if the primary database or primary site experiences an unplanned outage. Platinum uses GoldenGate bi-directional replication in combination with Oracle Active Data Guard to meet the zero data loss requirement. A local GoldenGate replica is used to execute planned maintenance with zero downtime while an Oracle Active Data Guard standby provides continuous zero data loss failover protection should an unplanned outage occur while maintenance is in progress.

7.5.4 Edition Based Redefinition

Edition-Based Redefinition (EBR) enables an online application upgrade that changes back-end database objects with uninterrupted availability of the application. When an upgrade installation is complete, the pre-upgrade application and the post-upgrade application can be used at the same time. Existing sessions can continue to use the pre-upgrade application until their users decide to end them, and all new sessions can use the post-upgrade application. When there are no longer any sessions using the pre-upgrade application, it can be retired. EBR used in this manner enables hot rollover from the pre-upgrade version to the post-upgrade version.

EBR enables online application upgrades in the following manner:

- Code changes are installed in the privacy of a new edition.
- Data changes are made safely by writing only to new columns or new tables not seen by the old edition. An editioning view exposes a different projection of a table into each edition to allow each to see just its own columns.
- A cross-edition trigger propagates data changes made by the old edition into the new edition's columns, or (in hot-rollover) vice-versa.

Similar to Oracle GoldenGate zero downtime application upgrades, the use of EBR requires deep knowledge of the application and a non-trivial effort on the part of the developer to incorporate it. Unlike Oracle GoldenGate, there is a one-time investment to utilize EBR. From that point forward minimal investment is required to use EBR for subsequent new releases of the application. EBR has proven that it can be implemented even for the most complex applications, for example, Oracle E-Business Suite 12.2 uses EBR for online patching. EBR is a feature included with Oracle Database as a zero cost option to encourage its adoption by application developers.

7.5.5 Global Data Services

Global Data Services (GDS) is a complete automated workload management solution for replicated databases that use Oracle Active Data Guard or Oracle GoldenGate. GDS achieves better system utilization and offers better performance, scalability, and availability for application workloads running on replicated databases. GDS provides the following capabilities for a set of replicated databases:

- Region-based workload routing
- Connect-time load balancing
- Run-time load balancing advisory for Oracle integrated clients
- Inter-database service failover
- Replication lag based workload routing for Oracle Active Data Guard

- Role-based global services for Oracle Active Data Guard
- Centralized workload management framework

7.5.6 Platinum Summary: Data Protection, RTO, and RPO

The Platinum tier provides the same corruption protection as the Gold tier. The differences between the Platinum and Gold tiers are recovery time (RTO) and data loss potential (RPO) for Platinum-ready applications. RTO/RPO for the Platinum tier is summarized in Table 7-6 (page 7-19).

Table 7-6 Platinum Tier Recovery Time (RTO) and Data Loss Potential (RPO)

Type	Event	Downtime	Data Loss Potential
Unplanned	Database instance failure	Zero application outage (vs. seconds)	Zero
Unplanned	Recoverable server failure	Zero application outage (vs. seconds)	Zero
Unplanned	Data corruptions, unrecoverable server failure, database failures or site failures	Zero application outage (vs. zero to minutes)	Zero (vs. near-zero)
Planned	Online File Move, Online Reorganization and Redefinition, Online Patching	Zero	Zero
Planned	Hardware or operating system maintenance and database patches that cannot be done online	Zero application outage	Zero
Planned	Database upgrades: patch sets and full database releases	Zero application outage (vs. seconds)	Zero
Planned	Platform migrations	Zero application outage (vs. seconds)	Zero
Planned	Application upgrades that modify back-end database objects	Zero application outage (vs. hours to days)	Zero

7.6 Oracle Database Sharding Reference Architecture

Oracle Sharding is a true shared-nothing architecture that provides linear scalability and high availability by distributing data and workloads across a pool of independent Oracle databases known as shards.

The pool of shards is presented to the application as a single logical database. The single logical database is known as a sharded database. Applications elastically scale

(data, transactions, and users) to any level, on any platform, simply by adding shards to the sharded database. Data and workloads are automatically balanced across the shards transparent to the application. Scaling a sharded database up to 1,000 shards is supported in the first release of Oracle Database 12*c* Release 2.

Oracle Sharding uses a sharded database to provide linear scalability and fault isolation for suitable applications. A sharded database eliminates the possibility of a single physical database being unable to scale to meet application requirements. Similarly, a sharded database prevents a physical database from being a single point of failure for an application due to unplanned outages or planned maintenance.

The Oracle Sharding MAA reference architecture uses the Bronze, Silver, Gold, and Platinum MAA reference architectures as building blocks to provide shard-level high availability given that each shard is a standalone Oracle Database:

- Bronze: Database restart and backups for recovery of a shard
- Silver: Bronze, plus Oracle RAC or Oracle Active Data Guard for shard-level high availability
- Gold: Silver, plus Oracle Active Data Guard or Oracle GoldenGate for shard-level high availability and disaster recovery
- Platinum: Gold, plus advanced Oracle features for shard-level high availability to make all unplanned outages, and even the most complex planned maintenance tasks, completely transparent to an application

The Oracle Sharding MAA reference architecture also includes best practices that address several unique considerations for a sharded database.

Oracle Sharding is a special purpose architecture because applications must have the following characteristics in order to benefit from Oracle Sharding:

- OLTP applications with high transaction volumes which require low latency and extreme fault isolation. The current release of Oracle Sharding is not intended for data warehouse or analytical applications.
- Data for OLTP applications has to be partitionable on a stable key, for example, customer ID, and mostly accessed using the key.
- For each key-based request, the application should establish a new database session and provide the key using the API provided with Oracle Sharding.

The effort required to use Oracle Sharding depends on the design of the application and the data model. For example:

- New OLTP applications can be easy to build. Oracle Sharding provides a simple declarative way of specifying sharded table families and duplicated tables. There are no routing or multi-shard query capabilities required in the application. Administrators can add or subtract shards, and the sharding infrastructure will rebalance data and workload automatically (for system-managed sharding). Applications never need to know how many shards there are or how data is distributed across them. Oracle Sharding provides a convenient API for providing the sharding key, load balancing across shard replicas, and so on.
- Home grown OLTP applications that were designed to shard require some amount of change in order to achieve the benefits of Oracle Sharding. Instead of using existing routing code, the application should use Oracle Sharding APIs. This may be a simple or more complex change depending on how closely integrated the home-grown routing code is with the application.

- Commercial off-the-shelf applications or home grown OLTP applications that were never designed to shard can prove challenging to convert. Such applications need to change their database requests to access data by sharding key. The application should also eliminate global secondary indexes and integrity constraints that have to be enforced across shards and global sequences. Existing databases might require denormalization. The root table and all child tables must contain the sharding key. In spite of these challenges, customers with existing Oracle applications who wish to migrate to a sharded architecture will find it easier to move to Oracle Sharding than to alternative sharding solutions from various NoSQL vendors.

7.7 Integrating Oracle Fusion Middleware High Availability

Flexible and automated high availability solutions ensure that applications you deploy on Oracle Fusion Middleware meet the required availability to achieve your business goals.

This section contains the following topics:

7.7.1 Oracle WebLogic Server High Availability Architectures

Oracle WebLogic Server provides high availability and disaster recovery solutions for maximum protection against any kind of failure with flexible installation, deployment, and security options. These solutions are categorized into local high availability solutions that provide high availability in a single data center deployment, and disaster-recovery solutions, which are usually geographically distributed deployments that protect your applications from disasters such as floods or regional network outages.

At a high level, Oracle WebLogic Server local high availability architectures include several active-active and active-passive architectures. Although both types of solutions provide high availability, active-active solutions generally offer higher scalability and faster fail over, although they tend to be more expensive. With either the active-active or the active-passive category, multiple solutions exist that differ in ease of installation, cost, scalability, and security.

7.7.2 Redundant Architectures

Oracle WebLogic Server provides redundancy by offering support for multiple instances supporting the same workload. These redundant configurations provide increased availability either through a distributed workload, through a failover setup, or both.

From the entry point to an Oracle WebLogic Server system (content cache) to the back-end layer (data sources), all the tiers that are crossed by a request can be configured in a redundant manner with Oracle WebLogic Server. The configuration can be an active-active configuration using Oracle WebLogic Server Cluster or an active-passive configuration using Oracle WebLogic Server Cold Cluster Failover.

7.7.3 High Availability Services in Oracle Fusion Middleware

Oracle Fusion Middleware provides the following high availability services:

- **Process death detection and automatic restart**

  Oracle WebLogic Server Node Manager monitors the Managed Servers. If a Managed Server goes down, Node Manager tries to restart it for a configured number of times.

- **Clustering**

  Oracle Fusion Middleware uses WebLogic clustering capabilities, such as redundancy, failover, session state replication, cluster-wide JNDI services, Whole Server Migration, and cluster wide configuration.

- **State replication and routing**

  Oracle WebLogic Server can be configured for replicating the state of stateful applications.

- **Load balancing and failover**

  Oracle Fusion Middleware has a comprehensive feature set around load balancing and failover to leverage availability and scalability of Oracle RAC databases. All Oracle Fusion Middleware components have built-in protection against loss of service, data or transactions as a result of Oracle RAC instance unavailability due to planned or unplanned downtime.

- **Server migration**

  Oracle Fusion Middleware components leverage WebLogic Server capabilities to provide failover an automatic restart on a different cluster member.

- **Rolling patching**

  Oracle WebLogic Server allows for rolling patching where a minor maintenance patch can be applied to the product binaries in a rolling fashion without having to shut down the entire cluster.

- **Configuration management**

  Most of the Oracle Fusion Middleware component configuration can done at the cluster level. Oracle Fusion Middleware uses WebLogic Server's cluster wide-configuration capabilities for server configuration, such as data sources, EJBs, and JMS, as well as component application artifacts, and ADF and WebCenter custom applications.

- **Backup and recovery**

  Oracle Fusion Middleware backup and recovery is a simple solution based on file system copy for middle-tier components.

See Also:

High Availability Guide for information about ensuring high availability in Oracle Fusion Middleware

MAA Best Practices - Oracle Fusion Middleware

7.8 Integrating High Availability for All Applications

A highly available and resilient application requires that every component of the application tolerate failures and changes. A highly available application must analyze

every component that affects the application, including the network topology, application server, application flow and design, systems, and the database configuration and architecture. *Oracle Database High Availability Overview* focuses primarily on the database high availability solutions.

See Also:

http://www.oracle.com/goto/maa for high availability solutions and recommendations for Oracle Fusion Middleware, Oracle Fusion Applications, Oracle Enterprise Manager, and Oracle Applications Unlimited

8

Oracle Engineered Systems

Oracle offers engineered systems designed specifically to run Oracle Database in a high availability environment:

- Oracle Exadata Database Machine (page 8-1) - the highest performance and most available platform for running the Oracle Database
- Oracle SuperCluster (page 8-3) - the best multi-purpose engineering system for Oracle Database and applications
- Oracle Database Appliance (page 8-4) - the simplest, high availability Oracle Database appliance
- Zero Data Loss Recovery Appliance (page 8-5) - a cloud-scale engineered system and optimized backup and recovery appliance that is designed to protect all the Oracle Databases in your enterprise

8.1 Oracle Exadata Database Machine

The Oracle Exadata Database Machine is an engineered system, complete with preoptimized and preconfigured software, servers, and storage, configured to current best practices, that provides an optimal solution for all database workloads, ranging from scan-intensive data warehouse applications to highly concurrent OLTP applications. It combines Oracle Exadata Storage Server Software, Oracle Database software, and hardware components to deliver extreme performance in a highly available and highly secure environment. Along with Oracle's unique clustering and workload management capabilities, the Database Machine is also well-suited for consolidating multiple databases onto a single grid.

Oracle Exadata Database Machine is designed for high performance, scalability, and availability for OLTP, Data Warehouse applications, database consolidation, memory intensive workloads, and cloud computing. It is the only Engineered System focused on Oracle Database functionality and fully optimized for all database workloads using Exadata Smart Flash Technology, Exadata I/O Resource management, and Exadata smart offloading capabilities and features. For the best combination of database performance, scalability, and availability, use the Exadata MAA architecture.

The Oracle Exadata Database Machine hardware is fully redundant without any single points of failure. The Oracle software used on the Oracle Exadata Database Machine, used in conjunction with MAA best practices, yields a fault-tolerant system with the following benefits:

- Lowest brownout optimizations such as fast Exadata database or storage node detection and failover in less than 2 seconds versus 30-60 plus seconds with other customized configurations
- Continuous database availability across node and instance failures through the use of Oracle RAC

- Smart high availability and performance Quality of Service features such as I/O latency capping for read or write operations, detection, isolation, and removal of an underperforming or sick disk, and storage and network resource management to preserve highest application and database performance and availability
- Simplifies high availability management with features such as patching automation and optimizations, smart hardware alerts, and Exadata AWR reports
- Data protection and continuous database accessibility across disk and cell failures through the use of Oracle ASM and the Oracle Exadata Storage Server Software
- Prevents and automatically repairs data corruption using the Oracle ASM automatic repair mechanism, the built-in corruption checks within the Exadata storage, and the recommended database default settings (`db_block_checksum`, `db_lost_write_protect`)
- Inspects and repairs hard disks with damaged or worn out disk sectors (cluster of storage) or other physical or logical defects periodically when there are idle resources with Exadata Automatic Hard Disk Scrub and Repair
- Provides redundant and fault tolerant networking, cabling, and server interconnectivity
- Provides the ability to quickly reestablish database service if the Oracle Exadata Database Machine--or the data center the machine resides in--is damaged, through the use of Oracle Active Data Guard or Oracle GoldenGate

For planned maintenance, Oracle Exadata Database Machine provides the following benefits:

- Supports Oracle ASM, Oracle Clusterware, and Oracle RAC rolling upgrade and software changes
- Supports Oracle Exadata Storage Server Software rolling upgrades for patches
- Allows application and system changes with Data Guard and Oracle GoldenGate
- Supports all of the online maintenance capabilities that are available in the Oracle Database
- Provides tools and Oracle Enterprise Manager 13*c* support to automate patching Grid Infrastructure and Oracle Database software, database server operating system and firmware (`dbnodeupdate.sh` only), and all Exadata servers and InfiniBand switches (`patchmgr` only) in a rolling manner

With Oracle Exadata Database Machine, your Oracle Database High Availability architecture choices are simplified and Exadata is applicable for all MAA reference architectures.

Oracle Exadata Database Machine is the recommended platform for the MAA tiers and for database consolidation.

The recommended Exadata MAA Architecture consists of three elements: 1) a production Exadata system (primary), 2) a standby Exadata system, and 3) an Exadata test or development system.

The second element can be an Active Standby Exadata system that is a replica of the primary that contains all the benefits of any Exadata Database Machine and the benefits described in Oracle Data Guard (page 3-2).

The third element can be a development and test Exadata system that is independent of the primary and standby Exadata systems, following the best practices described in Establish Test Practices and Environment (page 6-2).

It is recommended that you run Oracle's Exadata Health Check (`exachk`) monthly because it provides the most comprehensive configuration checks for Exadata software, network, and hardware components, and it reports any variance from MAA best practices..

The MAA configuration best practices will continue to be integrated and incorporated during the initial installation and deployment of the Exadata Database Machine and Exadata Cell.

See Also:

- High Availability Architectures (page 7-1) for details about the MAA reference architectures
- MAA white paper High Availability Best Practices for Database Consolidation: The Foundation for Database-as-a-Service for information about using Exadata as your DBaaS platform
- Operational Prerequisites to Maximizing Availability (page 6-1), and the MAA white paper MAA Best Practices for Oracle Exadata Database Machine for other post-deployment and operational best practices specific to Exadata
- OTN Exadata MAA web site for other Exadata MAA best practices
- https://www.oracle.com/engineered-systems/exadata/index.html for data sheets, videos, and other resources
- Oracle Exadata Database Machine documentation
- My Oracle Support Note 1070954.1 for information about `exachk`

8.2 Oracle SuperCluster

The Oracle MiniCluster S7-2, Oracle SuperCluster M7 and Oracle Supercluster M6-32 are multi-purpose engineered systems for consolidating a wide range of mission critical applications, databases and cloud services.

Oracle SuperCluster is ideally targeted to existing SPARC and IBM database customers or customers preferring a multi-purpose engineered system that hosts a combination of various database releases (Oracle 10*g* and later, compared to only Oracle 11*g* and later for Exadata) and application servers. Oracle SuperCluster has additional shared storage with ZFS storage cluster that can be used for non-database files and has virtualization support through Oracle VM Server for SPARC and Oracle Solaris Zones. Oracle Exadata Database Machine is still the recommended database machine; however Oracle SuperCluster provides other functionality for your application tier if you require a multi-purpose solution.

Oracle SuperCluster is recommended for the MAA service level tier architectures when a multi-purpose database and application processing system is required. Oracle SuperCluster M6-32 provides the largest in-memory footprint of any Oracle engineered system. Additionally, if non-database files need to be protected in the case of full stack failover or switchover, you can use a combination of Data Guard with ZFS

storage remote mirroring. It is also recommended to run Oracle's Exadata health check (`exachk`) monthly because it provides the most comprehensive configuration checks for Exadata software, network, and hardware components, and it reports any variance from MAA best practices.

See Also:

Oracle Engineered Systems Documentation for Oracle SuperCluster documentation

Oracle SuperCluster for data sheets, videos, and other resources

See My Oracle Support Note 1070954.1 for information about `exachk`.

8.3 Oracle Database Appliance

Oracle Database Appliance is an engineered system consisting of hardware and software that saves customers time and money by simplifying deployment, maintenance, and support of high availability database solutions. Built with Oracle Real Applications Clusters (Oracle RAC) and Oracle Automatic Storage Management best practices, it offers customers a fully integrated system of software, servers, storage and networking in a single box delivering high availability database services for a wide range of home grown and packaged OLTP and Data Warehousing workloads. It comes as a 4RU (rack unit) server appliance that consists of two server nodes and 18TB raw storage and 800GB of SSD capacity running on Oracle Linux with an optional expansion shelf to double the storage to 36TB raw and 1.6TB of SSD.

Building highly available systems can be difficult and complex and may require advanced integration skills that many organizations don't have and be risky and error-prone with no vendor accountability. The Oracle Database Appliance is simple, reliable, and affordable.

- **Simple**

  To deploy and use the Oracle Database Appliance, simply unpack it, plug in the power cord, plug in the network cables, and run the Oracle Appliance Manager installation to create a clustered, highly available database. The Oracle Database Appliance and its specially engineered software enables "one button" patching for all the elements of the software stack - firmware, operating system, clusterware, storage manager, and database software.

- **Reliable**

  The Oracle Database Appliance is built on the Oracle software stack which is completely integrated along with the storage that include 600 GB SAS Hard Disk Drives between the two server nodes that can be triple-mirrored or double mirrored to provide highly available shared storage. This appliance also contains SAS Solid State Drives for redo logs, triple-mirrored to protect the Oracle database in case of instance failure. The appliance manager in conjunction with Oracle Automatic Storage Management (ASM) automatically configures, manages, and monitors the disk for performance and availability. The Oracle Appliance Manager provides alerts on performance and availability events as well as automatically configures replacement drives in case of a hard disk failure.

- **Affordable**

The Oracle Database Appliance can minimize licensing costs while providing a capacity-on-demand platform for deploying databases by initially licensing as few as 4 cores for the bare metal option – additional cores can be activated at any time. Virtualization provides additional licensing flexibility by providing isolation between databases and other workloads by leveraging Oracle VM hard partitioning.

The Oracle Database Appliance is an ideal **database appliance** for customers who value simplicity and who seek to avoid the complexity, costs, and risks in deploying a highly available database solution. The Oracle Database Application is also ideal for database customers who do not require Exadata Database Machine's additional performance and scalability with its Exadata software capabilities and additional availability (rolling patch upgrades as opposed to "one button patching for all the elements of the software stack). Customers can now benefit from high availability (HA) database solutions without having special skills or HA expertise.

Oracle Database Appliance is an option for the MAA service level tier architectures when a standard platform with high consolidation density for smaller environments is required. The recommended Oracle Database Appliance MAA Architecture consists of three elements: 1) a production Oracle Database Appliance system (primary), 2) a standby Oracle Database Appliance system, and 3) an Oracle Database Appliance test or development system. It is recommended that you run Oracle health check (`orachk`) for its comprehensive configuration checks on operating system, Grid Infrastructure and database settings.

See Also:

Oracle Engineered Systems Documentation for Oracle database Appliance documentation

Oracle Database Appliance for white papers, videos, and other resources

My Oracle Support Note 1268927.2 for information about `orachk`.

8.4 Zero Data Loss Recovery Appliance

The cloud-scale Zero Data Loss Recovery Appliance, commonly known as Recovery Appliance, is an engineered system designed to dramatically reduce data loss and backup overhead for all Oracle databases in the enterprise. Integrated with Recovery Manager (RMAN), the Recovery Appliance enables a centralized, incremental-forever backup strategy for large numbers of databases, using cloud-scale, fault-tolerant hardware and storage. The Recovery Appliance continuously validates backups for recoverability. The Recovery Appliance can augment and improve any MAA reference architecture and is part of Oracle MAA's recommended backup and recovery solution, especially due to its inherent database backup validation and protection.

Most database backup and restore processing is performed by the centralized Recovery Appliance, making storage utilization, performance, and manageability of backups more efficient. The Recovery Appliance stores and manages backups of multiple databases in a unified disk pool, using an RMAN incremental-forever strategy. The Recovery Appliance continually compresses, deduplicates, and validates backups at the database block level, while creating virtual full backups on demand.

A virtual full backup is a complete database image as of one distinct point in time, maintained efficiently through Recovery Appliance indexing of incremental backups from protected databases. A virtual full backup can correspond to any incremental backup that was received.

Administrators use Oracle Enterprise Manager Cloud Control (Cloud Control) to manage and monitor the environment. Cloud Control provides a "single pane of glass" view of the entire backup lifecycle for each database, whether backups reside on disk, tape, or another Recovery Appliance.

Recovery Appliance provides the following benefits:

- Elimination of Data Loss
- Minimal Backup Overhead
- Improved End-to-End Data Protection Visibility
- Cloud-Scale Protection

See Also:

Zero Data Loss Recovery Appliance Administrator's Guide

Zero Data Loss Recovery Appliance Protected Databases Configuration Guide

MAA Best Practices - Zero Data Loss Recovery Appliance for MAA white papers

9

Optimizing Return on Investment

The Oracle MAA reference architectures provide for high availability and data protection while also achieving high return on investment (ROI) on all systems and software that are deployed. This is accomplished by efficiently managing workloads in consolidated environments and actively using all systems for productive purposes at all times. This eliminates costly excess capacity and expensive over-provisioning of systems that remain idle until after an outage occurs.

While Oracle MAA reference architectures may be deployed on any commodity platform, there are substantial benefits to their deployment on Oracle Engineered Systems. Oracle integrated hardware and software systems reduce total life cycle costs by using standard, high performance platforms that achieve economies of scale for consolidated environments and DBaaS along multiple dimensions: performance, reliability, manageability, and support. Along the way, Oracle Engineered Systems free your IT staff from mundane systems integration tasks that distract them from focusing on other tasks that deliver higher ROI for your enterprise.

Collectively, these capabilities enable enterprises to achieve their desired goals when embarking upon strategic initiatives for database consolidation and deployment of Database as a Service (DBaaS) in private or public clouds.

This chapter contains the following topics:

9.1 High ROI Using Grid Computing

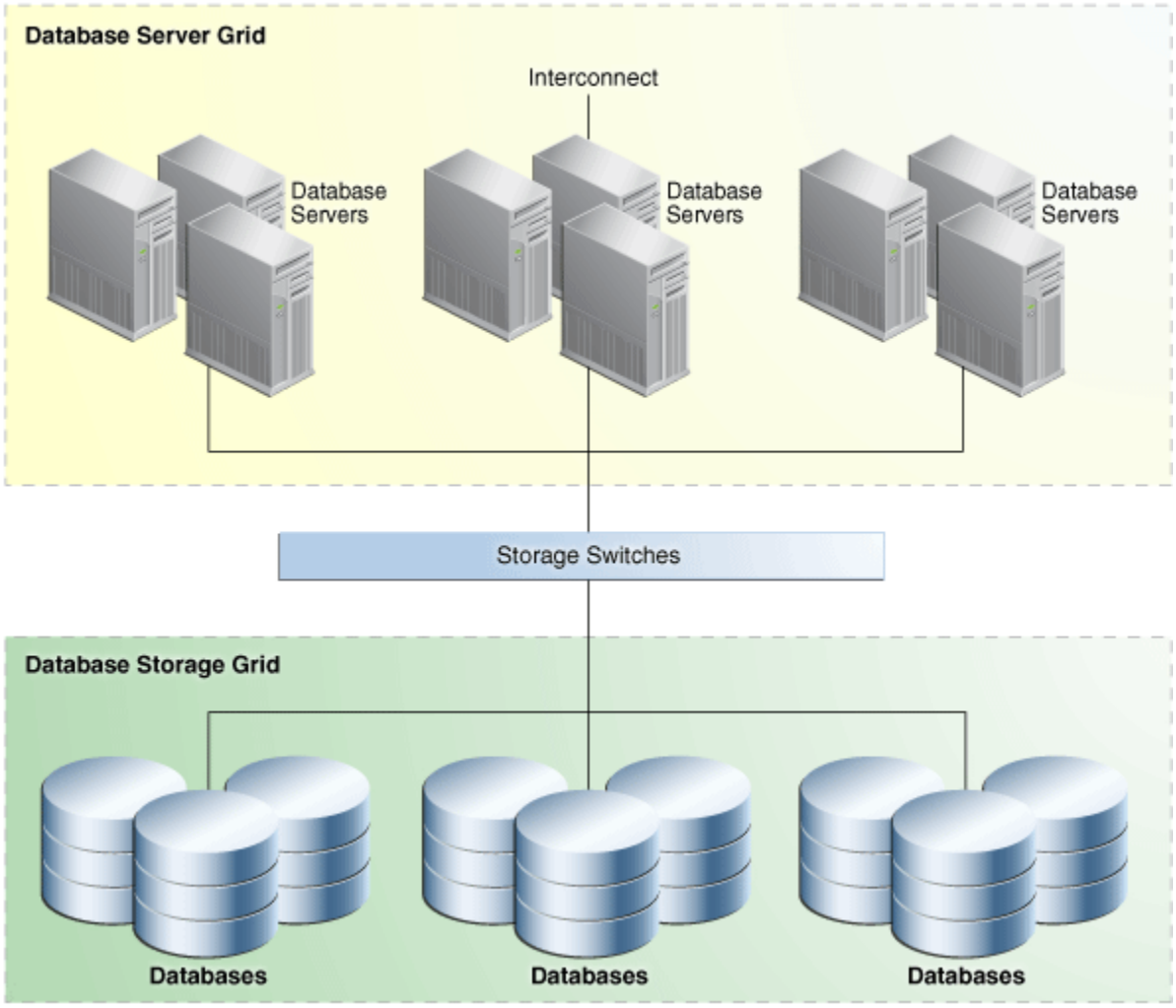
Grid computing is a computing architecture that effectively pools large numbers of servers and storage into a flexible, on-demand computing resource for all enterprise computing needs.

The Oracle Database achieves the cost advantages of Grid enterprise computing without sacrificing performance, scalability, security, manageability, functionality, or system availability.

- A Database Server Grid is a collection of commodity servers connected to run one or more databases.
- A Database Storage Grid is a collection of low-cost modular storage arrays accessed by the servers in the Database Server Grid.

The same grid computing concept applies to primary as well as standby database environments. Figure 9-1 (page 9-2) illustrates the Database Server Grid and Database Storage Grid in a grid enterprise computing environment.

Figure 9-1 Grid Computing Environment



9.1.1 Database Server Grid

Oracle Real Application Clusters is the technology that enables a Database Server Grid. Oracle RAC is an active-active solution that enables read-write workload for an Oracle database to be automatically load-balanced across all nodes in the cluster. Oracle RAC provides the flexibility to dynamically provision resources and services, and to add or subtract systems from the grid as capacity demands change. In addition, Oracle RAC provides protection from system failures by automatically transitioning clients and redistributing the processing of the failed node to surviving nodes running the same Oracle RAC database.

9.1.2 Database Storage Grid

The Oracle Storage Grid is implemented using Oracle Automatic Storage Management (Oracle ASM) with any third party storage or with storage optimized for the Oracle

Database: Oracle Exadata Storage Server, Oracle ZFS storage or Pillar SAN Storage Systems.

A database administrator can use the Oracle ASM interface to specify the disks in the Database Storage Grid that Oracle ASM will manage. Oracle ASM partitions the disk space and evenly distributes the data storage throughout the entire storage array. Additionally, Oracle ASM automatically rebalances the location of data to eliminate hot spots, and redistributes data as disks or storage arrays are added or removed from the Database Storage Grid.

I/O Resource Management (available only for Exadata storage) is used to manage I/O performance and meet service-level requirements in consolidated environments. The resource manager allows you manage the storage grid and prioritize applications within the database or between databases.

9.2 High ROI Using Active Standby Databases

Data Guard standby databases provide data protection, availability, and disaster recovery regardless of the cause or scope of an outage. Outages can range anywhere from data corruption that can affect an individual database, to natural disasters that impact a large geographic area.

Advanced Data Guard capabilities deliver high ROI by enabling standby databases to be used for productive purposes, such as read-only queries and reporting, while operating in the standby role. Rather than allowing standby databases to remain idle, they can be used for workloads that would otherwise require the purchase of additional capacity to support. This effectively reduces the cost of deploying a Data Guard standby to achieve optimal data protection and availability. Oracle Active Data Guard provides advanced data protection for every Oracle Database by supporting all data types, workloads, and applications - there are no limitations or restrictions. Any application that can utilize a database that is open read-only for ad-hoc queries, reporting, data extracts, etc, can also utilize an Oracle Active Data Guard standby database to offload workload from the primary production system

The following sections describe the Data Guard scenarios that provide high business utilization and a maximum return in investment:

- Oracle Active Data Guard Option for Physical Standby Databases (page 9-3)
- Oracle Active Data Guard Reader Farms (page 9-4)
- Data Guard and the Cloud (Data Protection as a Service) (page 9-6)

9.2.1 Oracle Active Data Guard Option for Physical Standby Databases

Data Guard Redo Apply (physical standby database) is an attractive disaster recovery solution due to its relative simplicity, high performance, and superior level of data protection. The Oracle Active Data Guard option (available with Oracle Database 11*g* Release 1 (11.1) and later releases) enables a physical standby database to be opened for read-only access while Redo Apply is active.

Note:

Oracle Active Data Guard is referred to as real-time query in the Data Guard documentation.

Oracle Active Data Guard includes the following capabilities:

- Oracle Database In-Memory queries and analytics (new in Oracle Database 12*c* Release 2)
- Offload of reporting and ad-hoc queries that include DML to global temporary tables
- Support for unique global or session sequences on an active standby
- Real-time cascade for the efficient servicing of multiple remote destinations
- The ability to extend zero data loss protection to a remote standby database without impacting primary database performance - Active Data Guard Far Sync
- The ability to perform off-host redo transport compression
- Automation that simplifies reducing planned downtime using database rolling upgrades

9.2.2 Oracle Active Data Guard Reader Farms

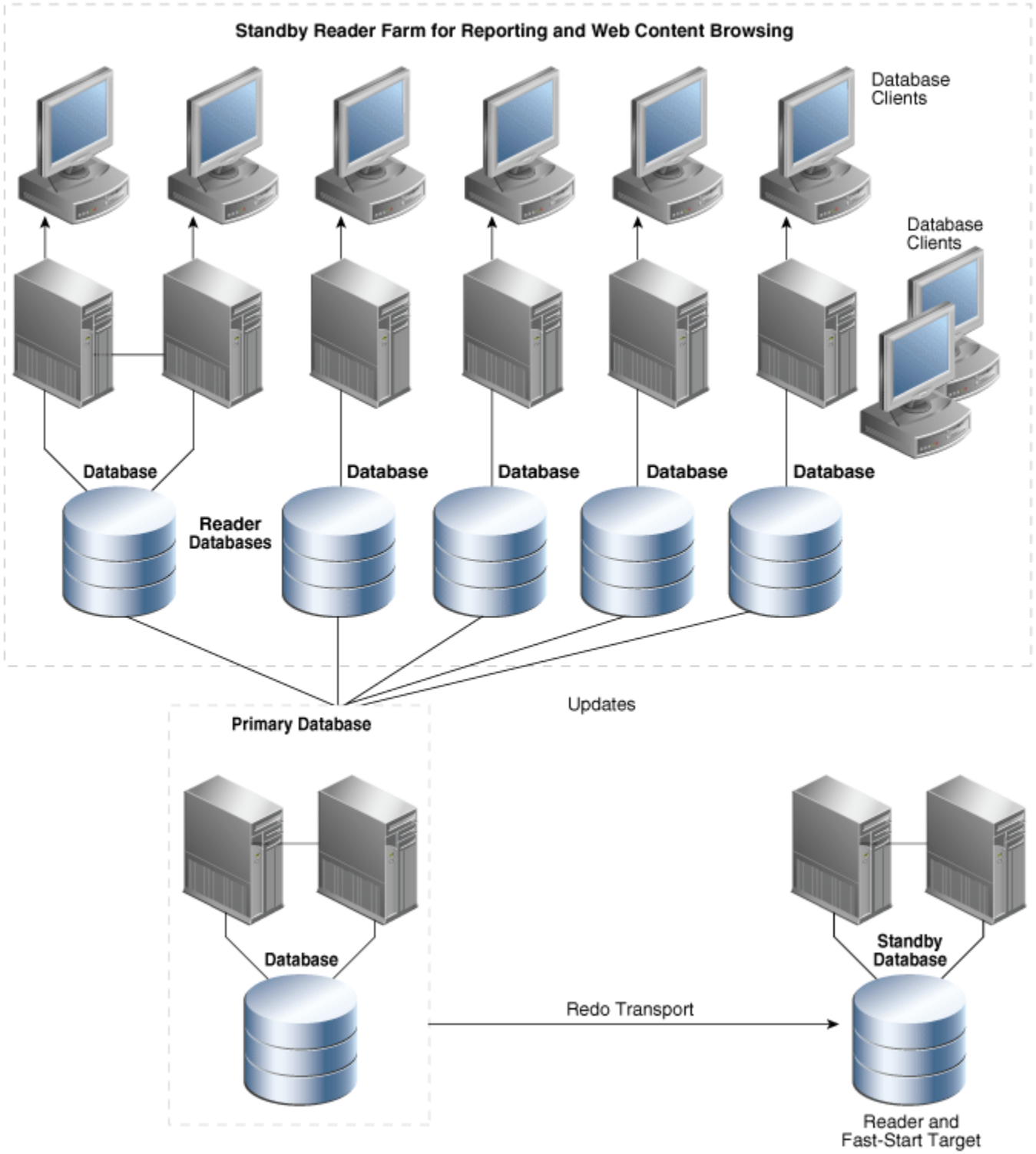
Active Data Guard can be used to deploy multiple active standby databases to easily scale read performance, also referred to as a reader-farm. An example of such a configuration is provided in Figure 9-2 (page 9-5), complete with the use of Data Guard fast-start failover to automatically fail over should the primary database fail. Note that all standby databases in the reader farm automatically recognize the new primary database after a failover occurs.

A reader farm enables an application to scale read performance of the most demanding web applications beyond what the underlying system and storage architecture can support. This provides a relatively low-cost method of scaling out simply by adding more Oracle Active Data Guard standby databases.

An Oracle Active Data Guard reader farm provides the following benefits:

- Simplicity
- Fault isolation
- High performance with physical standby databases and Redo Apply
- Seamless support for all DDL and data types using Redo Apply
- All reader databases are kept up-to-date with changes made to the primary database
- Automatic (Fast-Start Failover) or manual failover
- Zero or near-zero data loss potential
- Management as a unified configuration through Grid Control
- Scale-out using single writer database and n reader databases
- Rolling upgrade capabilities
- Integrated client failover to production database or other standby databases using Global Data Services

Figure 9-2 Standby Database Reader Farms



If a fast-start failover is triggered in the Data Guard configuration in Figure 9-2 (page 9-5), then:

- Automatic failover occurs to the designated standby database
- All standby databases accept data from the new primary database

- You can perform a switchover at a convenient time in the future to return all databases to their original roles

9.2.3 Data Guard and the Cloud (Data Protection as a Service)

Oracle Active Data Guard can provide Data Protection as a Service for public, private and hybrid clouds. Oracle Enterprise Manager Cloud Control can be used for self-service provisioning of standby databases. Oracle Engineered Systems provide a scalable platform for providing cloud services. Standby databases deployed on the cloud can also support cloning for development and test purposes. For example:

- A primary database on premises can transmit redo directly or indirectly by using an Active Data Guard Far Sync instance to a standby database deployed on a public cloud. The combination of the on premises primary and public cloud standby is referred to as a hybrid cloud deployment.
- A standby hub can be used to provide data protection for a private cloud. The standby hub is comprised of a database server grid that centrally hosts standby databases for primary databases distributed across many different data centers.
- A database deployed on a public cloud may also have a standby database deployed in a different availability zone provided by the public cloud vendor to provide increased data availability and disaster protection.
- A standby database deployed on the public cloud may be cloned for development and test purposes or may be used directly as a test system using Data Guard snapshot standby.

9.3 High ROI Using Oracle Database Consolidation

Enterprises are under intense pressure to do more with less, to reduce risk, and increase agility. The aggressive consolidation of information technology (IT) infrastructure and deployment of Database as a Service (DBaaS) on public or private clouds is a strategy that many enterprises are pursuing to accomplish these objectives.

Several key elements are needed to realize the full potential for cost reduction through database consolidation and DBaaS. High consolidation density and management simplicity are required to achieve maximum reduction in hardware and administrative costs. These attributes must then be combined with intelligent software infrastructure capable of achieving service level agreements (SLAs) for availability, performance, and data protection.

9.3.1 Multitenant Architecture

Oracle Multitenant fundamentally changes Oracle Database architecture by introducing the concepts of multitenant container databases (CDB) and pluggable databases (PDB). Existing databases can be easily converted to a PDB. Consolidation is achieved by 'plugging in' multiple PDBs into a single CDB. Oracle Database 12*c* with Oracle Multitenant is engineered to deliver the most efficient platform in every aspect for database consolidation.

A CDB has a single set of background processes and shared memory area (SGA) that is used by all PDBs. This architecture requires less CPU and memory compared to traditional approaches of consolidating multiple independent databases onto a single physical machine, or multiple virtual machines (VMs), or an Oracle RAC cluster. While a CDB can be deployed in either physical or virtual environments, it achieves the highest management and performance efficiency for the database tier when

deployed on a physical machine. The CDB itself becomes the virtualization technology for the database tier, eliminating the overhead of multiple VMs and guest operating systems.

Oracle Multitenant also provides a high degree of isolation. A PDB can be easily unplugged from one CDB and plugged into another to allow database administrators the option of performing maintenance on an individual PDB if required. An individual PDB can be provisioned, patched, cloned, consolidated, restored, or moved without impacting other PDBs in the same CDB.

Oracle Multitenant is unique in accomplishing the positive attributes of alternative consolidation methods while avoiding each of their drawbacks. Oracle Multitenant achieves:

- The simplicity and flexibility of VMs, without the limits to consolidation density, performance, or increased management complexity
- The high consolidation density of schema consolidation, without the implementation complexity, limited flexibility, and limited isolation
- The HA, scalability, and automated workload management of simple database consolidation using Oracle RAC with Oracle Database 12*c*, without the limitations in consolidation density or management complexity of a separate database (each with its own operational overhead) for each application

Oracle Multitenant seamlessly integrates with the HA and data protection capabilities of Oracle Database. This integration combined with Oracle Maximum Availability Architecture (MAA) best practices provides an evolutionary upgrade path to a revolutionary technology for database consolidation.

See Also:

MAA white paper "Oracle Maximum Availability Architecture Best Practices for Database Consolidation" to leverage the maximum benefits of Oracle Multitenant for database consolidation

Oracle Database Administrator's Guide for information about creating and administering pluggable databases.

9.3.2 Oracle Virtualization

Data centers today use virtualization techniques to make abstraction of the physical hardware, create large aggregated pools of logical resources consisting of CPUs, memory, disks, file storage, applications, networking, and offer those resources to users or customers in the form of agile, scalable, consolidated virtual machines. Even though the technology and use cases have evolved, the core meaning of virtualization remains the same: to enable a computing environment to run multiple independent systems at the same time with the main intent of saving people and hardware resources.

Oracle has three main virtualization technologies:

- Oracle VM for X86 and Oracle VM Manager are an enterprise-class server virtualization solution. Oracle VM Server for x86 is the most scalable x86 server virtualization solution in the market today, and it has been tested to handle mission critical enterprise workloads with support for up to 160 physical CPUs and 2 TB of memory. For virtual machines, Oracle VM 3 can support up to 128 virtual CPUs

and 1TB memory per guest VM. Oracle VM supports industry standard x86 operating systems and servers from Oracle and other leading vendors, and it supports a broad range of network and storage devices, making it easy to integrate into your environment. Oracle VM Manager provides an easy-use-centralized management environment for configuring and operating your server, network, and storage infrastructure from a browser based interface (no Java client required), and it is accessible from just about anywhere.

- Oracle VM Server for SPARC provides highly efficient, enterprise-class virtualization capabilities for Oracle's SPARC T-Series servers. Using the Oracle VM Server for SPARC software, you can create up to 128 virtual servers, called logical domains, on a single system. This kind of configuration enables you to take advantage of the massive thread scale offered by SPARC T-Series servers and the Oracle Solaris OS.
- Oracle Solaris Zones software partitioning technology, which provides a means of virtualizing operating system services to create an isolated environment for running applications. This isolation prevents processes that are running in one zone from monitoring or affecting processes running in other zones. Zones can be used on any machine that is running the Oracle Solaris 10 or a later Oracle Solaris release. The upper limit for the number of zones on a system is 8192.
- Note: The Oracle Virtual Compute Appliance is the Oracle Engineered System specifically designed to radically simplify the way customers install, deploy, and manage virtual infrastructures for any Linux, Oracle Solaris, or Microsoft Windows application.

Oracle virtualization can be used in conjunction with HA features and HA architectures to reap the benefits of both target goals. Here are some of the HA benefits when integrating Oracle virtualization with HA architecture and features.

- Auto restart of VMs in the event of a failure making applications HA
- Oracle Real Application Clusters ensure business availability at the application layer and is integrated with Oracle VM to ensure business availability on the server as well as application data in a single or multiple geographic locations
- Generally any Oracle high availability feature, such as RMAN, flashback technologies, Data Guard, and Oracle GoldenGate, that works natively in non-virtualized environments will work seamlessly in a virtualized environment.
- Oracle VM accelerates the delivery of services to meet changing business need. This allows online growing of capacity

---

See Also:

- The Oracle VM website on OTN at `http://www.oracle.com/virtualization`
- *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*, Part II, "Oracle Solaris Zones" at `http://docs.oracle.com/cd/E26502_01/html/E29024/zones.intro-2.html#scrolltoc`

9.4 High ROI Using Oracle Global Data Services

Oracle Global Data Services (GDS) is a complete automated workload management solution for replicated databases that use Oracle Active Data Guard or Oracle GoldenGate. GDS increases ROI by enhancing system utilization, performance, scalability, and availability for application workloads running on replicated databases. GDS provides the following capabilities for a set of replicated databases:

- Region-based workload routing
- Connect-time load balancing
- Run-time load balancing advisory for Oracle integrated clients
- Inter-database service failover
- Replication lag based workload routing for Oracle Active Data Guard
- Role-based global services for Oracle Active Data Guard
- Centralized workload management framework

Replicated databases within a GDS configuration can be globally distributed or located within the same data center. Clients can securely connect to the GDS configuration by simply specifying a service name, without needing to know anything about the components and topology of the GDS configuration, enabling a highly flexible private cloud deployment for the enterprise.

Geographically dispersed data centers, whether regional or global, can now be effectively utilized within a uniform framework based on business, throughput, and localized demands, without affecting run-time applications.

See Also:

Oracle Database Global Data Services Concepts and Administration Guide

Glossary

business impact analysis

An impact analysis that categorizes the business processes based on the severity of the impact of IT-related outages.

clusterwide failure

The whole cluster hosting the Oracle RAC database is unavailable or fails. This includes failures of nodes in the cluster, and any other components that result in the cluster being unavailable and the Oracle database and instances on the site being unavailable.

computer failure

An outage that occurs when the system running the database becomes unavailable because it has crashed or is no longer accessible.

cost of downtime

A complete business impact analysis provides the insight needed to quantify the cost of unplanned and planned downtime. Understanding this cost is essential because it helps prioritize your high availability investment and directly influences the high availability technologies that you choose to minimize the downtime risk.

data corruption

A corrupt block is a block that has been changed so that it differs from what Oracle Database expects to find. Block corruptions fall under two categories: physical and logical block corruptions.

See also physical block corruption and logical block corruption.

hang or slow down

Hang or slow down occurs when the database or the application cannot process transactions because of a resource or lock contention. Perceived hang can be caused by lack of system resources.

human error

An outage that occurs when unintentional or malicious actions are committed that cause data in the database to become logically corrupt or unusable. The service level impact of a human error outage can vary significantly depending on the amount and critical nature of the affected data.

logical block corruption

The contents of the block are logically inconsistent. Examples of logical corruption include corruption of a row piece or index entry.

logical unit numbers (LUNs)

Three-bit identifiers used on a SCSI bus to distinguish between up to eight devices (logical units) with the same SCSI ID.

lost write

A lost write is another form of **data corruption** that can occur when an I/O subsystem acknowledges the completion of the block write, while in fact the write I/O did not occur in the persistent storage. No error is reported by the I/O subsystem back to Oracle Database.

MAA environment

An architecture that provides the most comprehensive set of solutions for both unplanned and because it inherits the capabilities and advantages of both Oracle Database 11*g* with Oracle RAC and Oracle Database 11*g* with Data Guard.

The MAA environment consists of a site containing an Oracle RAC primary database and a second site containing a standard cluster that hosts both logical and physical standby databases, or at least one physical or logical standby database.

manageability goal

More subjective than either the RPO or the RTO, the manageability goal results from an objective evaluation of the skill sets and management resources available in an organization, and the degree to which the organization can successfully manage all elements of a high availability architecture. Understanding manageability goals helps organizations differentiate between what is possible and what is practical to implement.

network failure

A network failure occurs when a network device stops or reduces network traffic and communication from your application to database, database to storage, or any system to system that is critical to your application service processing.

network server processes

The Data Guard network server processes, also referred to as LNS*n* processes, on the primary database perform a network send to the RFS process on the standby database. There is one network server process for each destination.

Oracle Active Data Guard option

A physical standby database can be open for read-only access while Redo Apply is active if a license for the Oracle Active Data Guard option has been purchased. This capability, known as Oracle Active Data Guard, also provides the ability to have block-change tracking on the standby database, thus allowing incremental backups to be performed on the standby.

Note: The Oracle Active Data Guard option may also be referred to as "real-time query" in other documentation.

physical block corruption

The database does not recognize the block at all: the checksum is invalid, the block contains all zeros, or the header and footer of the block do not match. A physical corruption is also called a media corruption.

recovery point objective (RPO)

The maximum amount of data an IT-based business process may lose before causing harm to the organization. RPO indicates the data-loss tolerance of a business process or an organization in general. This data loss is often measured in terms of time, for example, five hours or two days worth of data loss.

recovery time objective (RTO)

The maximum amount of time that an IT-based business process can be down before the organization suffers significant material losses. RTO indicates the downtime tolerance of a business process or an organization in general.

return on investment (ROI)

Return on Investment (or Rate of return) is used to evaluate the efficiency of an investment in finance and economics.

site failure

An outage that occurs when an event causes all or a significant portion of an application to stop processing or slow to an unusable service level. A site failure may affect all processing at a data center, or a subset of applications supported by a data center.

storage failure

An outage that occurs when the storage holding some or all of the database contents becomes unavailable because it has shut down or is no longer accessible.

total cost of ownership (TCO)

A financial estimate designed to help consumers and enterprise managers assess direct and indirect costs. It is used in many industries and is a form of full cost accounting.

transient logical standby database

A transient logical standby database allows you to reuse your current physical standby database by temporarily converting it into a logical standby on which to perform a rolling database upgrade, incurring minimal downtime.

Index

Symbols

A

B

C

D

E

F

G

H

P

Q

R

S

T

U

V

W

Z