ORACLE
CloudWorld

Achieving Continuous Availability for Your Applications with Oracle MAA

October 17–20, 2022

Caesars Forum and The Venetian

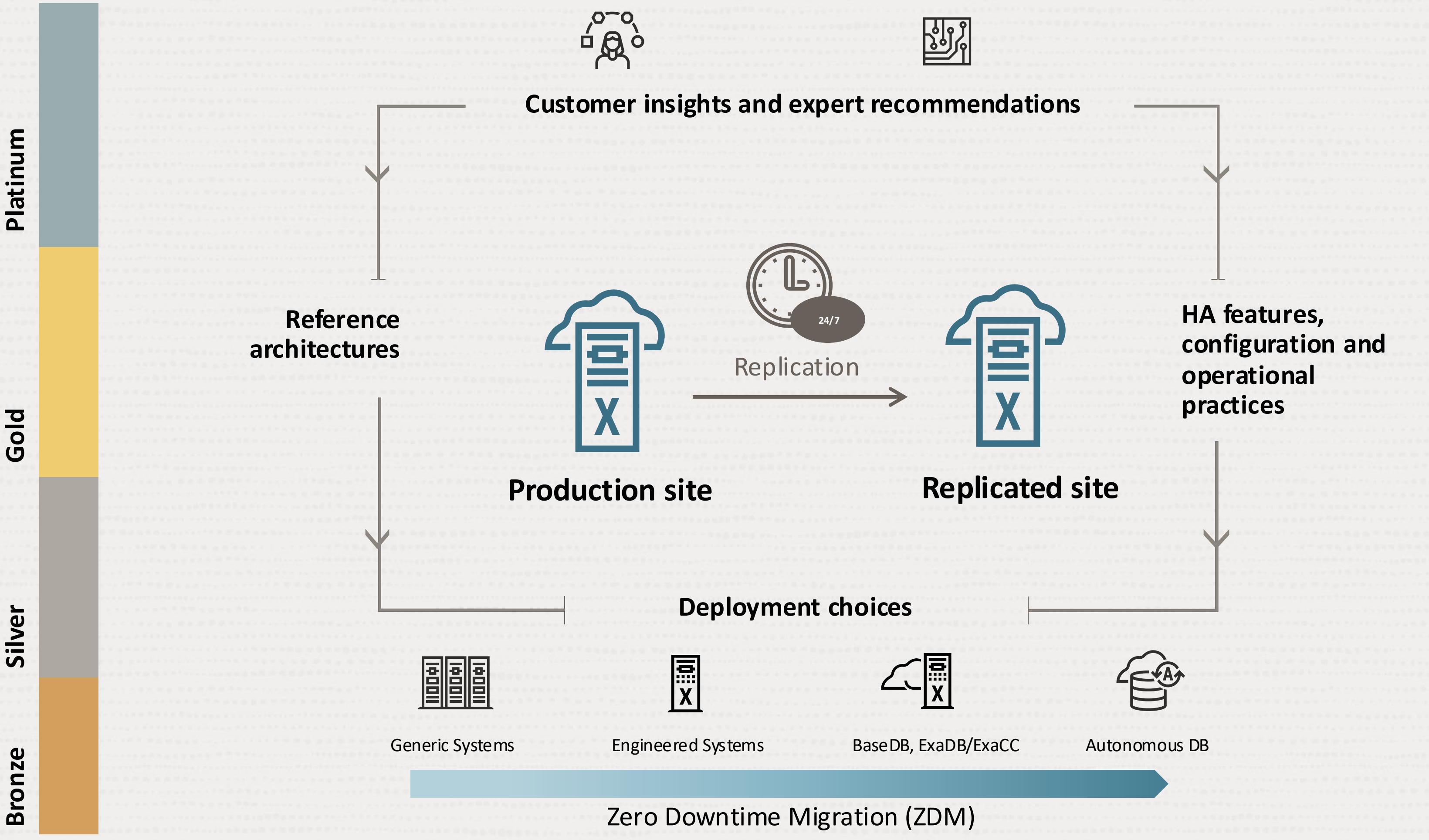
Las Vegas, NV

Topics Covered in this Presentation

1. Why Transparent Application Continuity (TAC) should be your default choice
2. High Availability requirements to ensure Business Continuity
3. How to efficiently use TAC for planned maintenance and unplanned outages
4. What to do when Transparent Application Continuity does not fit
5. Customer Example

Oracle Maximum Availability Architecture (MAA)

Standardized Reference Architectures for Never-Down Deployments



Customer insights and expert recommendations

Reference architectures

Production site

Replication

Replicated site

HA features, configuration and operational practices

Deployment choices

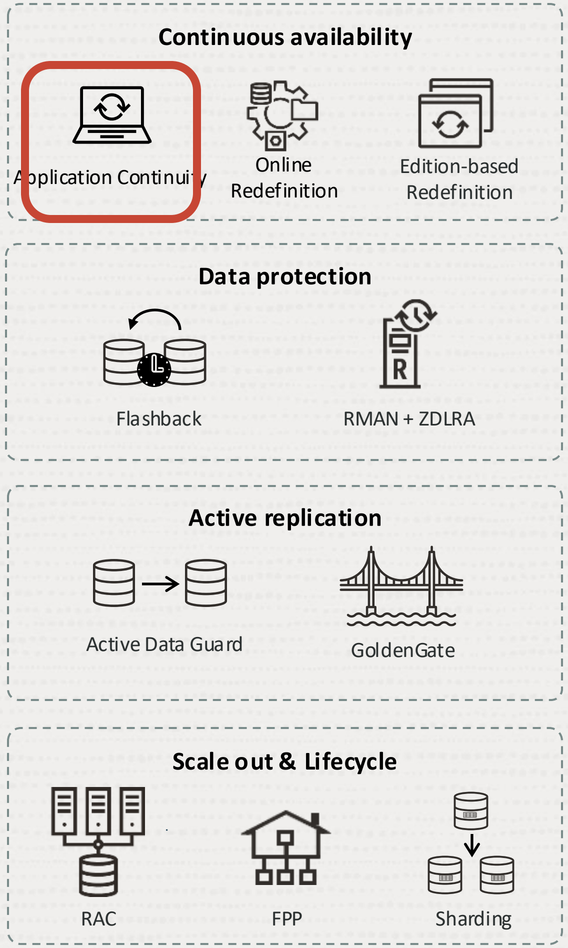
Generic Systems

Engineered Systems

BaseDB, ExaDB/ExaCC

Autonomous DB

Zero Downtime Migration (ZDM)



Continuous availability

- Application Continuity
- Online Redefinition
- Edition-based Redefinition

Data protection

- Flashback
- RMAN + ZDLRA

Active replication

- Active Data Guard
- GoldenGate

Scale out & Lifecycle

- RAC
- FPP
- Sharding

From Maximum Availability to Business Continuity

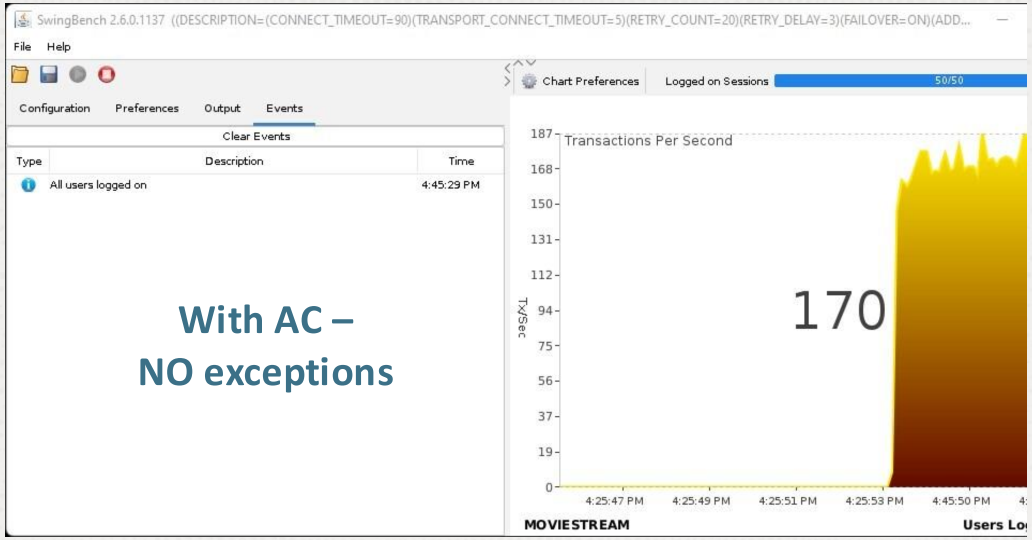
Maximum Availability

- Minimizes downtime
- In-flight work is lost
- Rolling maintenance on DB-level
- Predictable runtime performance
- Errors may be visible
- High Availability building blocks

Business Continuity

- ***Eliminates downtime for users***
- In-flight work is preserved
- ***Maintenance is hidden***
- Predictable performance
- ***Errors visible only if unrecoverable***
- Builds on top of High Availability

Application Continuity (AC) in Simple Terms

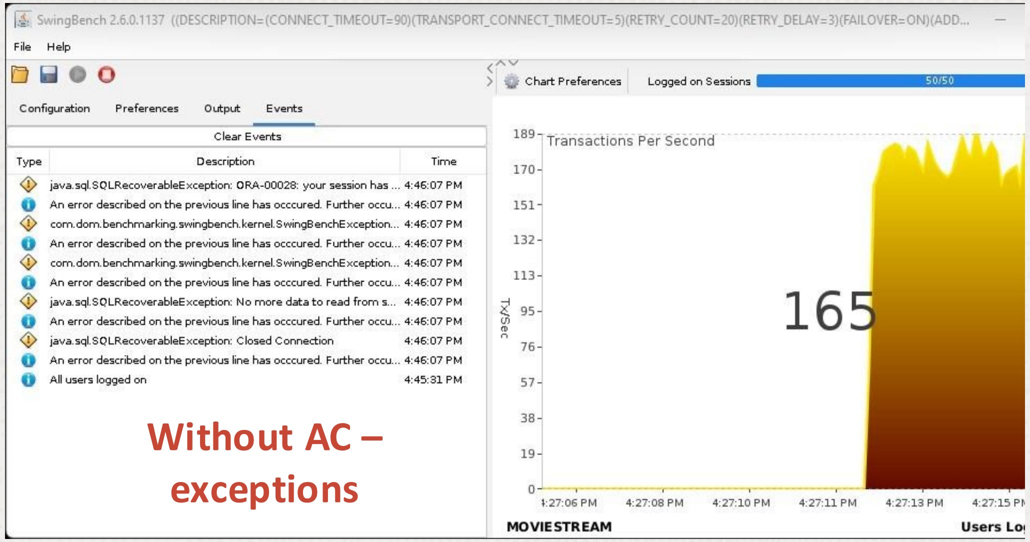


SwingBench 2.6.0.1137 ((DESCRIPTION=(CONNECT_TIMEOUT=90)(TRANSPORT_CONNECT_TIMEOUT=5)(RETRY_COUNT=20)(RETRY_DELAY=3)(FAILOVER=ON)(ADD...

Type	Description	Time
ℹ	All users logged on	4:45:29 PM

Logged on Sessions 50/50

With AC –
NO exceptions



SwingBench 2.6.0.1137 ((DESCRIPTION=(CONNECT_TIMEOUT=90)(TRANSPORT_CONNECT_TIMEOUT=5)(RETRY_COUNT=20)(RETRY_DELAY=3)(FAILOVER=ON)(ADD...

Type	Description	Time
⚠	java.sql.SQLRecoverableException: ORA-00028: your session has ...	4:46:07 PM
ℹ	An error described on the previous line has occcured. Further occu...	4:46:07 PM
⚠	com.dom.benchmarking.swingbench.kernel.SwingBenchException...	4:46:07 PM
ℹ	An error described on the previous line has occcured. Further occu...	4:46:07 PM
⚠	com.dom.benchmarking.swingbench.kernel.SwingBenchException...	4:46:07 PM
ℹ	An error described on the previous line has occcured. Further occu...	4:46:07 PM
⚠	java.sql.SQLRecoverableException: No more data to read from s...	4:46:07 PM
ℹ	An error described on the previous line has occcured. Further occu...	4:46:07 PM
⚠	java.sql.SQLRecoverableException: Closed Connection	4:46:07 PM
ℹ	An error described on the previous line has occcured. Further occu...	4:46:07 PM
ℹ	All users logged on	4:45:31 PM

Logged on Sessions 50/50

Without AC –
exceptions

One Solution – Two Flavors

Application Continuity (AC)

- For planned maintenance and unplanned outages
- Available with Oracle RAC and Active Data Guard
- Requires Oracle connection pool

Transparent Application Continuity (TAC)

- For planned maintenance and unplanned outages
- Available with Oracle RAC and Active Data Guard
- Default on Oracle Autonomous Database

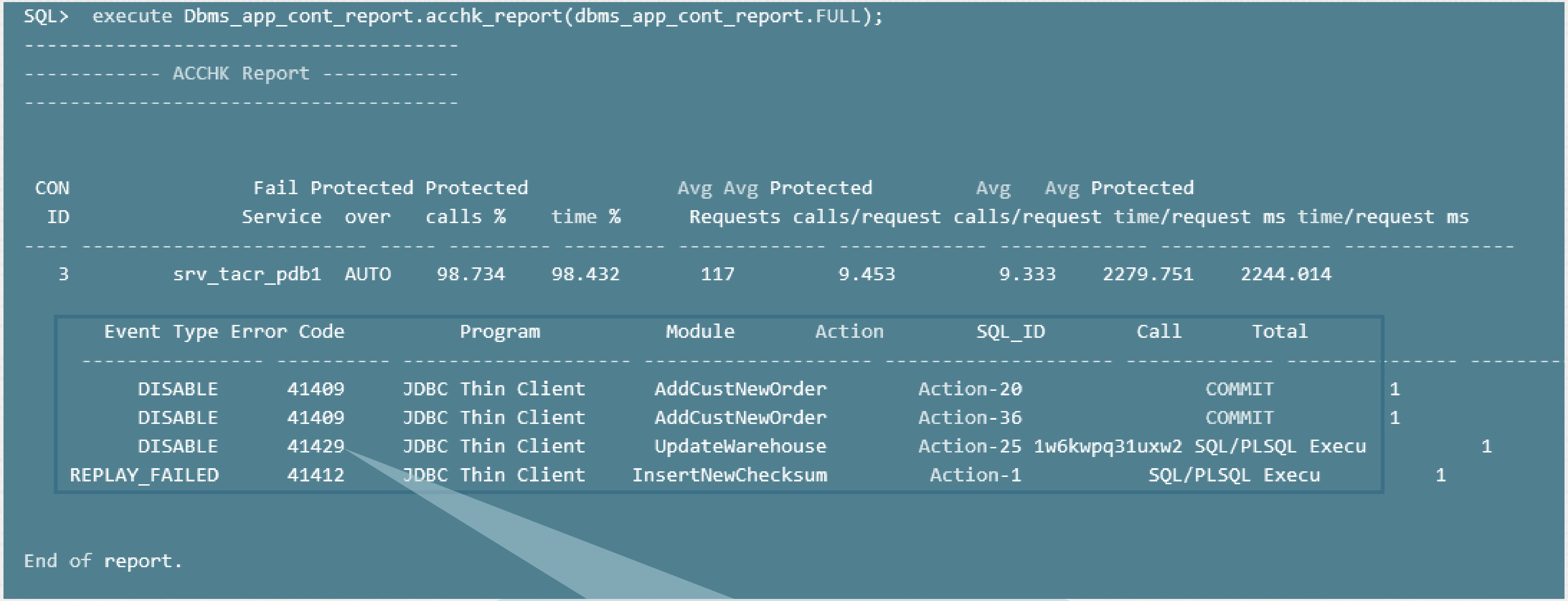
Transparent Application Continuity

Default on Oracle Autonomous Database – Why?

- Based on the same HA requirements as AC, TAC covers most SQL use cases:
  - TAC works with and without Oracle connection pools.
  - ACchk can be used to determine the protection provided.
  - Oracle 19c Databases and Clients and later provide best results.
- High Availability (HA) requirements include:
  - Use of Oracle Database services.
  - Using an optimized TNS connection string.
  - Draining is part of maintenance operations.
  - Active Data Guard for Disaster Recovery.
- TAC seamlessly improves business continuity on Autonomous Database
  - Using the same standards, TAC can help improve your application availability, too.

Determine the Protection Provided by TAC with ACchk

And catch applications that use coding practices that prevent safe replays



```
SQL>  execute Dbms_app_cont_report.acchk_report(dbms_app_cont_report.FULL);
--------------------------------------
------------ ACCHK Report ------------
--------------------------------------



 CON                Fail Protected Protected             Avg Avg Protected         Avg   Avg Protected
  ID               Service  over   calls %    time %      Requests calls/request calls/request time/request ms time/request ms
---- ------------------------- ----- --------- --------- ------------- ------------- ------------- --------------- ---------------
   3         srv_tacr_pdb1  AUTO    98.734    98.432       117         9.453         9.333    2279.751    2244.014

       Event Type Error Code          Program            Module       Action        SQL_ID        Call      Total
     ---------------- ---------- -------------------- -------------------- -------------------- ------------- --------------- --------
          DISABLE      41409     JDBC Thin Client      AddCustNewOrder        Action-20                COMMIT          1
          DISABLE      41409     JDBC Thin Client      AddCustNewOrder        Action-36                COMMIT          1
          DISABLE      41429     JDBC Thin Client      UpdateWarehouse        Action-25 1w6kwpq31uxw2 SQL/PLSQL Execu         1
    REPLAY_FAILED      41412     JDBC Thin Client    InsertNewChecksum         Action-1           SQL/PLSQL Execu       1


End of report.
```

Disable Reason
ORA-41429: side effect detected

Business Continuity

High Availability Requirements

Use Database Services

- Always use user-defined database services to connect with any database/PDB

```
$ srvctl add service -d <DATABASE NAME> -s <SERVICE NAME> [-pdb <PDB NAME>]
                      ⎧ -preferred <INSTANCE NAME, …> -available <INSTANCE NAME, …>
 RAC HA               ⎨ -notification true
                      ⎩ -reset_state level1 [21c+]
                      ⎧ -failover_restore AUTO
 Application          ⎪ -commit_outcome TRUE
 Continuity (AC)      ⎨ -failovertype AUTO
                      ⎩ -replay_init_time 600
 Workload             ⎧ -drain_timeout 300
 Draining             ⎨ -stopoption IMMEDIATE
                      ⎩ -role PRIMARY
```

- **Never use the default service**

Example: Autonomous Pre-Created Services

TYPE	NONTLS (ADB-D)	TLS NAME	STANDBY (ADB-D)	TLS STANDBY (ADB-D)
High priority OLTP [1]	tpurgent	tpurgent_tls	tpurgent_ro	tpurgent_ro_tls
Typical OLTP [1]	tp	tp_tls	tp_ro	tp_ro_tls
High priority Reporting [2]	high	high_tls	high_ro	high_ro_tls
Typical Reporting [2]	medium	medium_tls	medium_ro	medium_ro_tls
Low priority Reporting [2]	low	low_tls	low_ro	low_ro_tls

[1] Transparent Application Continuity enabled by default

[2] Use DBMS_APP_CONT_ADMIN.ENABLE_TAC to enable TAC, or
Use DBMS_APP_CONT_ADMIN.ENABLE_AC to enable AC

Connection Best Practices

- Connecting to a custom database service for a “sales application”
- Full Description:

```
jdbc:oracle:thin:@
(DESCRIPTION =
 (CONNECT_TIMEOUT=90) (TRANSPORT_CONNECT_TIMEOUT=3)
 (RETRY_COUNT=50) (RETRY_DELAY=3) (FAILOVER=ON)
 (ADDRESS_LIST=
  (LOAD_BALANCE=on)
  (ADDRESS=(PROTOCOL=tcp)(HOST=CLOUD-SCANVIP.example.com)(PORT=1522)))
 (CONNECT_DATA=(SERVICE_NAME=Sales_svc)))
```

Configure Fast Application Notification and use Draining

Fast Application Notification (FAN)

- Breaks applications out of TCP timeouts.
- Notifies clients of status changes on the database service level via ONS.
- The Oracle Notification Server (ONS) is automatically configured.

Draining describes

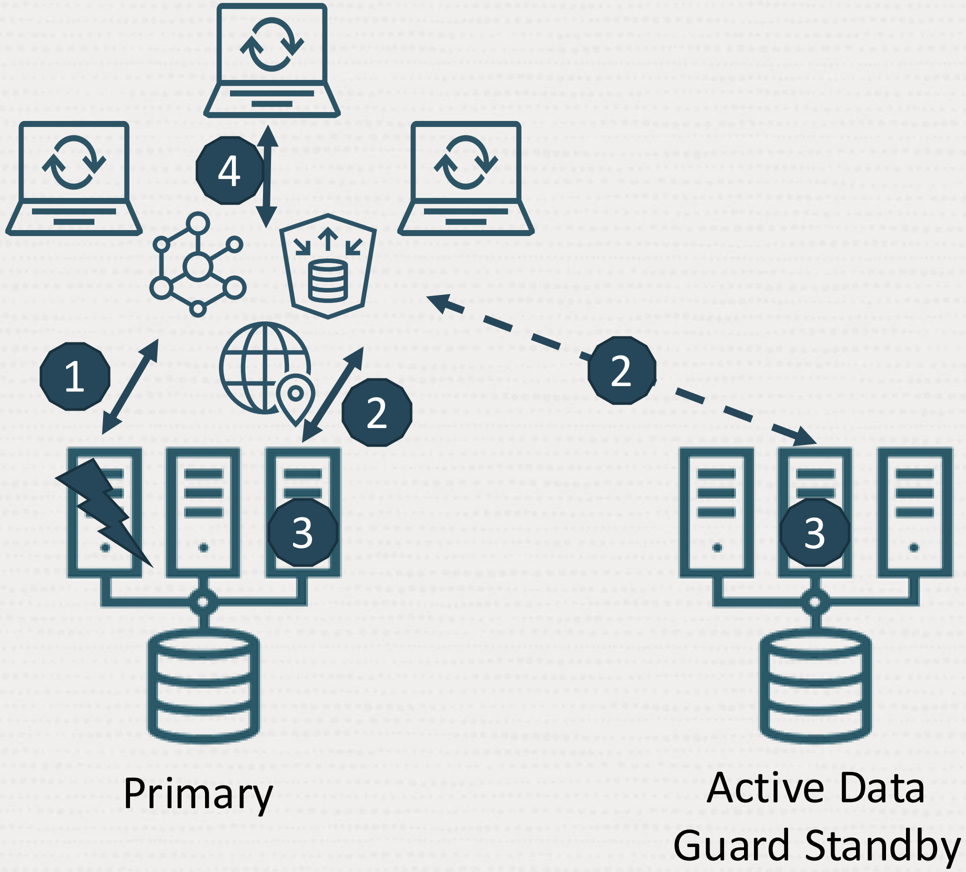
- an operation that causes sessions to complete their work on a given instance
- to prepare the node or the database for maintenance.

Draining is a function of the Oracle Database, supported by Oracle drivers, connection pools, and natively by SQL.

Planned and Unplanned Outages

With Transparent Application Continuity

Unplanned Outages and Planned Failover



- Actions on database level:
  1. DB Request interrupted by an error or FAN or drain
  2. Session reconnects to the available service
  3. Database (DB) Request replays automatically
  4. Result from Database Request returned to user

- (Transparent) Application Continuity
  - Fails over within SLA's
  - Guarantees commit at most once
  - Verifies:
    - Correct database
    - Correct session state
    - Correct results

What to do when Transparent Application Continuity does not fit

Oracle Database 19c – Offering Two Flavors

	TAC – simple apps	AC – pooled apps
Oracle or 3rd party connection pools with request boundaries	Yes, if no unrestorable state*	Yes
Non-pooled or no request boundaries	Yes, if no cursors in fetch and no unrestorable state*	No
Reenables after COMMIT or other disabling call	Yes, if no cursors in fetch and no unrestorable state*	No, until request boundary
SQL, Transactions & Queries	Yes	Yes
PL/SQL	Yes, disables side effects	Yes, allows side effects
Restores state at start of replay	Yes	Yes
Restores mutables	Yes	Yes
Replays over DBlinks	No	Yes
Replays when client state and results match	Yes, server state must also match	Yes

*Unrestorable state examples: PL/SQL global session state, temp tables, session lobs, OJVM

Which Flavor to Use

- Use Transparent Application Continuity (TAC) for applications
  - Using INSERT, UPDATE, DELETE, SELECT, ALTER SESSION – basic ANSI SQL calls
  - That may or may not use Oracle connection pools
  - That can accept that a replay of side effects is disabled automatically
  - Not requiring transactions over database links to replay

- Use Application Continuity (AC) for applications
  - Using SQL and PL/SQL global session state, temp tables, session LOBs
  - Can use an Oracle connection pool or other compliant pools (RedHat JBoss)
  - That need replay of side effects enabled (such as email, audit, logging, file transfer)
  - Need database link support

Application Continuity for Packaged Applications

- Packaged applications, from Oracle and other vendors, pose special challenges
  - What has the vendor tested regarding workload draining and Application Continuity?
  - Certification and support will constrain what versions and configuration options are available
  - Often designed for older technology stacks, limiting what can be achieved
- Oracle & other packaged applications are adopting draining and Application Continuity
  - E-Business Suite – support for workload draining with SQL connection tests (under development)
  - JD Edwards – support for workload draining (TAC replay planned to be available soon)
  - Fusion Middleware – support for workload draining (GA) and Application Continuity (under development)
  - PeopleSoft – supports draining and TAC replay
  - Siebel – has its own replay mechanism
  - SAP – support for workload draining (TAC replay support planned to be available soon)

Customer Example

Oracle Application Continuity in Gaia Oracle Service

Improving app resilience across the firm by simplifying resilient design for our application engineers

Madhu Bhimaraju
Executive Director
Head of Development & Modernization
Consumer & Community Banking (Chase)

Doug Burns
Vice President
Oracle Engineering Lead
GTI Platform Services

Challenges & Solutions

Planned Outages

- Connection Pool Draining, FAN & TAF
- Drain connection pools and allow transactions to complete within application-specific timeframes.
- Transactions that don't complete within the timeout are rolled back. **Apps need to detect and replay.**

Unplanned Outages

- In-flight Transactions are *always* rolled back to preserve database integrity. **Apps need to detect and replay.**

Oracle 19c Transparent Application Continuity

- Automatically tracks in-flight transactions and **replays them** when service is restored.
- Zero/Low Dev Effort. Other than connecting to a different Service we provide; possibly defining transaction boundaries; and using a different driver - no detection/recovery code required.
- Modern App Design is still relevant. Works best with connection pools and well-designed apps that borrow connections, execute one transaction then return to the pool.

Gaia Oracle Service

(GOS)

1

Private Oracle Cloud

Opinionated. Rapid Provisioning. No-DBA model.

2

Customers

Strategic platform for internal Oracle databases
60-70% cloud v. non-cloud in 3 years and on a great trajectory
Critical zero-downtime applications e.g. Debit, Payments etc.,

3

MAA

Closely follows detailed MAA recommendations
All Production databases are RAC clusters and have at least one ADG Standby.

4

Security & Hygiene

Mandatory application of Release Updates.
Opinionated security model.

5

Self-Service

… means self-service! Full Repave capabilities, Rolling Patching, Switchover/Failover, Create/Stop/Start Services.

Key Lessons

1

Application Feature

Application, Application Server configuration and great software engineers.
For the DBAs in the audience … this requires *collaboration.*

2

Client Versions

Non-negotiable and not a significant issue for us.

3

Testing

There is play-testing/PoC and real testing.
How do you recreate all possible scenarios?

4

Sense of wonder!

Once all the guidance has been followed and requirements met, it's an amazing feature!

5

Information Sources

Google Search is not your friend. Best sources of information are from the Oracle Product Team and are all available at "Oracle Application Continuity"

ORACLE
CloudWorld

Thank you

Application Continuity Best Practices