ORACLE

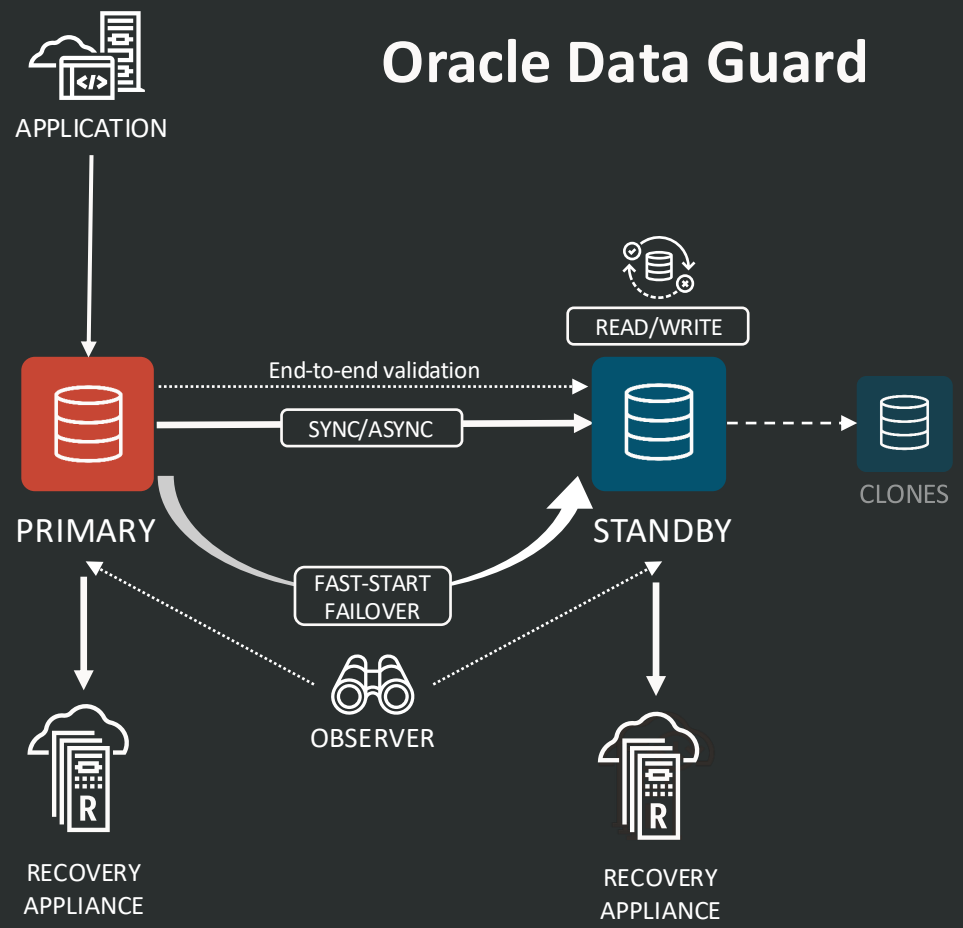
Faster, Safer, Always Awesome: Active Data Guard for AI workloads

Oracle Community Tour EMEA 2026

Ludovico Caldara

Product Manager Oracle Data Guard, Active Data Guard, and Flashback Technologies

The best disaster recovery solution for the Oracle AI Database



Disaster Recovery

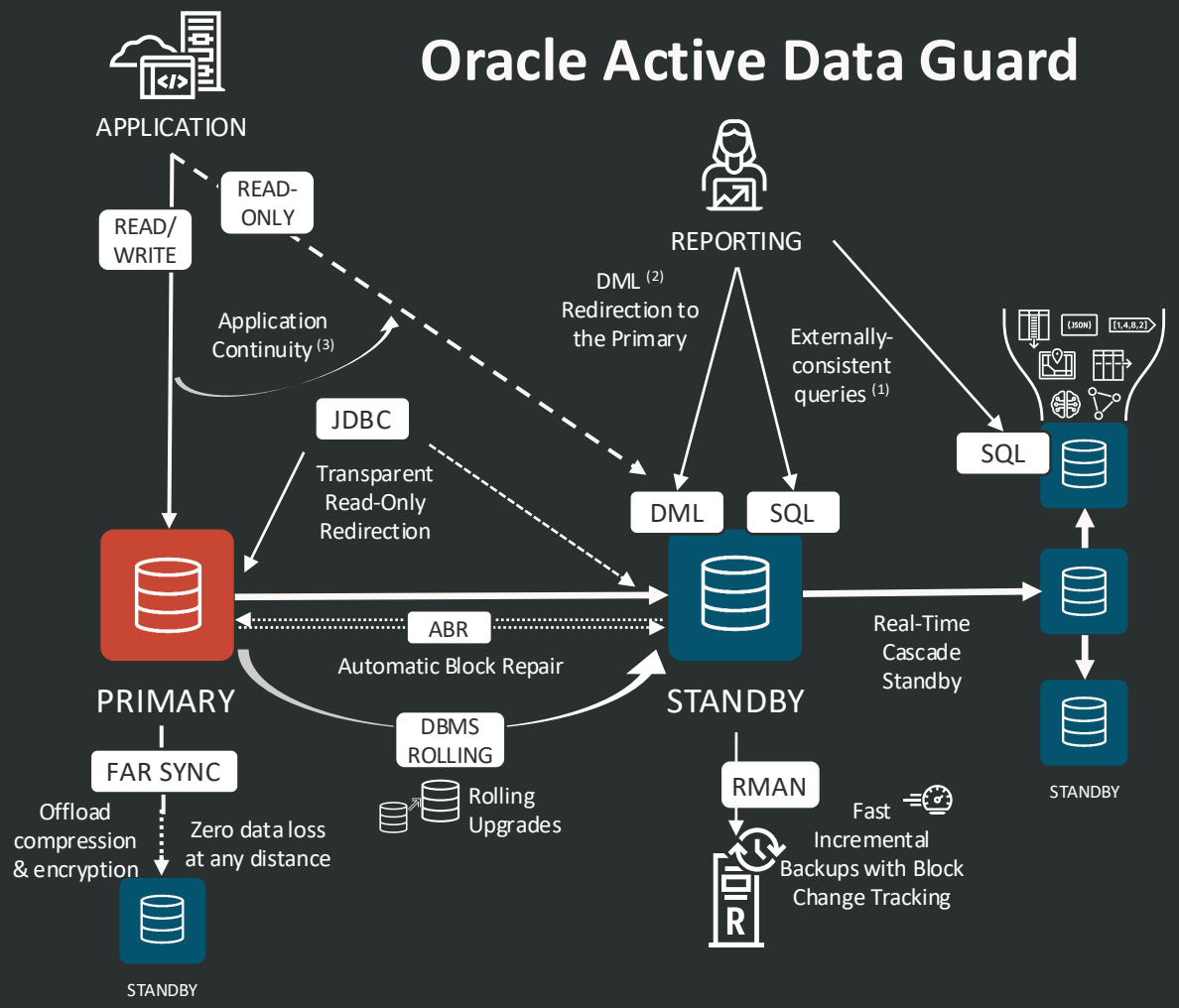
- Standby database(s) ready to take over
- End-to-end validation: corruptions are not replicated
- Synchronous or asynchronous replication
- Zero or near-zero data loss protection
- Automatic failover, low Recovery Time Objective (RTO)
- Backups can run on the standby database

Testing and cloning

- The standby database can be temporarily open read/write
- The standby database can be used as a source for clones

Copyright © 2026, Oracle and/or its affiliates

Higher Performance, Protection, Availability, and Return on Investment



(1) Full support for Vectors, Graph, Spatial, JSON, In-memory, etc.
(2) And PL/SQL, and some DDLs such as CREATE GLOBAL TEMPORARY TABLE
(3) Mask outages and failovers to the application with minimal or no additional coding

Offload & Scale

Don't leave your standby infrastructure idle:

- **Offload** queries, reports, and backups
- **Linearly scale** the application's read-only activity
- **Increase** read/write **throughput** by freeing important resources
- Transparently load balance across replicas

Enhanced Protection

- All your transactions are **protected regardless of the distance**
- Block corruptions are **fixed** on the fly **without application errors**
- Cascaded standbys protect your transactions without delay
- Backups are **incredibly fast**, also on the standby database

Higher Availability

- Major **upgrades are transparent** or require minimal downtime
- **Mask failovers** to the application **without additional coding**

Copyright © 2026, Oracle and/or its affiliates

Oracle Active (*) Data Guard

Actively protecting data for the future *both* on-premises and in the cloud

12c

- Active Data Guard Real-Time Cascade (*)
- Fast Sync
- Broker for Cascaded Standby Databases
- Resumable Switchover Operations
- Rolling Upgrade Using Active Data Guard (*)
- Single Command Role Transitions
- Data Guard Broker PDB Migration or Failover
- Multi-Instance Redo Apply
- Zero Data Loss at any distance – Far Sync (*)
- Protection During Database Rolling Upgrade (*)
- Password Files Synchronization
- Oracle Database In-Memory on Oracle Active Data Guard (*)
- Preserving Application Connections During Role Changes (*)
- Application Continuity (Active Data Guard or RAC) (*)

18c

- RMAN recover standby simplification
- Shadow Lost Write Protection
- Transparent Application Continuity (*)
- AWR reports for the standby workload (*)
- Automatic Correction of Non-logged Blocks at Standby Database (*)

19c

- Flashback Standby when Primary database is flashed back
- In-Memory Column Store on Multi-Instance Redo apply (*)
- Propagate Restore Points from Primary to Standby site
- Simplified Database Parameter Management
- Finer granularity Supplemental Logging
- Dynamically Change FSFO target
- Updates on Active Data Guard (DML Redirect) (*)
- Observe only mode for FSFO

21c

- Data Guard per Pluggable Database
- Standby Result Cache preservation (*)
- Fast Start Failover Configuration Validation & Call Outs
- Data Guard Broker Client Side Standardized Directory Structure
- Data Guard Broker Far Sync Instance Creation (*)
- Fast Start Failover Lag Allowance in Max Availability Mode
- FarSync for Max Performance Mode (*)
- PDB recovery isolation (*)

26ai

- Faster role transitions
- Enhanced observability
- Enhanced diagnostics and validations
- Broker management with SQLcl, PL/SQL, and REST APIs
- Data Guard per Pluggable Database Real-Time Query (*)
- Automatic tempfile creation at Standby Database
- Pre-emptive transition to LAGGING state
- Transport lag for measuring data-loss
- Standby-native AWR snapshots (*)
- Broker command-line JSON output
- Broker configuration and member tagging
- AI inference and vector search on the standby (*)
- Rolling upgrades support for Application Continuity (*)
- Restrict primary role to a list of databases
- TLS authentication for redo transport
- Faster ASYNC redo transport
- …

Copyright © 2026, Oracle and/or its affiliates

Don't miss this blog series!

Oracle Data Guard 26ai New Features

30+ Blog Posts

https://www.ludovicocaldara.net/blog/dg-26ai-series/

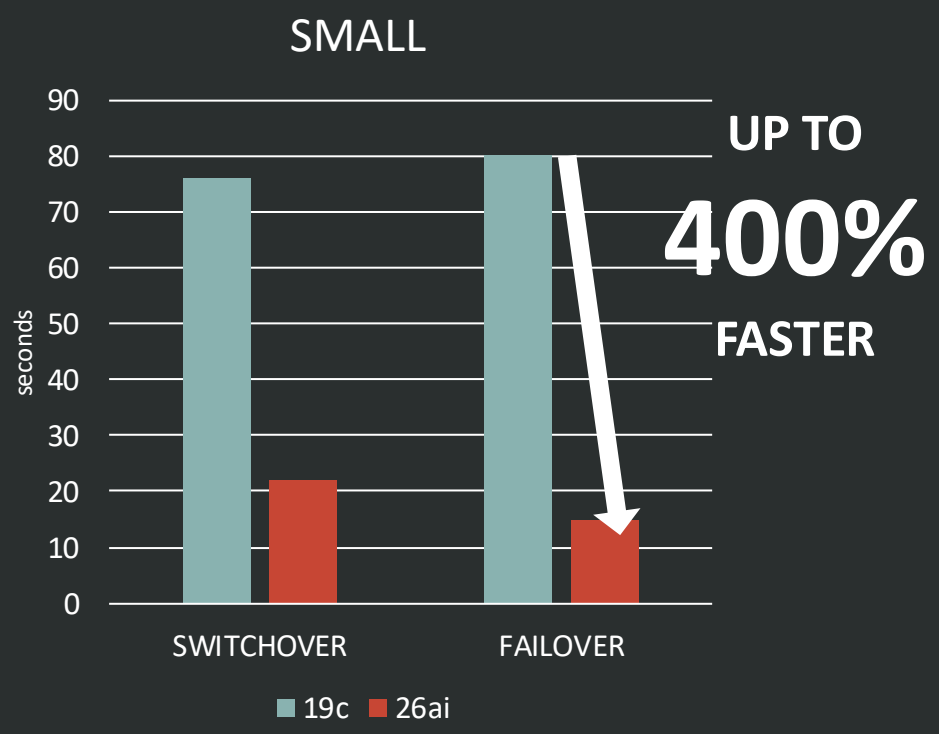


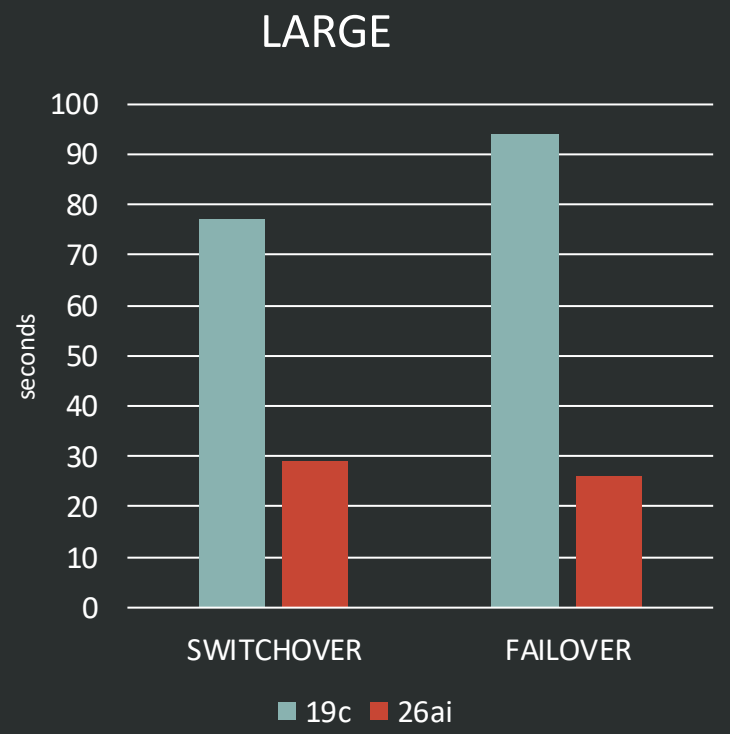
Copyright © 2026, Oracle and/or its affiliates

Faster

New In 26ai

Faster role transitions in Oracle Data Guard 26ai





	System	PDBs	Data Files	Services	Redo Rate
SMALL	2-node Exadata RAC	5	50	10	60 MB/s
LARGE	4-node Exadata RAC	12	500	12	100 MB/s

Read more:

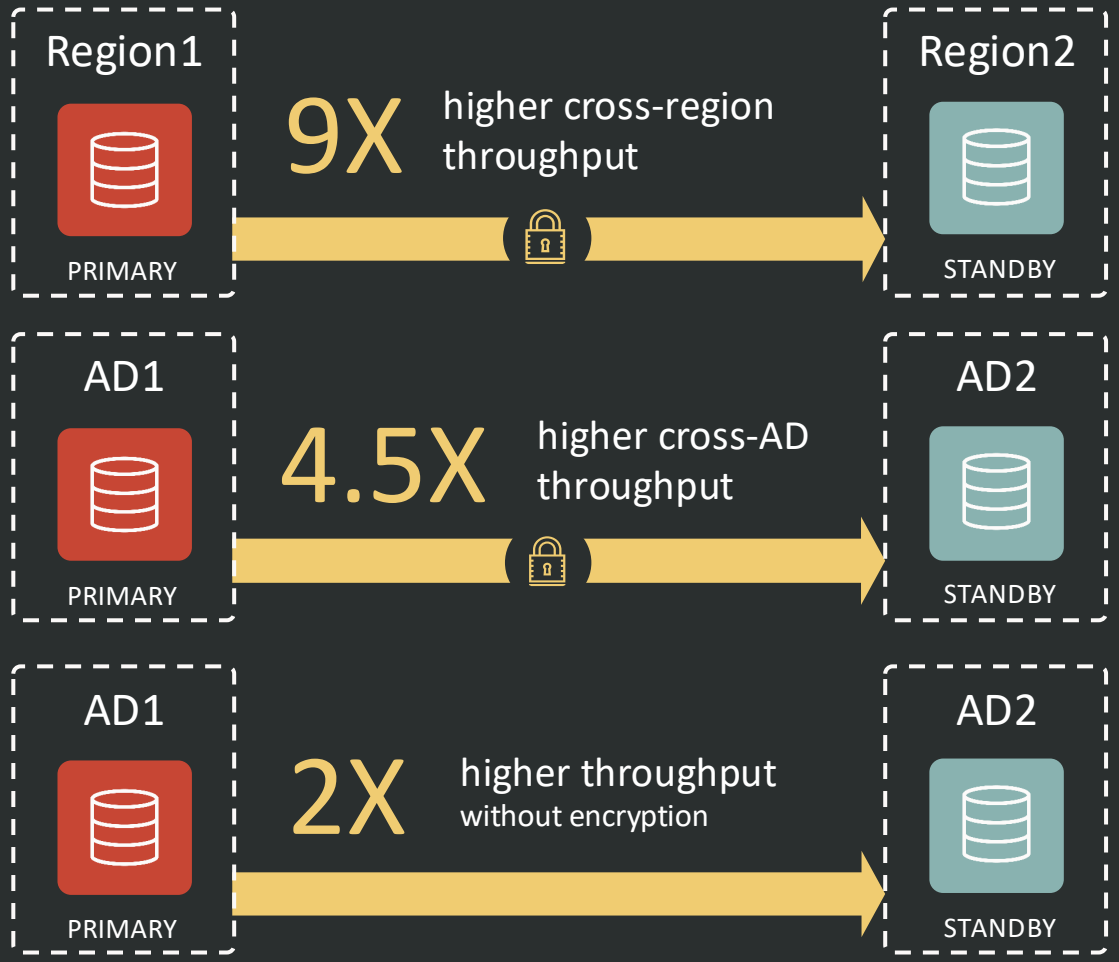
Copyright © 2026, Oracle and/or its affiliates

Faster!

New In 26ai

High-Performance Oracle Data Guard Encrypted Transport

Up to 9X higher throughput for encrypted transport across regions



Multiple redo transport optimizations

- TLS vectored I/O
- Standby redo logs on Exadata flash
- More efficient control file transactions
- TLS-based encryption instead of Native Encryption
- OpenSSL 3.5
- Parallel connections for ASYNC redo transport

Improves cross-region business continuity by reducing redo transport bottlenecks over longer-distance links

Expands throughput headroom for high-change-rate mission-critical workloads

Copyright © 2026, Oracle and/or its affiliates

Why TLS over NNE

Typically, 10-20% lower encryption latency

AES-NI acceleration

TLS can use CPU AES-NI instructions through its optimized TLS crypto library, reducing encryption CPU cost.

Lower per-message overhead

TLS works at a lower layer and adds less work per SQL*Net exchange.
Best for chatty traffic.

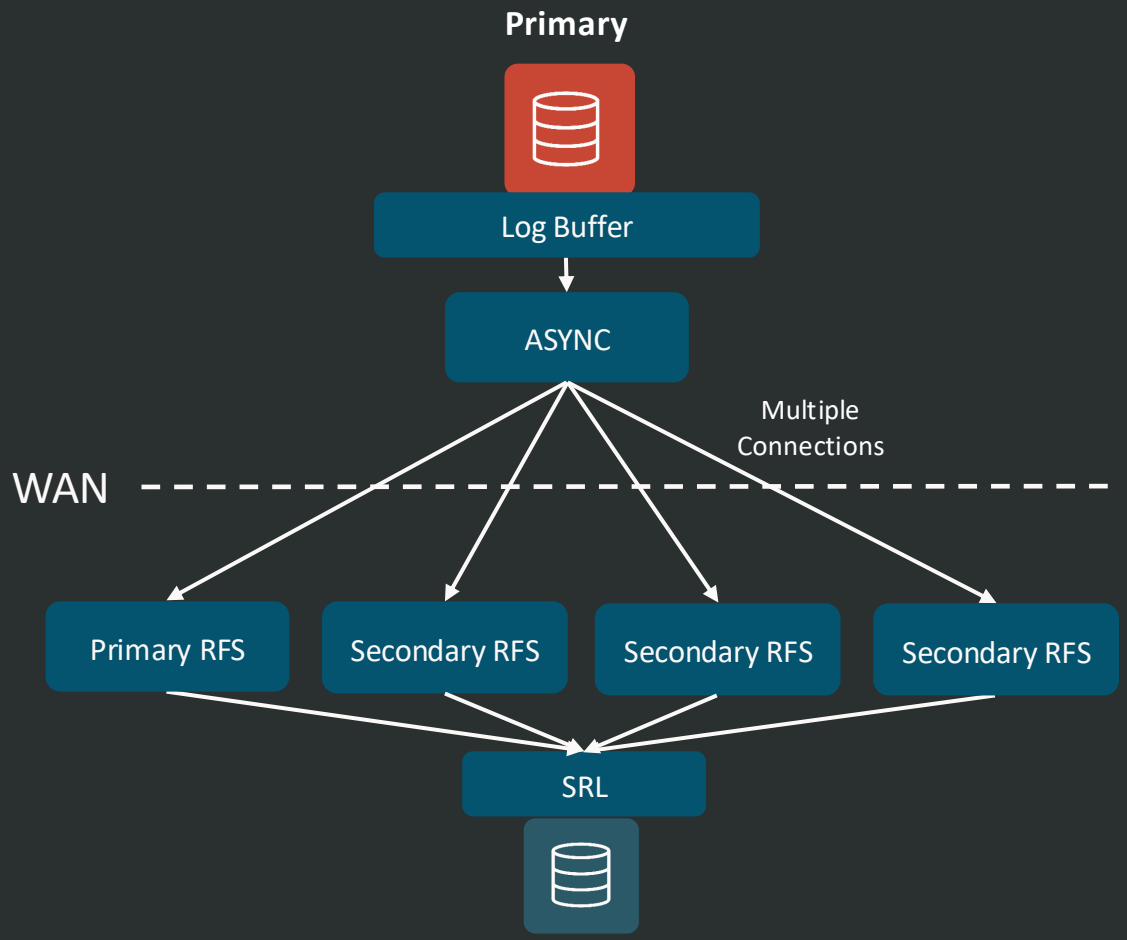
Fewer copies and framing steps

TLS avoids some of the extra encryption-record handling and buffer movement inherent in NNE's protocol-layer implementation.

Copyright © 2026, Oracle and/or its affiliates

Faster ASYNC redo log shipping with multiple connections

New In 26ai



How it works

- Multiple connections maximize network bandwidth usage
- There is still one ASYNC process
- CPU can still constrain the process, e.g. for encryption

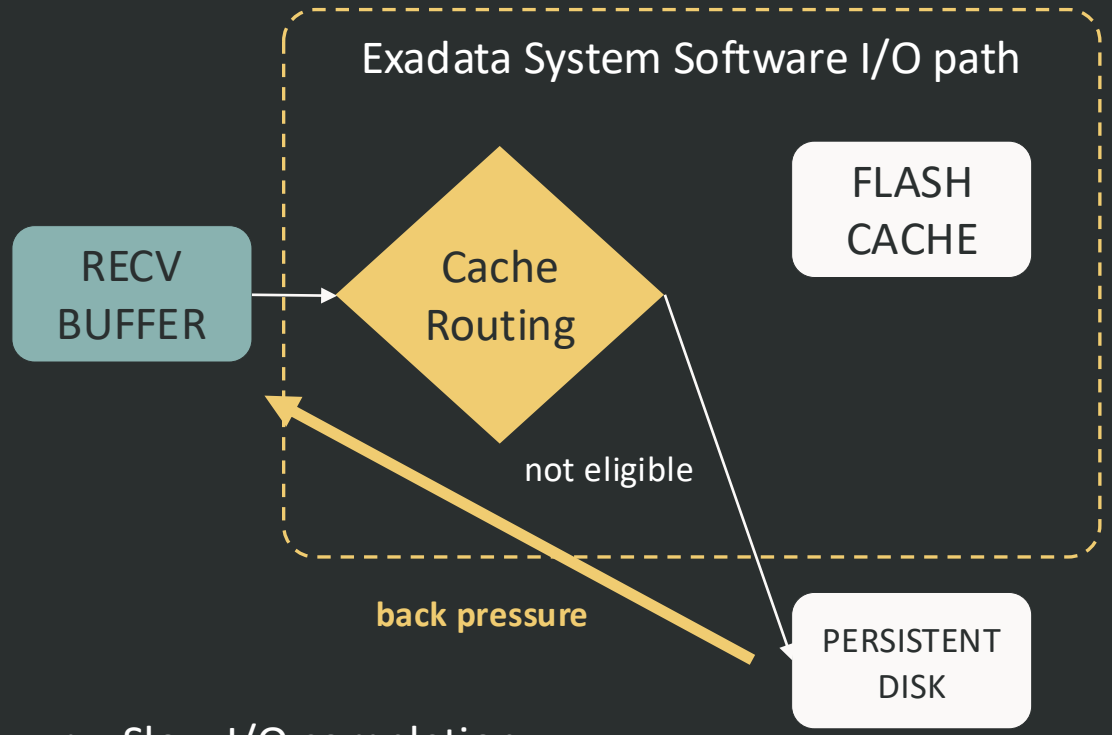
Benefits

- Higher throughput for ASYNC redo transport
- Reduced risk of disaster recovery lag
- Key in cloud setups where per-connection network is limited

Copyright © 2026, Oracle and/or its affiliates

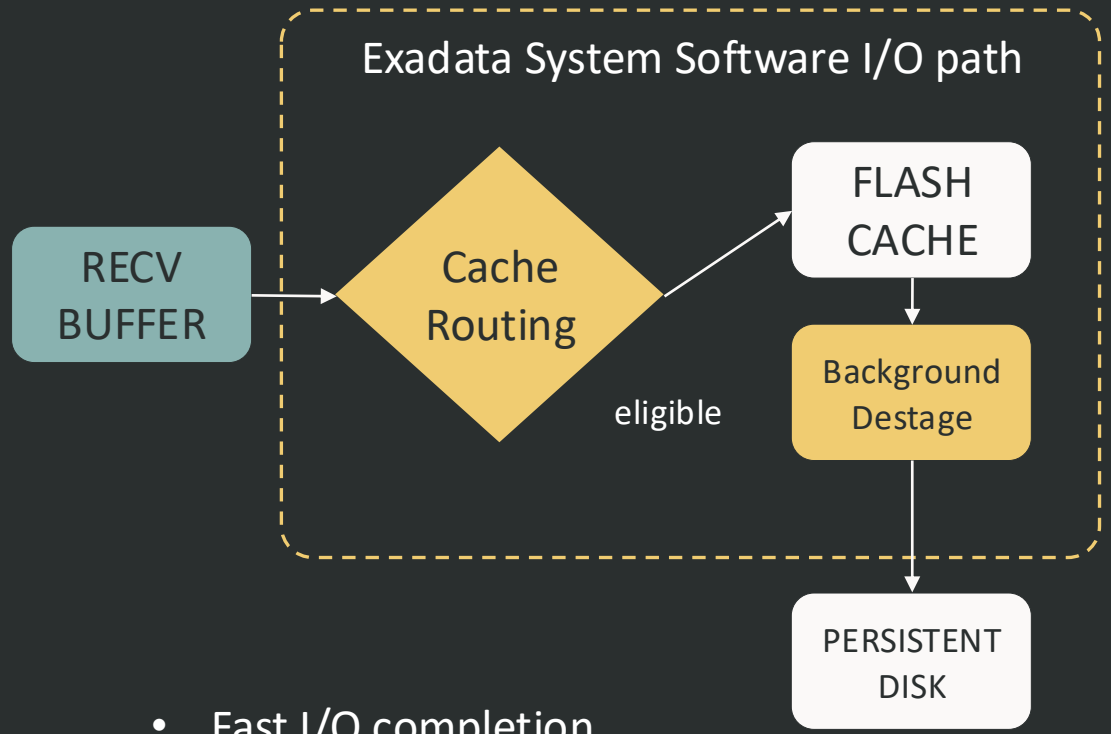
Standby Redo Logs and Exadata I/O Path

Before: I/O Bound (Problem)



- Slow I/O completion
- Higher memory consumption
- Slower Transport

After: Cached (Solution)



- Fast I/O completion
- No backpressure
- 1-line code change 😅

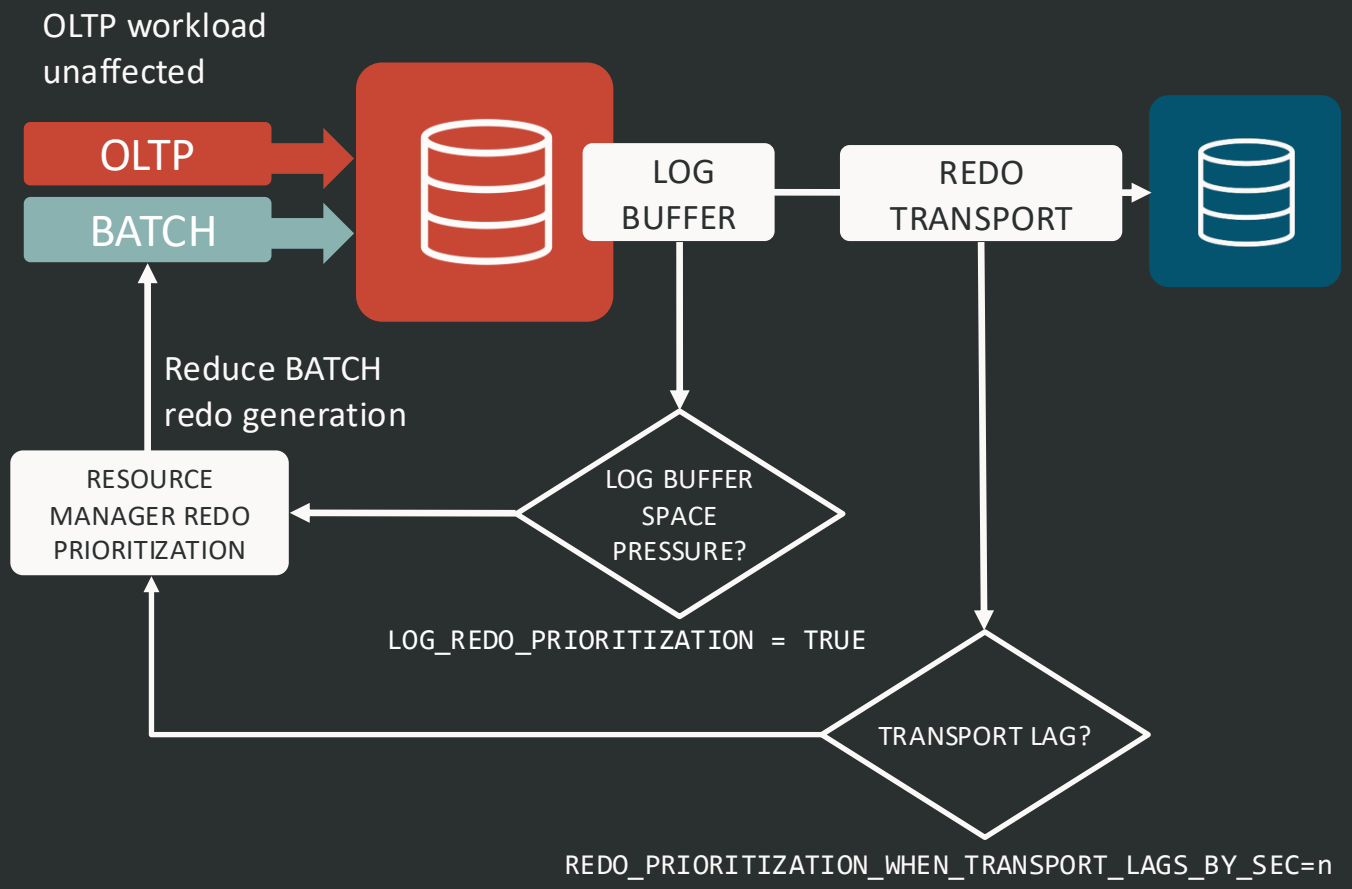
Copyright © 2026, Oracle and/or its affiliates

More resilient

New In 26ai

Auto Redo Prioritization

Helps manage redo generation spikes



What it does

- Adapts to current redo conditions
- Slows down BATCH redo only
- Activates and deactivates automatically
- Ideal in mixed workloads
- Responds to two independent conditions

Log buffer space-based

- Reduces `log buffer space` waits
- Can improve OLTP responsiveness

Transport lag-based

- Reduces asynchronous transport lag
- Keeps the standby closer to the primary
- Helps limit potential data-loss exposure

Copyright © 2026, Oracle and/or its affiliates

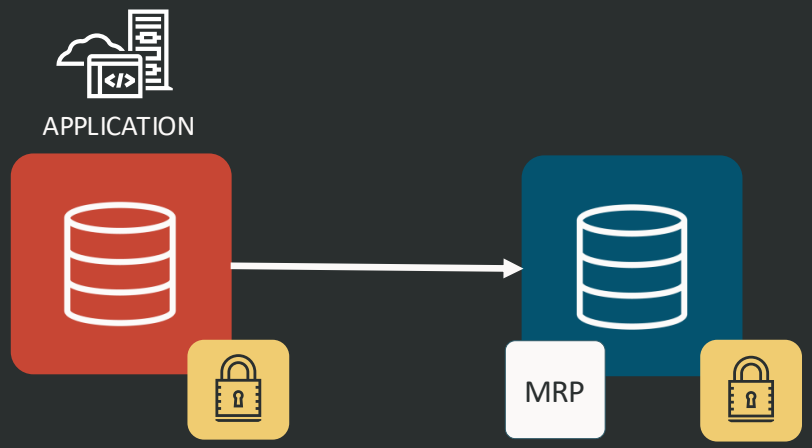
New In 26ai

Oracle Data Guard Online Tablespace Encryption

Encrypt at scale without stopping recovery: continuous protection, zero downtime impact.

Why it matters

- Prevent apply lag buildup on very large databases
- Preserve Data Guard protection and failover readiness
- Encrypt 100s of TB online without disruption



Decouple encryption from recovery to keep redo apply running—even on multi-hundred TB systems.

```
SQL> -- on the standby database
SQL> ALTER SYSTEM SET DB_RECOVERY_AUTO_REKEY=OFF;
```

```
SQL> -- on the primary database
SQL> ALTER TABLESPACE bigtbsp ENCRYPTION ONLINE ENCRYPT;
```

1. The tablespace encryption starts on the primary.
2. The recovery process marks the tablespace as ENCRYPTING.
3. The recovery continues without interruptions.

```
SQL> -- on the standby database
SQL> ALTER TABLESPACE bigtbsp
       ENCRYPTION ONLINE FINISH ENCRYPT;
```

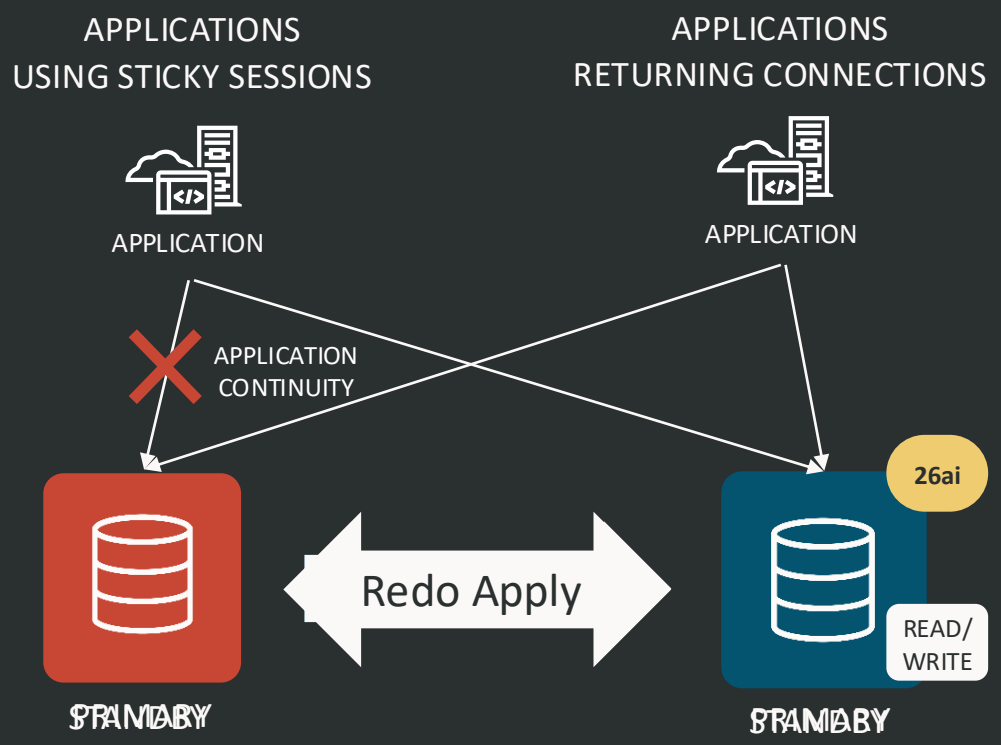
4. The user process encrypts the tablespace online on the standby.

Note: Standby CDB$ROOT must be open read-only (no ADG license required)

Copyright © 2026, Oracle and/or its affiliates

New In 26ai

Rolling Upgrade support for Application Continuity from 19c to 26ai



Applications returning connections

- Transparently disconnect at switchover time
- Reconnect to the new database a few seconds later

Applications using sticky sessions

- Application Continuity rebuilds the session state
- The disconnection is hidden from the application
- **Applications start transactions in 19c and commit in 26ai**

Backported to 19.30

Copyright © 2026, Oracle and/or its affiliates

More Secure

New In 26ai

Data Guard Redo Transport TLS Authentication

How it works

- Redo transport uses TCPS with TLS enabled between databases.
- Databases authenticate using certificates stored in wallets, OKV, or HSMs.
- Each database can present its own certificate-based identity.

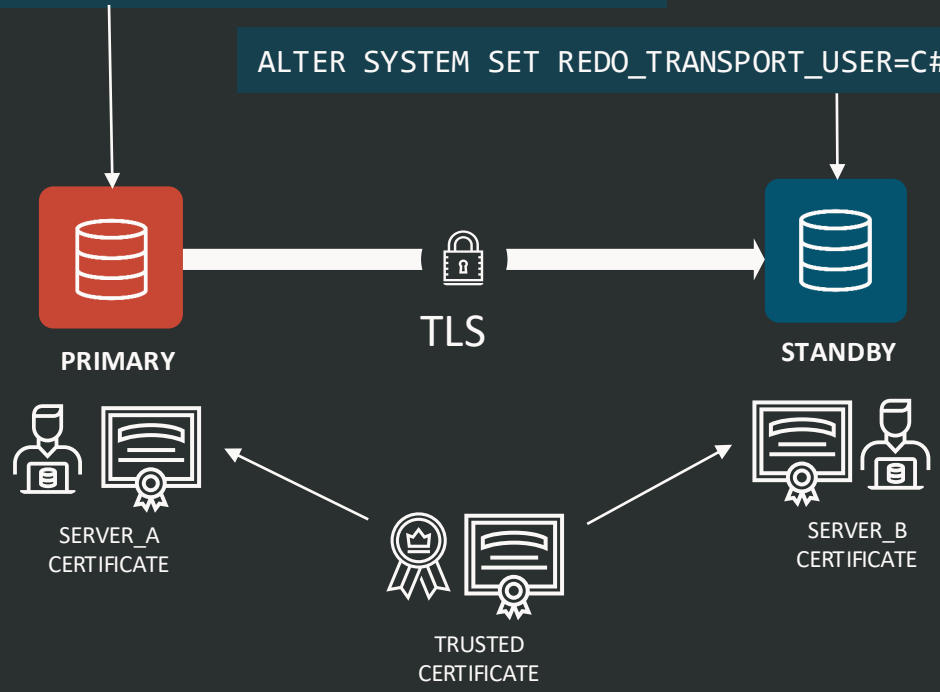
Benefits

- Stronger security and compliance via PKI-based authentication.
- Eliminates shared password files and improves auditability.
- Enables different identities for GoldenGate Downstream Capture sources and similar use cases.

```
CREATE USER C##SRV_A IDENTIFIED EXTERNALLY AS 'CN=SRV_A' CONTAINER=ALL;
GRANT SYSOPER TO C##SRV_A CONTAINER=ALL;

CREATE USER C##SRV_B IDENTIFIED EXTERNALLY AS 'CN=SRV_B' CONTAINER=ALL;
GRANT SYSOPER TO C##SRV_B CONTAINER=ALL;
```

```
ALTER SYSTEM SET REDO_TRANSPORT_USER=C##SRV_A;
```

```
ALTER SYSTEM SET REDO_TRANSPORT_USER=C##SRV_B;
```



Copyright © 2026, Oracle and/or its affiliates

New In 26ai

Data Guard Redo Transport Kerberos Authentication

How it works

- Redo transport authenticates using Kerberos instead of password files.
- Databases use centrally managed Kerberos principals for identity.
- Can integrate with enterprise KDCs, including Microsoft Entra ID.

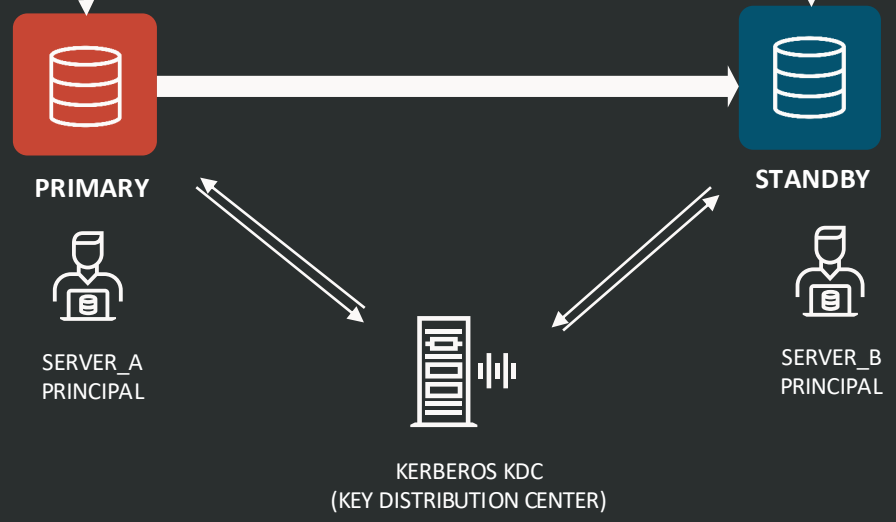
Benefits

- Reduces password management and aligns with enterprise identity services.
- Improves auditability and provides stronger identity separation.
- Enables different identities for GoldenGate Downstream Capture sources and similar use cases.

```
CREATE USER C##SRV_A IDENTIFIED EXTERNALLY
    AS 'oracle/SRV_A@domain' CONTAINER=ALL;
GRANT SYSOPER TO C##SRV_A CONTAINER=ALL;

CREATE USER C##SRV_B IDENTIFIED EXTERNALLY
    AS 'oracle/SRV_B@domain' CONTAINER=ALL;
GRANT SYSOPER TO C##SRV_B CONTAINER=ALL;
```

```
ALTER SYSTEM SET REDO_TRANSPORT_USER=C##SRV_A;
```

```
ALTER SYSTEM SET REDO_TRANSPORT_USER=C##SRV_B;
```



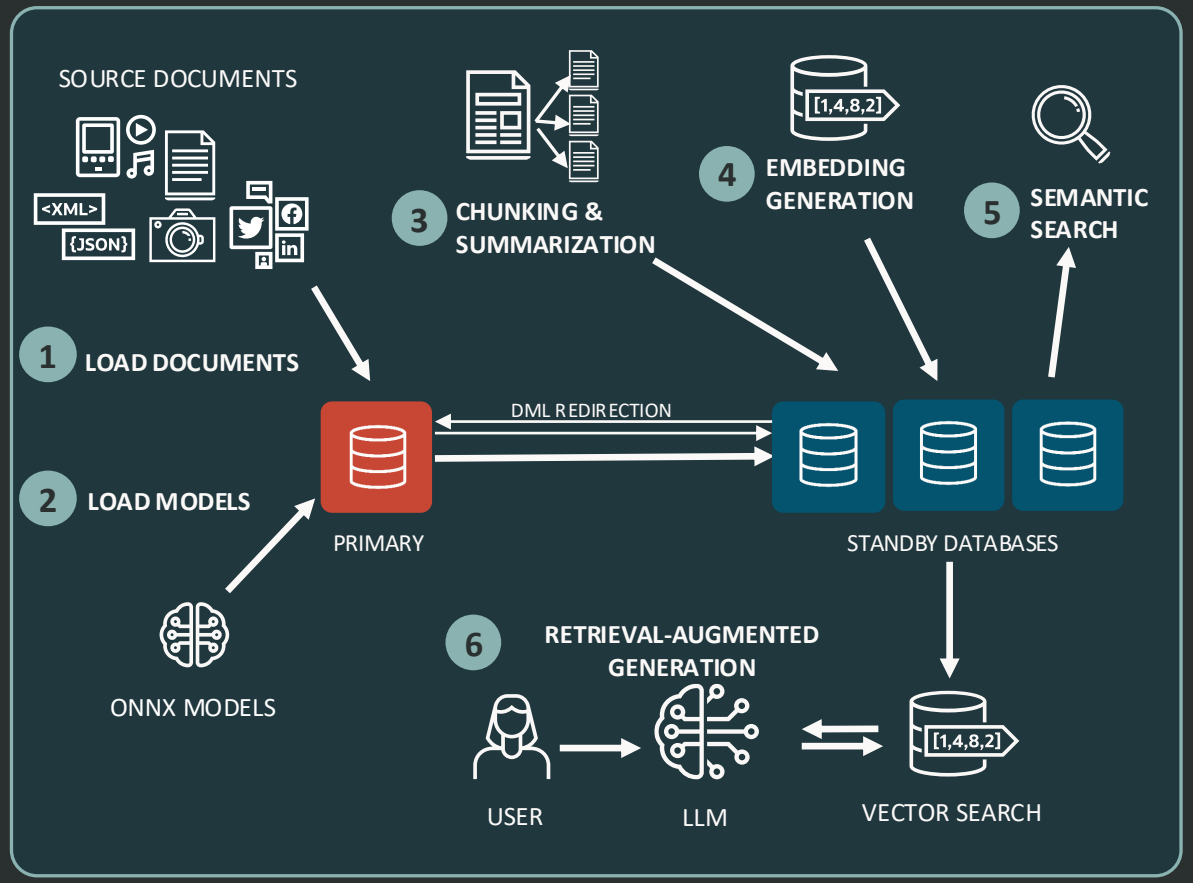
Copyright © 2026, Oracle and/or its affiliates

AI-ready

New In 26ai

Offload AI Inference and Vector Search to Oracle Active Data Guard

Move AI workloads to your standby. Keep the primary focused on core operations



Offload CPU-demanding tasks, including:

- Inference
- Chunking and summarization
- AI Vector Search

- Generate embeddings on the standby database
- Insert vectors into the primary via DML redirection
- Fully utilize standby resources

Primary database performance and reliability remain uncompromised

Copyright © 2026, Oracle and/or its affiliates

New In 26ai

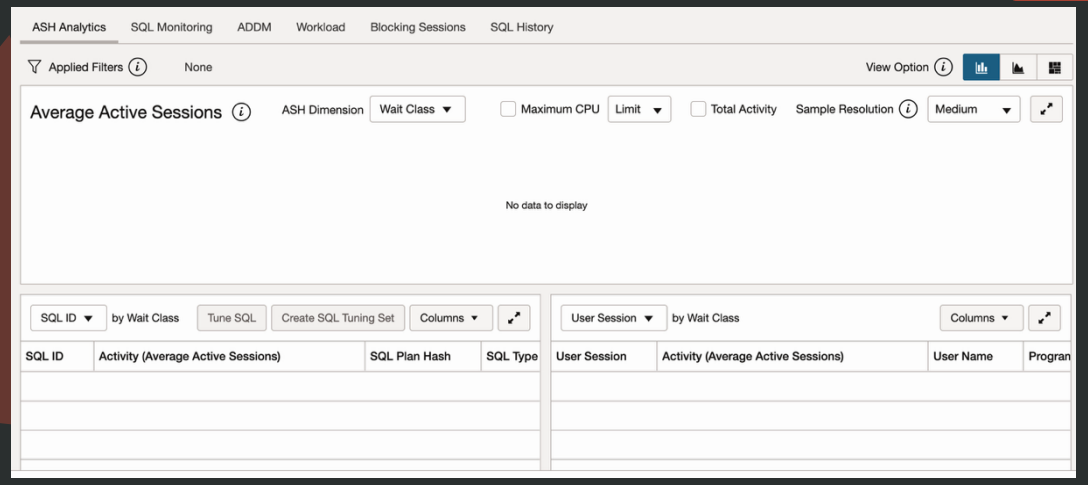
```
create or replace PROCEDURE ADGVEC.process_embeddings (
    p_batch_size IN PLS_INTEGER,
    p_iterations IN PLS_INTEGER
) AS
    TYPE t_embedding IS RECORD (
        id            cats.id%TYPE,
        embed_vector  cats_vec_clipimg.embedding%TYPE
    );

    TYPE t_embedding_table IS TABLE OF t_embedding;

    v_embeddings t_embedding_table;
    v_batch_count PLS_INTEGER := 0;
    v_total_processed PLS_INTEGER := 0;
    v_continue BOOLEAN := TRUE;

    CURSOR c_embeddings IS
        SELECT c.id,
               vector_embedding(clipimg USING img AS data) AS embed_vector
        FROM cats c
        LEFT OUTER JOIN cats_vec_clipimg v ON c.id = v.id
        WHERE v.id IS NULL AND c.img IS NOT NULL and c.id IS NOT NULL;
BEGIN
    OPEN c_embeddings;

    EXECUTE IMMEDIATE 'ALTER SESSION ENABLE ADG_REDIRECT_DML';

    LOOP
        FETCH c_embeddings BULK COLLECT INTO v_embeddings LIMIT p_batch_size;
        EXIT WHEN v_embeddings.COUNT = 0;

        FOR i IN v_embeddings.FIRST .. v_embeddings.LAST LOOP
            DBMS_OUTPUT.PUT_LINE(    'Processing ID: ' || v_embeddings(i).id);
            INSERT INTO cats_vec_clipimg (id, embedding)
            VALUES (v_embeddings(i).id, v_embeddings(i).embed_vector);
        END LOOP;
        COMMIT;

        v_total_processed := v_total_processed + v_embeddings.COUNT;
        v_batch_count := v_batch_count + 1;

        IF p_iterations > 0 AND v_batch_count >= p_iterations THEN
            v_continue := FALSE;
        END IF;

        EXIT WHEN NOT v_continue;
    END LOOP;
```

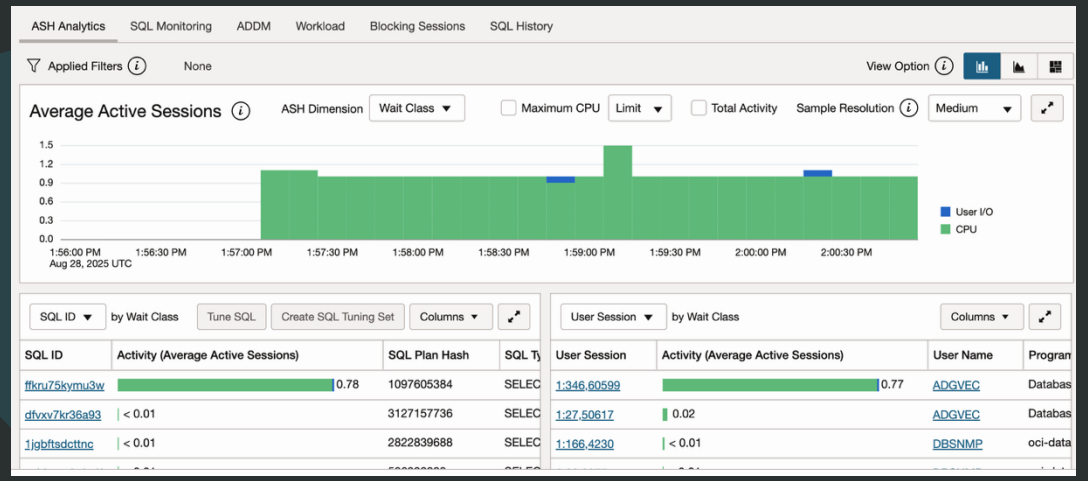


PRIMARY



STANDBY





https://blogs.oracle.com/maa/onnx-on-active-data-guard

Copyright © 2026, Oracle and/or its affiliates

Not "new" but still hot 26ai new features

Oracle Data Guard 26ai new features

New in 21c

Data Guard management from SQLcl

Everything under control with a single command-line tool

```
SQL> help dg
DG
------
 Run DG commands

 DG ADD DATABASE "<database name>" AS CONNECT IDENTIIFIER IS <connect identifier> [ INCLUDE CURRENT DESTINATIONS ];
 DG CREATE CONFIGURATION "<config_name>" AS PRIMARY DATABASE IS <database name> CONNECT IDENTIIFIER IS <connect_identifier>
        [ INCLUDE CURRENT DESTINATIONS ];
 DG DISABLE CONFIGURATION;
 DG DISABLE { DATABASE | RECOVERY_APPLIANCE | FAR_SYNC | MEMBER } <member name>;
 DG EDIT CONFIGURATION SET PROPERTY <property name> = '<property value>';
 DG EDIT { DATABASE | RECOVERY_APPLIANCE | FAR_SYNC | MEMBER } <member name> SET PROPERTY <property name> = '<property value>';
 DG ENABLE CONFIGURATION;
 DG ENABLE { DATABASE | RECOVERY_APPLIANCE | FAR_SYNC | MEMBER } <member name>;
 DG FAILOVER TO <database name> [IMMEDIATE];
 DG REINSTATE DATABASE <database name>;
 DG REMOVE CONFIGURATION [PRESERVE DESTINATIONS];
 DG REMOVE { DATABASE | RECOVERY_APPLIANCE | FAR_SYNC | MEMBER } <name> [PRESERVE DESTINATIONS];
 DG SHOW CONFIGURATION [<property name>];
 DG SHOW DATABASE <database name> [<property name];
 DG SWITCHOVER TO <database name> [WAIT [<timeout in seconds]];
```


Copyright © 2026, Oracle and/or its affiliates

New in 21c

ORDS REST API for Data Guard management

Ready for modern DevOps deployment

```
POST /database/dataguard/configuration/
{
  "primary_connection_identifier": "site1-scan:1521/mydb",
  "primary_database": "mydb_site1"
}

POST /database/dataguard/databases/
{
  "connection_identifier": "site2-scan:1521/mydb",
  "database_name": "mydb_site2"
}

PUT /database/dataguard/configuration/
{
  "operation": "ENABLE"
}
```

Create the configuration

Add the standby databases

Enable the configuration

Oracle REST Data Services API - Data Guard REST Endpoints
https://docs.oracle.com/en/database/oracle/oracle-rest-data-services/25.4/orrst/api-data-guard.html

Copyright © 2026, Oracle and/or its affiliates

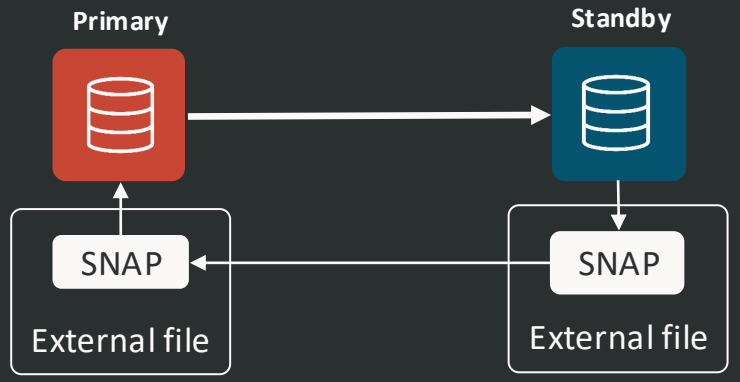
New In 26ai

Simplified AWR snapshots on Active Data Guard

Just create the snapshot on the standby, and you are ready to go.

```
-- at the PDB or CDB level
alter session set container = PDB1;

-- the snapshot can be created without additional configuration
select dbms_workload_repository.create_snapshot from dual;

-- create the report at the PDB or CDB level
@?/rdbms/admin/awrrpti
```

```
-- databases already using SYS$UMF can set this
-- to use the new framework instead:
alter system set "_umf_remote_enabled"=FALSE;
```



Copyright © 2026, Oracle and/or its affiliates

New In 26ai

Switchover and Failover Readiness

Checking if the database is ready for a role transition is as easy as selecting a column

Two new columns, SWITCHOVER_READY and FAILOVER_READY, computed every minute by the broker.

```
SQL> select database, dataguard_role, status, severity, switchover_ready, failover_ready, transport_mode
  2> from v$dg_broker_config;


DATABASE   DATAGUARD_ROLE       STATUS SEVERITY       SWITCHOVER_READY FAILOVER_READY TRANSPORT_MODE
---------- ---------------- ---------- -------------- ---------------- -------------- ---------------
boston     PRIMARY                   0 SUCCESS        YES              UNKNOWN        -N/A-
chicago    PHYSICAL STANDBY          0 SUCCESS        YES              YES            ASYNC
```

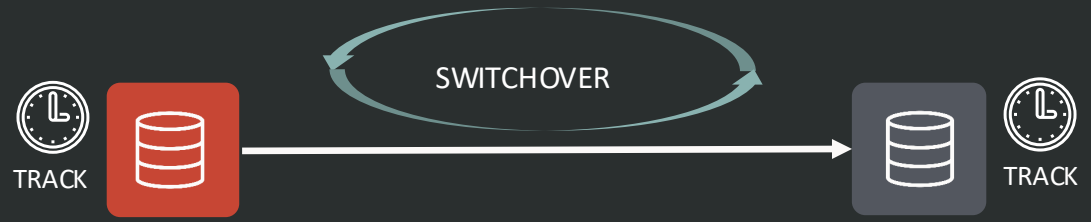
The checks done by the broker are a superset of the ALTER DATABASE SWITCHOVER VERIFY command:

```
SQL> alter database switchover to chicago verify;
alter database switchover to chicago verify
*
ERROR at line 1:
ORA-16470: Redo Apply is not running on switchover target
```


Copyright © 2026, Oracle and/or its affiliates

Easier tracking of role transitions

New In 26ai



The new fixed view **V$DG_BROKER_ROLE_CHANGE** tracks the last 10 role transitions

```
SQL> select * from V$DG_BROKER_ROLE_CHANGE;

EVENT              STANDBY_TYPE OLD_PRIMARY NEW_PRIMARY FS_FAILOVER_REASON   BEGIN_TIME          END_TIME
------------------ ------------ ----------- ----------- -------------------- ------------------- -------------------
Failover           Physical     mydb1       mydb1b      Manual Failover      30-SEP-2024 19:01:14 30-SEP-2024 19:01:35
Switchover         Physical     mydb1b      mydb1                            30-SEP-2024 19:04:53 30-SEP-2024 19:05:15
Switchover         Physical     mydb1       mydb1b                           30-SEP-2024 20:51:38 30-SEP-2024 20:52:03
Failover           Physical     mydb1b      mydb1       Manual Failover      30-SEP-2024 20:52:46 30-SEP-2024 20:53:04
Switchover         Physical     mydb1       mydb1c                           30-SEP-2024 19:53:14 30-SEP-2024 19:54:14
Switchover         Physical     mydb1c      mydb1                            30-SEP-2024 20:03:14 30-SEP-2024 20:04:04
Switchover         Logical      mydb1       mydb1d                           30-SEP-2024 20:24:46 30-SEP-2024 20:26:32
Switchover         Logical      mydb1d      mydb1                            30-SEP-2024 20:35:27 30-SEP-2024 20:35:48
Fast-Start Failover Physical    mydb1       mydb1b      Primary Disconnected 30-SEP-2024 20:13:51 30-SEP-2024 20:14:53
```

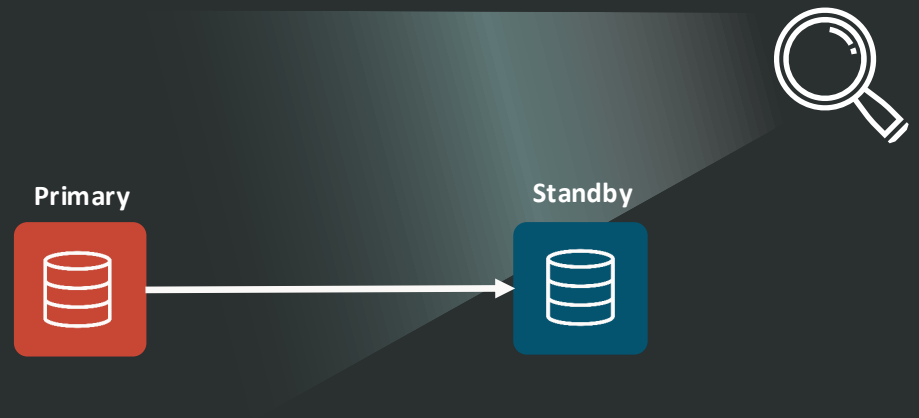

Copyright © 2026, Oracle and/or its affiliates

Easier checking of Data Guard configurations

New In 26ai

The new fixed view **V$DG_BROKER_PROPERTY** contains the properties of the configuration and all the members

```
SQL> select member, property, value from V$DG_BROKER_PROPERTY where value is not null;

MEMBER      PROPERTY                       VALUE
----------- ------------------------------ --------------------
mydb        FastStartFailoverThreshold     180
mydb        OperationTimeout               30
mydb        TraceLevel                     USER
mydb        FastStartFailoverLagLimit      300
mydb        CommunicationTimeout           180
mydb        ObserverReconnect              0
mydb        ObserverPingInterval           0
mydb        ObserverPingRetry              0
mydb        FastStartFailoverAutoReinstate TRUE
mydb        FastStartFailoverPmyShutdown   TRUE
...
mydb_site1  DGConnectIdentifier            mydb_site1
mydb_site1  FastStartFailoverTarget        mydb_site2
mydb_site1  LogShipping                    ON
mydb_site1  LogXptMode                     ASYNC
mydb_site1  DelayMins                      0
...
mydb_site1  StaticConnectIdentifier        (DESCRIPTION=<...>)))
mydb_site1  TopWaitEvents                  (monitor)
mydb_site1  SidName                        (monitor)
mydb_site2  DGConnectIdentifier            mydb_site2
mydb_site2  FastStartFailoverTarget        mydb_site1
...

66 rows selected.
```



Copyright © 2026, Oracle and/or its affiliates


ORACLE