
Oracle Data Guard

Caleb Small
Caleb@Caleb.com

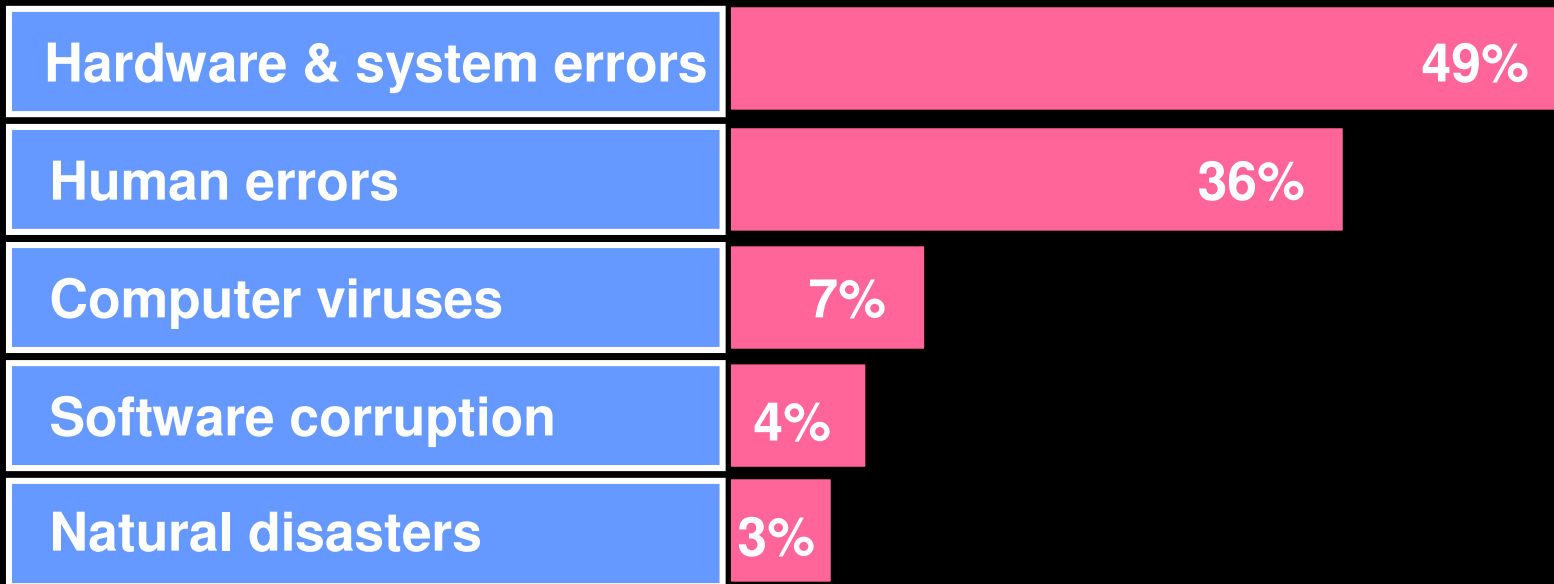
Outline

- Overview
- Planning a DataGuard Installation
- Architecture
- Setting up DataGuard
- Managing & Monitoring
- Crash & Burn Demo

Overview

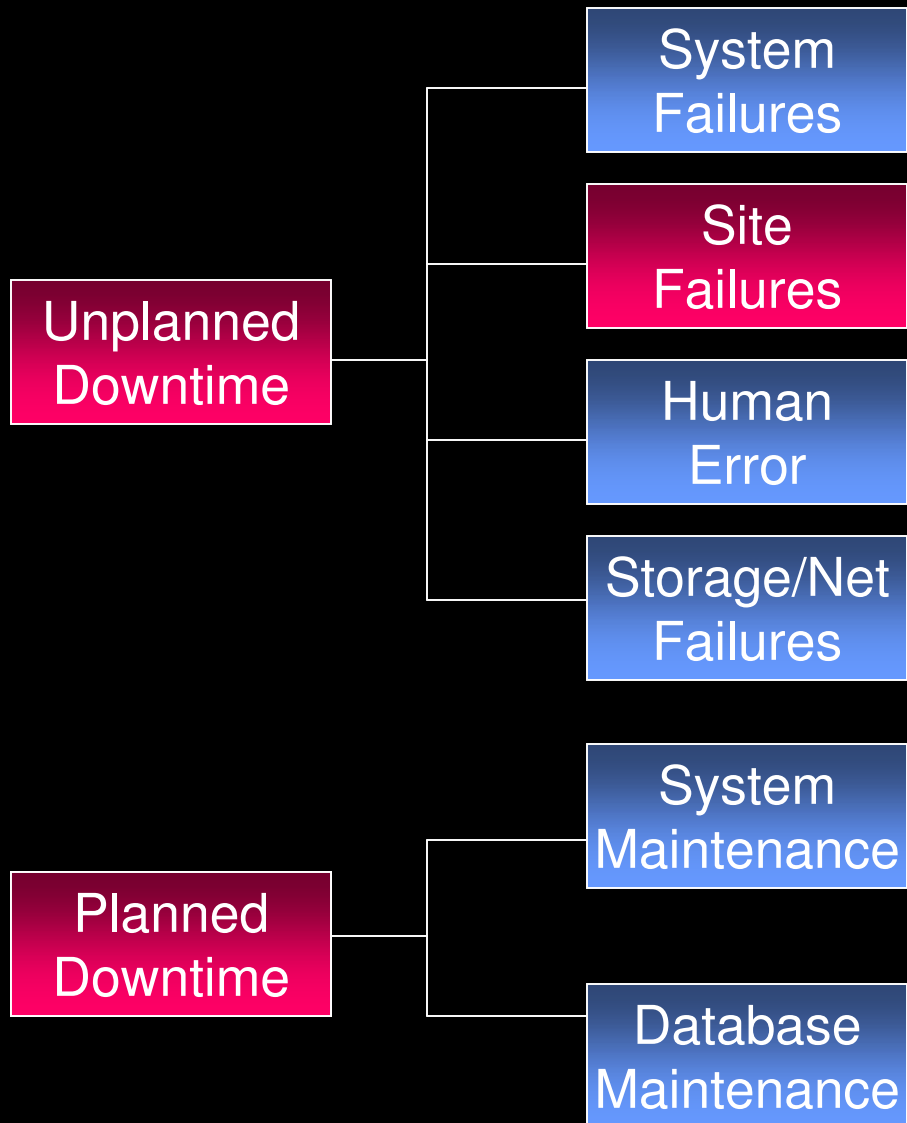
Data Guard Concepts

Causes of Data Loss



Source: Disaster Recovery Journal

Fault Tolerance



Real Application Clusters

Continuous Availability for all Applications

Data Guard

Guaranteed Zero Data Loss

Flashback

Guaranteed Zero Data Loss

ASM Mirroring

Storage Failure Protection

Dynamic Reconfiguration

Capacity on Demand without Interruption

Data Guard

Guaranteed Zero Data Loss

Online Redefinition

Adapt to Change Online

A Complete High-Availability Solution

- Recovery Manager (RMAN)
- Flashback Database / Drop / Query / Table
- Real Application Clusters (RAC)
- Data Guard
- Data Pump
- Grid Control

Data Guard is one part of a complete HA solution

What is Data Guard?

- Oracle's disaster recovery solution
- Automates the creation and maintenance of one or more transitionally consistent copies (standby) of the production (or primary) database
- If the primary database becomes unavailable (disasters, maintenance), a standby database can be activated and assumes the primary role
- Can cascade changes from standby-to-standby
- Built by the same team that wrote RMAN, Streams, and Change Data Capture

Why Data Guard?

- High Availability (HA) Disaster Preparation
 - Replication between geographically separated data centers, or servers within a data center.
- Included in the Enterprise Edition license
- Oracle block aware
- Rolling hardware, operating system, and database upgrades, patches, and system maintenance
- Remove backup process overhead from primary production systems
- Production mirror for reporting

What is a Data Guard Standby Database

- Production database copy for disaster protection
- Kept current with archived redo logs
- If a disaster destroys the production DB then standby can be activated as new production DB
- Can be maintain in one of the following modes:
 - For physical standby databases
 - Redo Apply
 - Open read-only mode
 - For logical standby databases
 - SQL Apply
 - Open read/write mode

Planning a Data Guard Installation

- Standby DB type (logical vs physical)
- Database protection mode
- Network configuration/capacity
- Standby server capacity
- Failover method
- Archivelog management
- Secondary reporting/backup

Standby Database Types

1. Physical standby database
2. Logical standby database
3. Snapshot standby database (11g)

Combinations are possible

Standby Database Types

Physical standby database

- Block-for-block identical with the primary database
- Synchronized with the primary database by application of redo data
- Not open while redo is being applied (10g)
- Can be opened temporarily for read-only (10g)
- In 11g, can be left open read-only for reporting
- ***MUST NOT*** be opened read-write!

Standby Database Types

Logical standby database

- Shares the same schema definition
- Synchronized with the primary database by transforming redo into SQL statements
- Database is open while redo-SQL is being applied
- May contain additional indexes and materialized views to support reporting
- Does NOT support all datatypes (eg. rowid, nested table, varray, object type REF, XMLtype)

Standby Database Types

Snapshot standby database

- Variant of Physical standby
- Open read-write and make changes
- Then loose changes and re-instate as physical standby

Standby Database Types

Combination of Logical and/or Physical

- Star and cascading configurations are possible
- Increase protection with multiple standby DBs
- Increase availability by adding logical (10g) standbys (eg. for reporting)
- Offload overhead of backup to physical standby

Database Protection Modes

1. Maximum Protection - safe
2. Maximum Availability - compromise
3. Maximum Performance – fast
- optional redo apply lag

Database Protection Modes

Maximum Protection

- No Data Loss and No data divergence
- Redo data *must* be written to both the local and standby online redo log on at least one standby
- Primary DB will shutdown when unable to access Standby
- Performance of Primary DB is limited by network and redo apply at Standby

Database Protection Modes

Maximum Availability

- Same as Maximum Protection except the protection level is auto lowered when Standby is unavailable
- Will catch up at next archive log switch
- Data loss is possible

Database Protection Modes

Maximum Performance (default)

- Transactions commit as soon as redo is written to the local online redo log.
- Redo is transmitted synchronously (by LGWR) or asynchronously (by ARCH) to standby
- If network or standby DB fails, primary DB continues to commit transactions and archive logs stack up
- When network / standby DB problems are fixed, gaps are resolved automatically
- Data loss is possible

Data Loss

- Data loss potential depends on Data Protection Mode of Data Guard
 - Maximum Protection: No Data Loss
 - Maximum Availability: Maybe no data loss
 - Maximum Performance –
 - ARCH mode: Data loss likely
 - LGWR mode: Data loss possible
- Sorry – you can't have your cake and eat it too...

Network Configuration/Capacity

- Bandwidth requirements – based on txn volume
 - To keep up during normal and peak load
 - To catch up after an outage (burstable)
 - Compression option (11gR2)
 - Physical vs managed leased line networks
- Firewalls/Routers
 - Issues with firewalls, vendor specific
 - DG packets are different from usual Oracle Net
 - Turn off SQL*Net Application Inspection (Cisco)
 - May be issues when rebooting firewall (reflexive access list)

Network Bandwidth

- Measured in Megabits/second (Mbps)
- Determine Redo Volume from AWR reports
 - 3MB/sec redo = 25.2 Mbps
 - T1/DS1 link bandwidth is 1.544 Mbps
 - T3/DS3 link bandwidth is 44.7 Mbps
- Latency rule of thumb is 1ms per 33mi (53km)
- Directly affects the speed of replication, and in some cases commits on the Primary DB
- SQL*Net and OS network tuning parameters

Standby Server Capacity

- Same considerations as network
- Must be able to keep up and catch up
- Need extra power if reporting or backing up
- Can be RAC or single-instance
- Must be same Oracle version
- Should be same storage option (ASM, OMF, etc)
- Some mixed OS/CPU architecture is supported, enhanced in 11g (see Metalink 413484.1)

Failover Method

- Must consider application failover, not just DB
- Manual failover
 - Some form of monitoring required
 - Judgment exercised before initiating failover
 - Manual intervention to failover DB and apps
- Data Guard Observer (Fast Start Failover)
 - Runs on separate host outside of DG configuration
 - Monitors primary DB
 - Automatically initiates failover if outage detected
 - Beware of false positives!
 - Does not failover apps

Archive Log Management

- Archive logs are generated on primary & standby
- Best practice is to mirror both
- Need a process to purge the logs or DB will halt
- RMAN works well, assuming it runs on primary

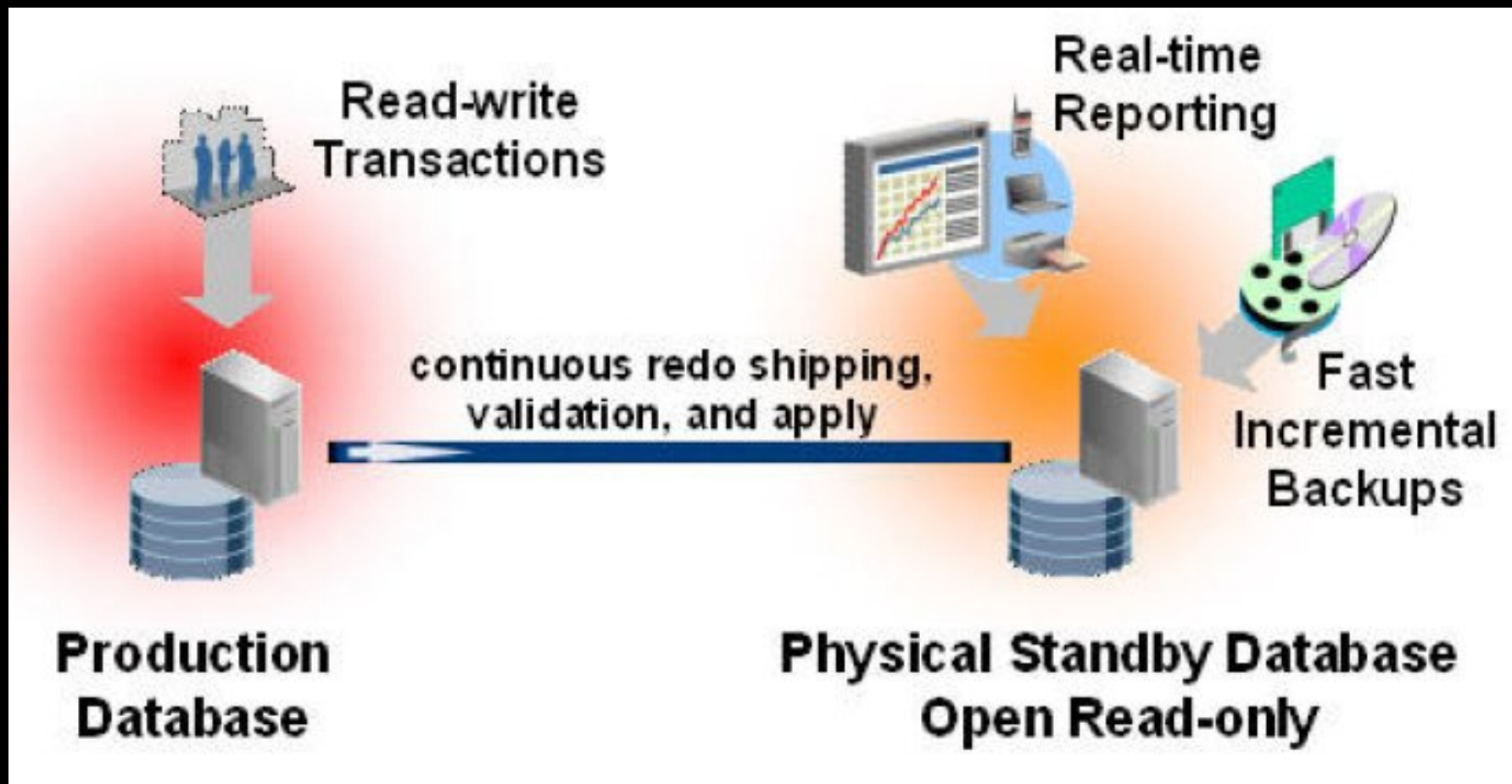
```
DELETE NOPROMPT OBSOLETE DEVICE TYPE DISK;  
BACKUP AS COMPRESSED BACKUPSET CHECK LOGICAL  
SKIP INACCESSIBLE FILESPERSET 20  
ARCHIVELOG ALL DELETE ALL INPUT;
```

Day1

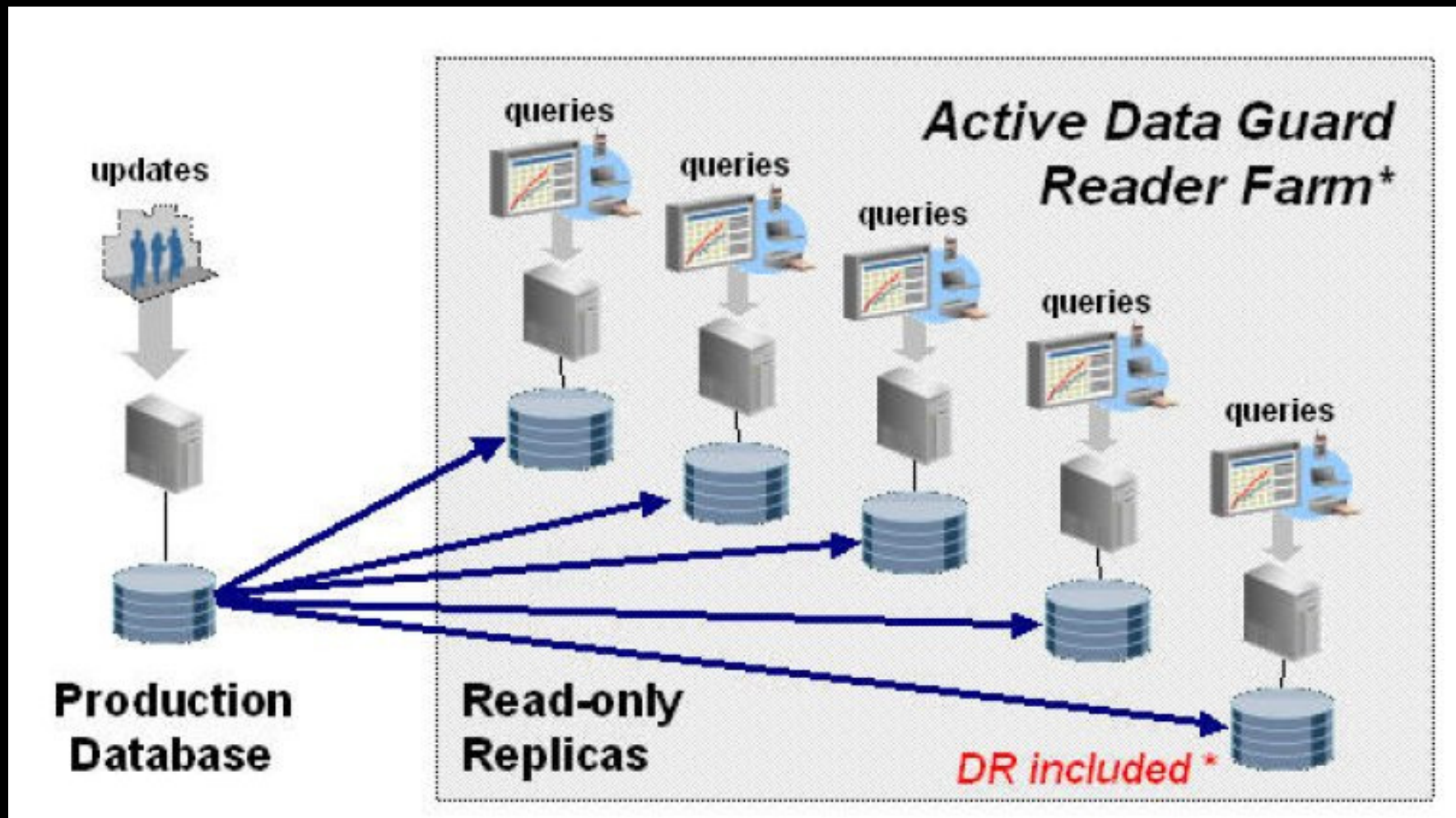
Day2 /oraflash: 01 02 03 04 05 06 07 08 09

Day3 /oraarch: 01 02 03 04 05 06 07 08 09

Secondary Reporting/Backup



Secondary Reporting/Backup



Secondary Reporting/Backup

- Off load backup and reporting to standby DB
- Backup can be done on standby DB instead of (or in addition to) primary DB
- Consider protection mode when using standby DB for backup
- 11g physical standby can be queried read-only for reporting
- 11g snapshot standby can be queried read-write but changes will be lost

Architecture

How It Works

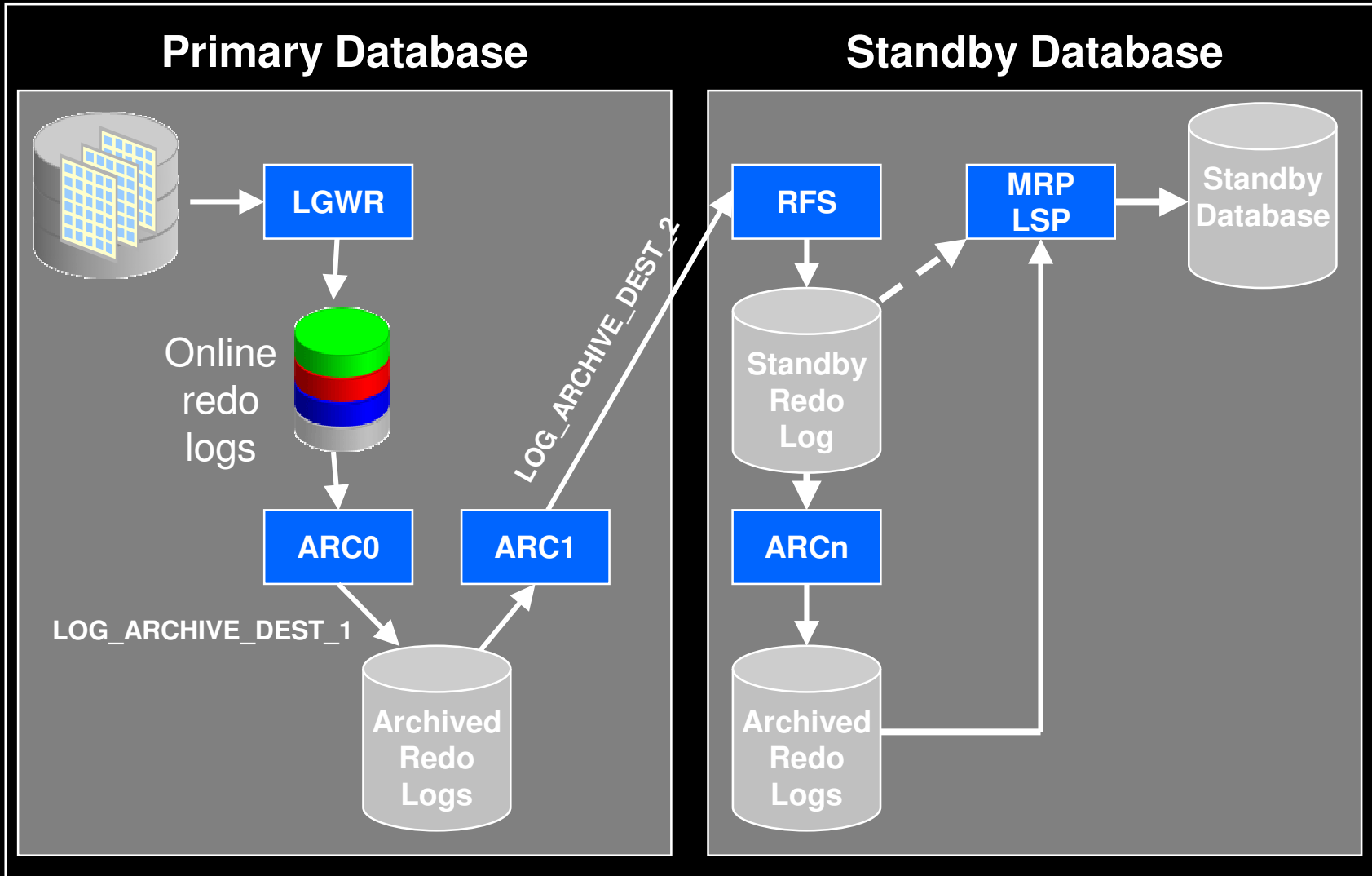
Data Guard Services

- Log transport services
 - Configurable using ARCH or LGWR
 - Used by Data Guard and Streams
- Log apply services
 - Redo Apply (Physical Standby)
 - Recovers the redo data received from the primary database and applies the redo to the physical standby database.
 - SQL Apply (Logical Standby)
 - Transforms the data in the redo received from the primary database into SQL statements and then executes the SQL statements on the standby database
- Role-management services
 - Change the role of a database from a standby to a primary, or from a primary to standby, using either a switchover or a failover

Transport Mechanisms

- ARCH
 - Can be used for maximum performance mode
 - Uses local archive logs as a buffer
 - Complete archive logs shipped when full
 - Minimal impact on system
 - Almost always means loss of data after failover
 - Tune with LOG_ARCHIVE_MAX_PROCESSES
 - Default is 2, Max is 30
 - Set MAX_CONNECTIONS to 2 or higher to enable remote parallel archiving
 - Max is 5

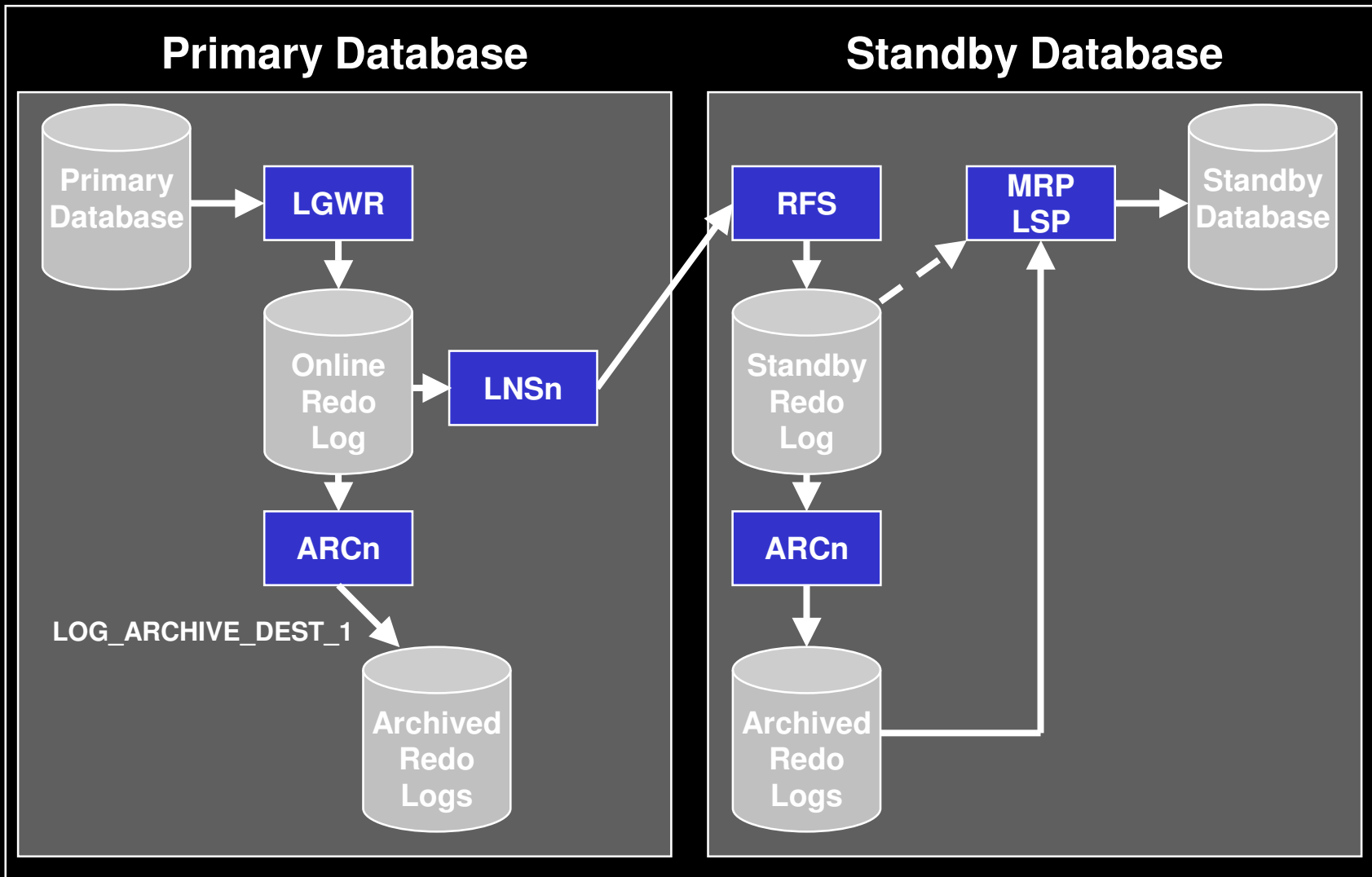
Standby Database ARCH Redo Transmission



Transport Mechanisms

- LGWR ASYNC
 - Can be used for maximum performance mode
 - Uses online redo logs to buffer peaks in redo generation
 - LGWR attempts to write to both local redo log, and remote standby redo log
 - Failure to write remote file will not stop primary database
 - Gaps are resolved automatically
 - Data loss may occur if there is lag between local and remote writes

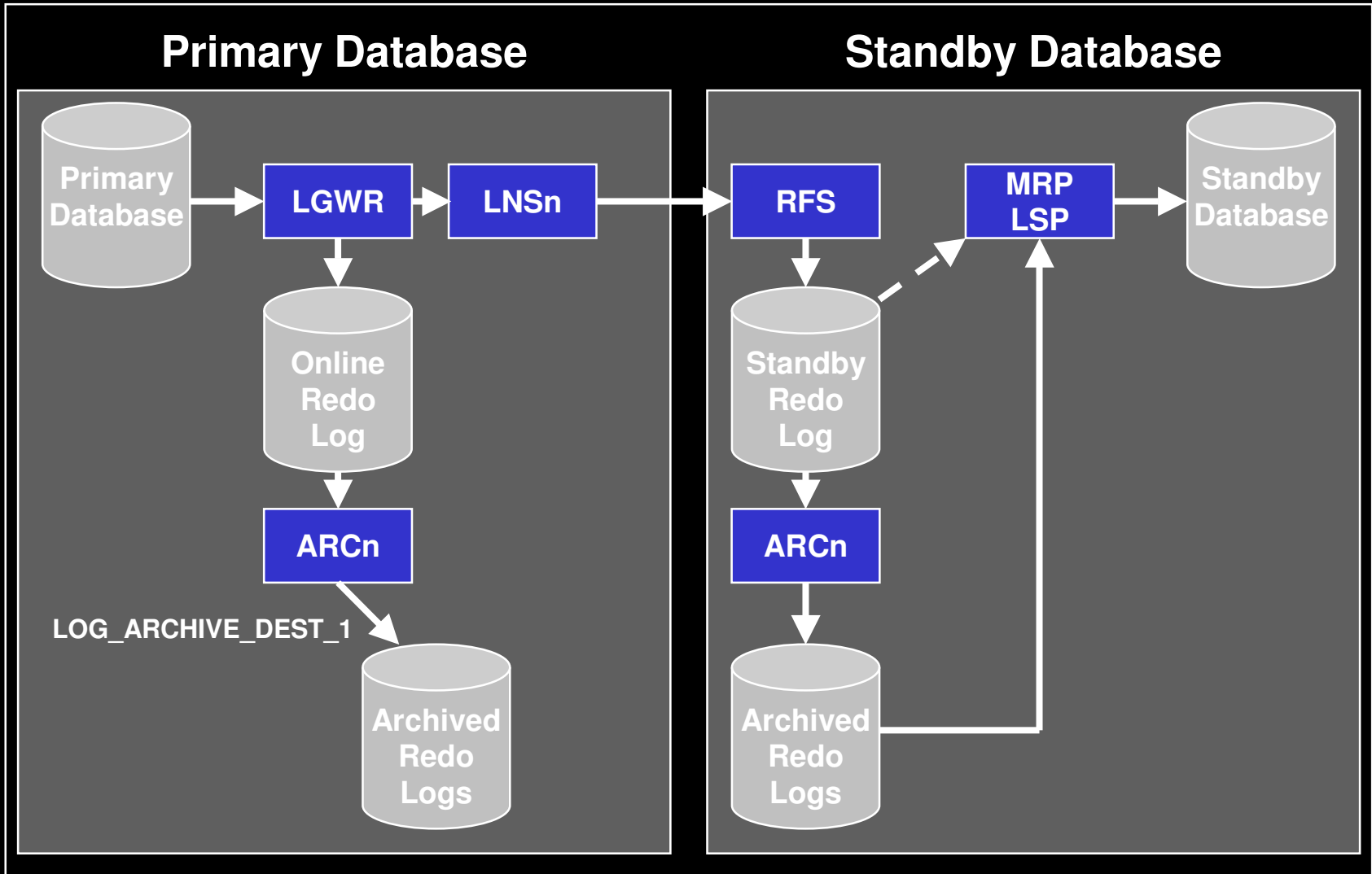
Standby Database LGWR (ASYNC) Redo Tx



Transport Mechanisms

- LGWR SYNC
 - Required for Maximum Protection Mode
 - LGWR *must* write to both local redo log and at least one remote standby redo log
 - Transaction will not commit until remote write finishes
 - Database will *stop* if remote write fails
 - Network latency can severely impact primary database performance
 - No data loss

Standby Database LGWR (SYNC) Redo Tx



Real-Time Apply

- Redo data is applied to the standby database as soon as it is received from the primary database
 - In Oracle9i Data Guard this apply has to wait till an archive log is created on the standby database

- **For Redo Apply:**

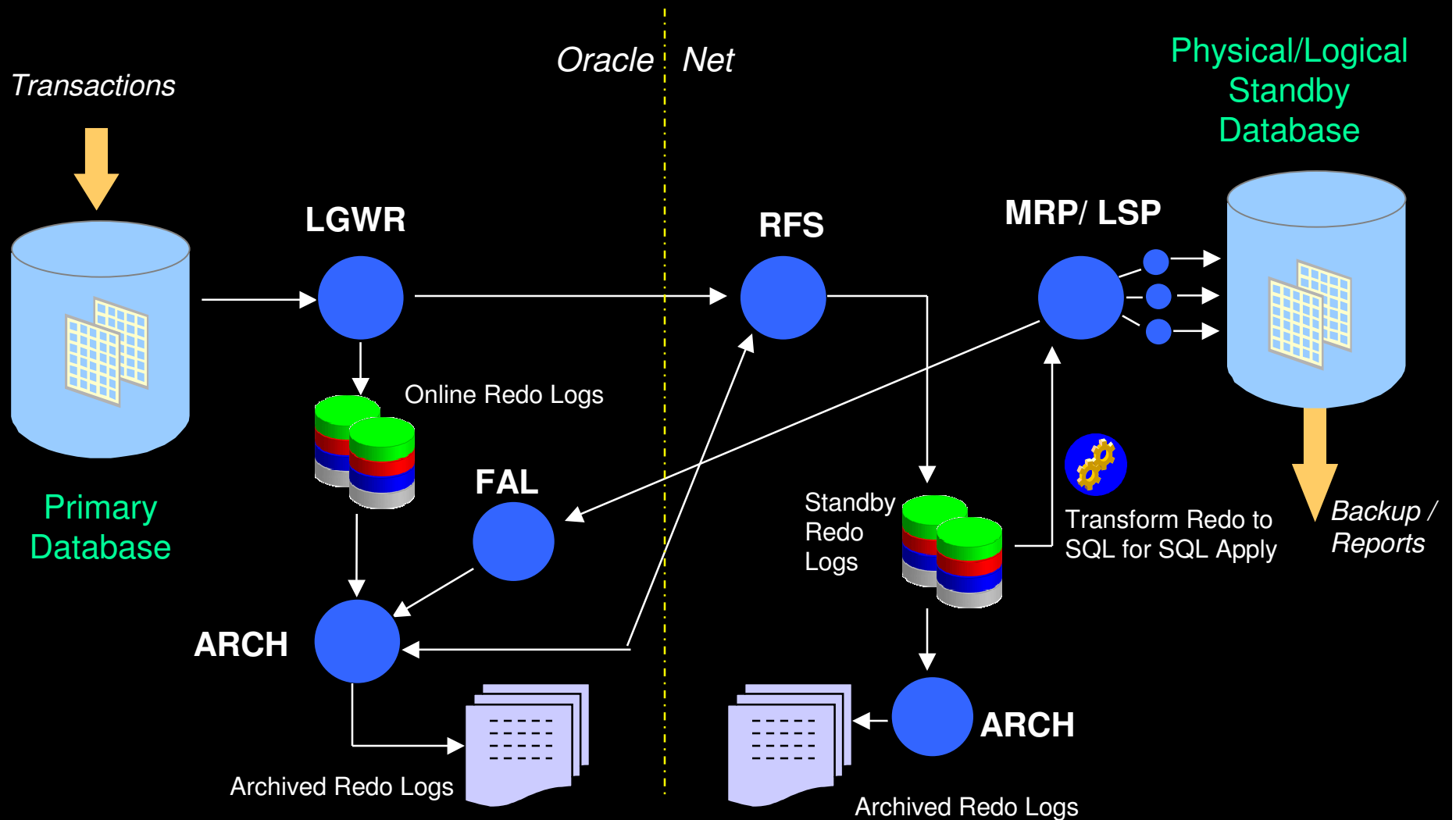
```
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE  
USING CURRENT LOGFILE
```

- **For SQL Apply:**

```
ALTER DATABASE START LOGICAL STANDBY APPLY  
IMMEDIATE
```

- **When real time apply is enabled, the**
`RECOVERY_MODE` column in `GV$ARCHIVE_DEST_STATUS`
displays “`MANAGED REAL TIME APPLY`”

Real-Time Apply Architecture



Role Transition: Switchover vs Failover

- Switchover

- Gracefully swapping roles between Primary DB and Standby DB
- No Data Loss
- Generally done for patching, upgrades, testing, etc

- Failover

- Sudden, unexpected failure of Primary DB
- Activating Standby DB as new Primary DB
- Maybe data loss
- Primary DB is lost and cannot function as Standby

Data Guard Broker

- Optional component
- Centralized management of Data Guard configuration
- Use ***INSTEAD*** of managing spfile parameters and using alter database commands!
- Starts optional DMON process on all instances
- Command line or GUI interface
 - Create configuration
 - Add / drop databases
 - Enable / disable databases / instances
 - Perform switchover / failover with single command

Fast Start Failover - Data Guard Observer

- Fast-start failover: Automatically, quickly, and reliably fail over to a designated, synchronized standby database
- After a fast-start failover occurs, the old primary database is automatically reconfigured as a new standby database upon reconnection to the configuration
- Maintain uptime and increase the availability, as well as the robustness of disaster recovery
- Less need for manual intervention

Data Guard and RAC

- Primary and/or standby database(s) can be RAC
- All instances of RAC primary ship redo to standby
- One full set of standby redo logs per thread/node
- Only one instance of RAC standby receives redo
- Only one instance of RAC standby applies redo (in DG Broker configuration these are the same)
- When transitioning roles, only one instance of each RAC database should be up
- DG Broker is strongly recommended

Highlights of New Features in 11g

- Real Time Query (Active Data Guard)
- Up to 30 standby databases
- Redo transport compression
- Corrupt block recovery from standby
- FAN/FCF integration into DG Broker
- `ALTER SYSTEM FLUSH REDO` (from primary)
- Apply lag tolerance for Real Time Query
`STANDBY_MAX_DATA_DELAY`
- Synchronize standby with primary
`ALTER SESSION SYNC WITH PRIMARY`
- Redundant parameters/statements deprecated
(eg. `FAL_CLIENT`, `set state='OFFLINE'`)

Setting Up Data Guard

Setting Up Data Guard

- Could use Enterprise Manager wizzard
(Requires Grid Control)
- Primary DB in archivelog mode
- Turn on FORCE LOGGING
- Enable FLASHBACK DATABASE
- Symmetrical configuration of standby
(eg. Storage layout, FRA, Oracle version, etc)
- Configure Oracle Net on both
 - Static register DG Broker connection in listener.ora
 - TNS entries for DG Broker and Log Transport
 - Consider tuning parameters in sqlnet.ora

Setting Up Data Guard

Create pfile from spfile and modify

```
*.db_name='chicago'  
*.db_unique_name='boston'  
*.instance_name='boston'  
*.log_archive_config='dg_config=(chicago,boston)'  
*.log_archive_dest_3='service=chicago LGWR ASYNC  
    valid_for=(online_logfile,primary_role)  
    db_unique_name=chicago'  
*.db_file_name_convert=  
    'c:\oradata\chicago\','c:\oradata\boston\  
*.log_file_name_convert=  
    'c:\oradata\chicago\','c:\oradata\boston\  
*.standby_file_management=auto  
*.fal_server='chicago'  
*.dg_broker_config_file1='C:\app\oracle\...\dr1_boston.dat'  
*.dg_broker_config_file2='C:\app\oracle\...\dr2_boston.dat'
```

Setting Up Data Guard

- Take complete full (hot) backup of primary
 - All datafiles
 - Archivelogs since start of backup
 - Controlfile for standby
- Copy backup set to standby (same directory)
- Copy pfile, password file and Oracle Net files
- Create instance on standby
 - Create admin, data, FRA, archive directories
 - Modify pfile for standby (reverse db names)
 - Create Windows service (oradim), etc
 - ***DO NOT*** configure to auto start (open)!

Setting Up Data Guard

- Start instance nomount
- Duplicate primary database as standby (RMAN)
- Shutdown and startup mount
- Create standby redo logs
 - One set for each thread of primary
 - One extra group for each thread
 - Do the same on the primary for role transition
- Start managed recovery on standby
- Verify configuration (health check, switchover)
- Optionally, start DG Broker and create a Broker configuration

Managing & Monitoring

Keeping the standby healthy

Maintaining The Standby Database

- All the usual DB health checks apply
 - In particular, watch the FRA on primary & standby
- Data Guard health checks
 - DG Broker
 - User scripts
 - Read-only reality checks
- Archive gap resolution
- Patching
- Periodic switchover tests

Data Guard Health Checks

- DG Broker built-in health check
 - Improved in 11g
 - Still don't trust it
- User scripts
 - Seems common in the industry
 - See example next slide
 - Run daily/hourly and report success/fail
 - Comments?
- Read-only reality check
 - Open the standby read-only
 - Query record count or recent data

Data Guard Health Checks – User Script

- Check FRA usage on both primary & standby
 - Regular backups should purge archived redo logs
 - An unresolved gap, or failed backup will cause FRA to fill
 - When FRA fills up, primary DB halts
- Check SCN of primary against highest SCN in standby redo logs
 - Should be small difference
 - Indicates lag of the standby *receiving* redo
 - High value indicates failure of *transport* services
- Check SCN of standby against online redo logs on primary
 - Should be within range of last 1 or 2 redo logs
 - Indicates lag of the standby *applying* redo
 - Higher difference indicates failure of *redo apply* services
 - Could be a gap resolution issue

Archive Gap Resolution

- When properly configured, gap resolution is automatic with the FAL mechanism
- Still, sometimes gaps are not detected or resolved automatically
- Manual gap resolution may be necessary
- Perform health check regularly to detect gaps
- Gaps represent lost data and if left too long may not be able to resolve
- If gap cannot be resolved (missing or corrupt archive log, too old, etc), then standby DB must be ***re-created !***

Manual Gap Resolution

- Determine missing archive log
 - Need thread & sequence number
 - Query V\$ARCHIVE_GAP on standby
 - Compare standby SCN to V\$ARCHIVED_LOG on primary
- Locate a copy of the missing log on the primary
 - In the FRA or other archive destination
 - Recover from backup
- Copy (backup & restore) log to standby
- `ALTER SYSTEM REGISTER LOGFILE;`
- Recovery should proceed automatically
- Perform health check
- May be more than one missing log!

Patching

- Special considerations when patching in a Data Guard environment
- Primary and standby must be same patch level
- ***MUST*** read the patch release notes carefully!
- Most (not all) patches are applied to standby first, then to primary
- These can usually be done as rolling patches with minimal down time
- The 10.2.0.4.4 patch required patching the primary first – necessitating an outage

Switchover Testing

- Periodic switchover tests validate the *entire* Data Guard configuration
 - Are the init parameters correct?
 - Does role transition work?
 - Can the primary function as a standby?
 - Does redo transport work in reverse?
 - Do databases assume correct role after reboot/restart?
- Highly recommended as part of quarterly PSU patching procedure
- May want to perform failover testing as well
 - Scary on production DB

Live Demo

That Other Operating System

Health Check

Data Guard Broker

Switchover

Real Time Query

Failover