ORACLE

Performance validation for a successful Oracle Database 23ai upgrade

Real-world best practices and United Airlines case study

Max Szulya
Senior Manager
United Airlines

Sunil Kalra
Principal DBA
United Airlines

Anusha Vojjola
Senior Product Manager
Oracle

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Agenda

- Database Upgrade Challenges
- Prepare to Upgrade to Database 23ai
- Validate Performance During Upgrade
- United Airlines Case Study

Database Upgrade Challenges

How can I identify the impact of updating initialization parameters as part of the upgrade?

How can I identify SQL regression caused by plan changes during an upgrade?

How can I identify the impact of updating underscore parameters as part of the upgrade?

How can I perform load and throughput testing to identify performance issues before putting changes in production?

How can I make sure the installed release or release update don't introduce new regression on my application?

How can I make sure that installed patches solves the identified problem it is intended to do?

Prepare to Upgrade to Database 23ai

Database source & destination: upgrade path

Two-step process

Oracle Database 11.2.0.4
Oracle Database 12.1.0.2
Oracle Database 12.2.0.1
Oracle Database 18c

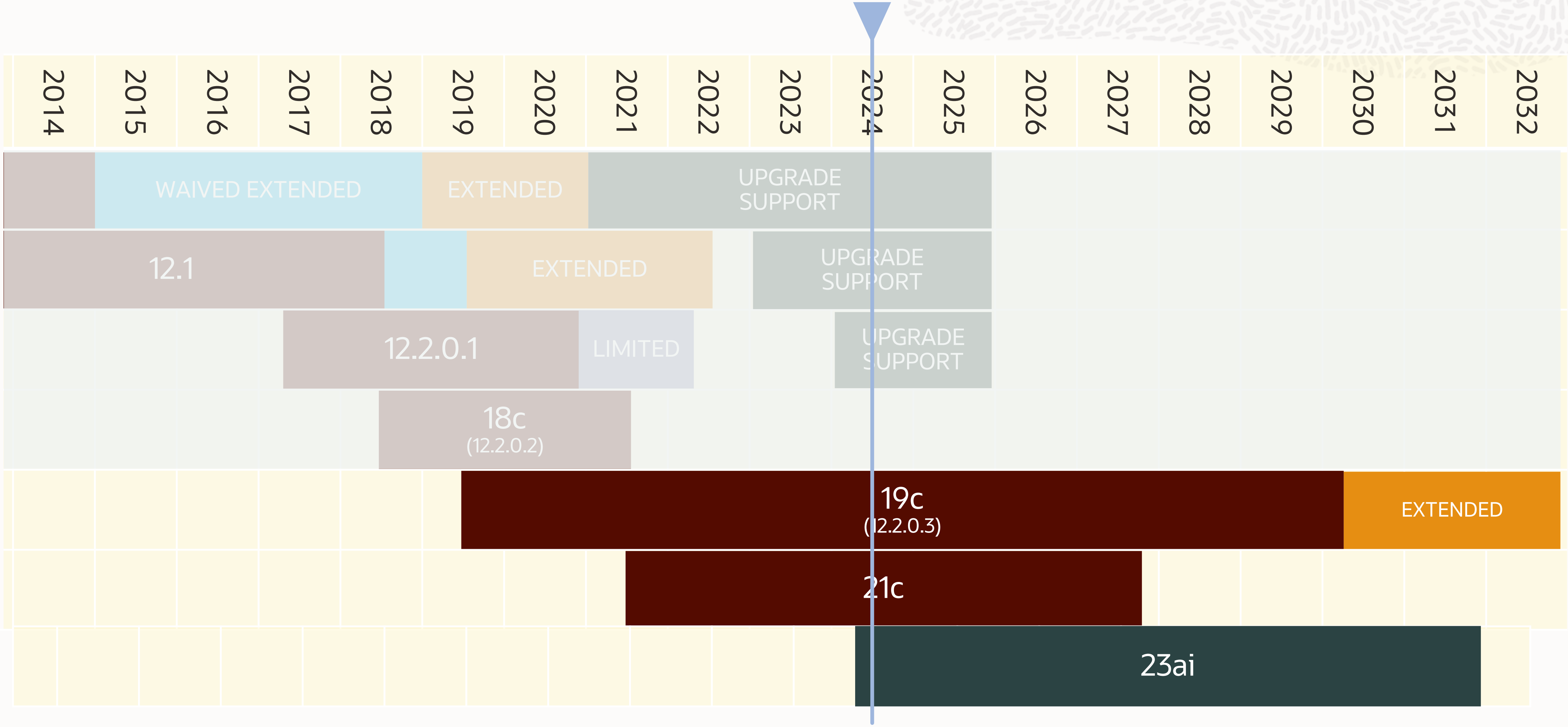
⇛ Oracle Database 19c ⇛ **Oracle Database 23ai**

One-step process

Oracle Database 19c
Oracle Database 21c

⇛ **Oracle Database 23ai**

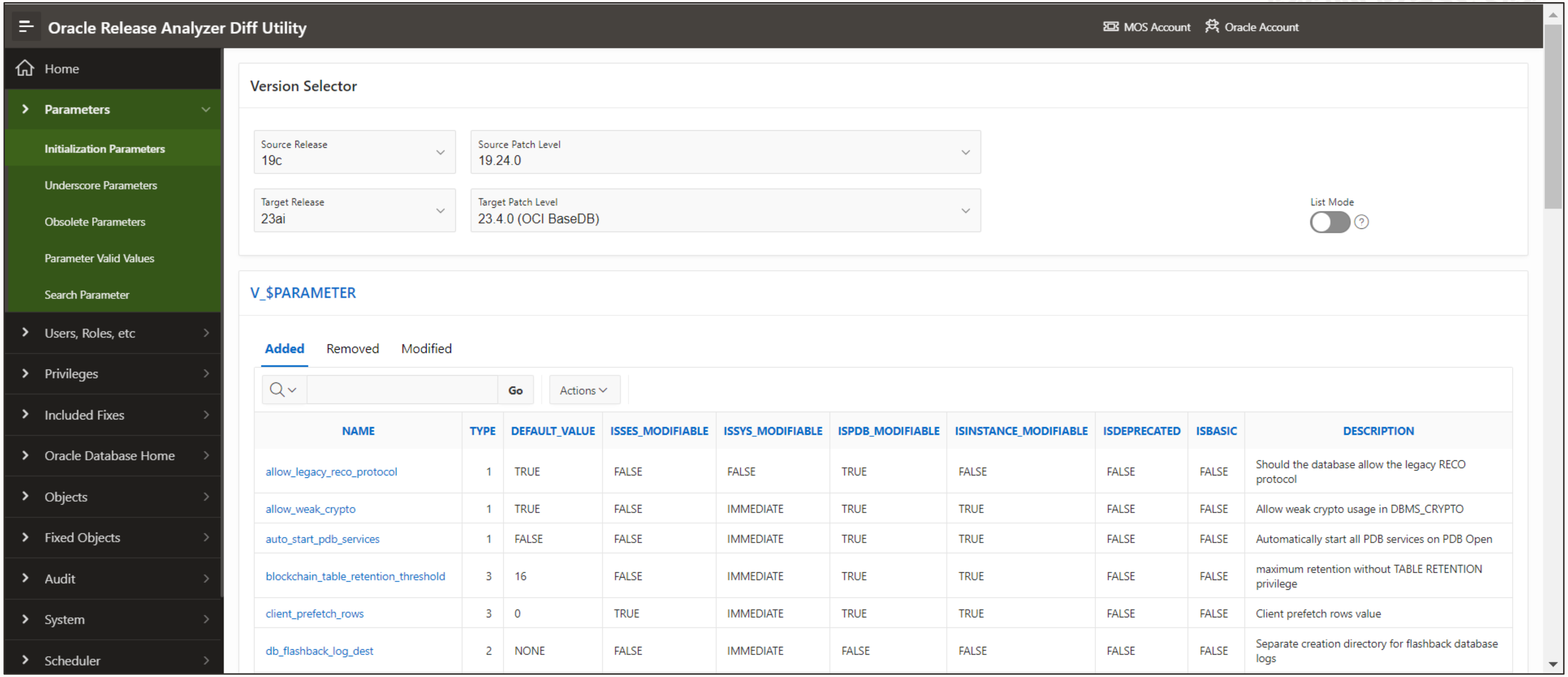
Database lifetime support policy



MOS: Doc ID 742060.1

What changed in Oracle Database 23ai?

init.ora, privileges etc,. use ORAdiff tool - http://oradiff.oracle.com



Oracle Release Analyzer Diff Utility

MOS Account
Oracle Account

Home
Parameters
Initialization Parameters
Underscore Parameters
Obsolete Parameters
Parameter Valid Values
Search Parameter
Users, Roles, etc
Privileges
Included Fixes
Oracle Database Home
Objects
Fixed Objects
Audit
System
Scheduler

Version Selector

Source Release: 19c
Source Patch Level: 19.24.0
Target Release: 23ai
Target Patch Level: 23.4.0 (OCI BaseDB)
List Mode

V_$PARAMETER

Added | Removed | Modified

Go
Actions

NAME	TYPE	DEFAULT_VALUE	ISSES_MODIFIABLE	ISSYS_MODIFIABLE	ISPDB_MODIFIABLE	ISINSTANCE_MODIFIABLE	ISDEPRECATED	ISBASIC	DESCRIPTION
allow_legacy_reco_protocol	1	TRUE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	Should the database allow the legacy RECO protocol
allow_weak_crypto	1	TRUE	FALSE	IMMEDIATE	TRUE	TRUE	FALSE	FALSE	Allow weak crypto usage in DBMS_CRYPTO
auto_start_pdb_services	1	FALSE	FALSE	IMMEDIATE	TRUE	TRUE	FALSE	FALSE	Automatically start all PDB services on PDB Open
blockchain_table_retention_threshold	3	16	FALSE	IMMEDIATE	TRUE	TRUE	FALSE	FALSE	maximum retention without TABLE RETENTION privilege
client_prefetch_rows	3	0	TRUE	IMMEDIATE	TRUE	TRUE	FALSE	FALSE	Client prefetch rows value
db_flashback_log_dest	2	NONE	FALSE	IMMEDIATE	FALSE	FALSE	FALSE	FALSE	Separate creation directory for flashback database logs

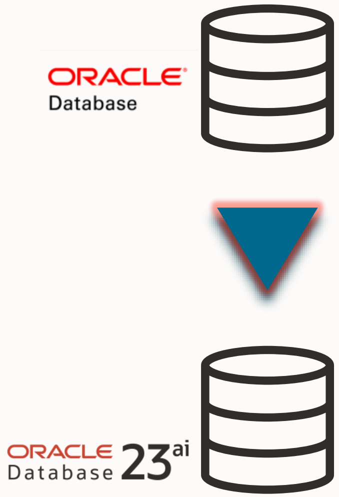
Best Practices

Convert Non-Multitenant database to Multitenant

- Use Enterprise Manager Migration Workbench to convert to multitenant (video link)

Remap workload connections to appropriate PDB by using

- Connection override, connector descriptor or TNS entry
- For RAC, use service load balancing



Ensure your clients can connect to Oracle Database 23ai

- Upgrade your clients well in advance of the upgrade (MOS DOC ID 207303.1)

Execute a dictionary check before upgrade

- Formerly known as Health Check (MOS Doc ID 136697.1)

If required, gather dictionary and fixed objects statistics to avoid potential downtime

- Using dbms_stats procedures: gather_schema_stats('SYS'), gather_schema_stats('SYSTEM'), gather_fixed_objects_stats;

Best Practices

Setup test system - Clone the production database

- At least full dataset

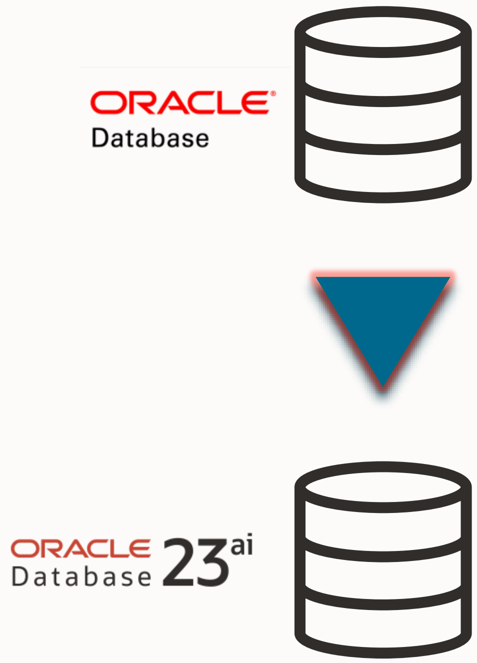
Use harware resources (memory, network, IO, CPU) similar to production

- 20-25% customers make this mistake

Backup strategy in place to restore to prior point-in-time

- Leverage standby databases/GoldenGate if possible
- Any support mechanism RMAN, Datapump, etc.

Use Flashback and Guaranteed Restore Points (GRP) for iterative testing



Validate Performance During Upgrade

Real Application Testing solution

SPA Quick Check

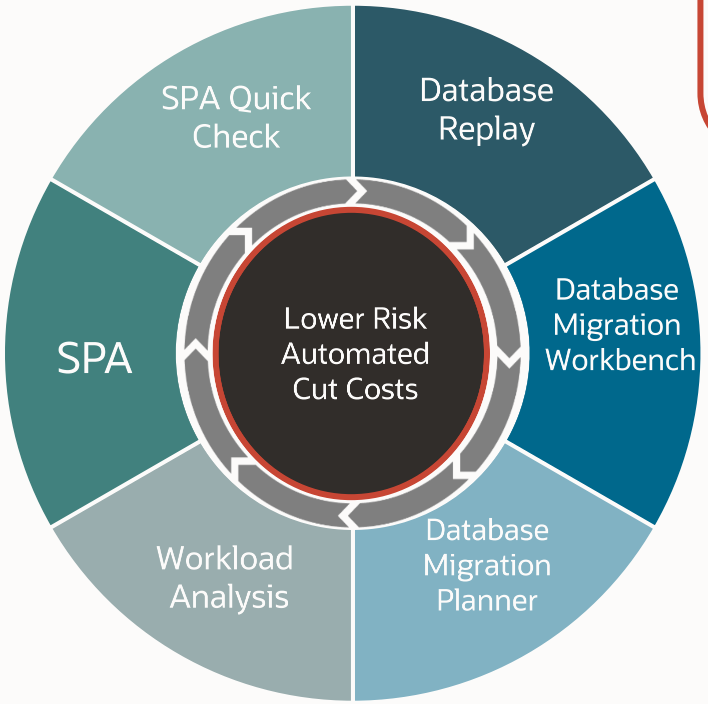
Proactively identifies impact of routine DBA tasks on database performance

SPA 1

SQL response time (unit) testing solution
Designed for identifying and tune regressed SQLs

Automatic Workload Analysis

Identifies impact and root cause of workload changes like changed execution plans, increased volumes, new SQL statements



Database Replay 2

Load and throughput testing solution
Designed for detecting and remediating throughput problems

Database Migration Workbench

Identifies the optimal migration strategy
Execute the migration of your databases

Database Migration Planner

Migration & consolidation simulation & capacity planning solution

Best Practices

Establish Performance Baseline

Capture important workload into SQL Tuning Set

- Split large SQL Tuning set, maximum of 5K statements in one STS

Transport STS and workloads to the test systems

- Use filters to remove noise (e.g., SYS, DBSNMP, scheduler, recursive SQL, etc.)
- Disable maintenance windows, backups, etc. on test system

Use Real Application Testing for highest quality testing

- SQL Performance Analyzer (SPA)
- DB Replay

Review SPA restrictions Distributed queries, complex datatypes

- Note we do test SQL within the PL/SQL not the PLSQL call
- DML – select part of the query (access path) and it is rollbacked by default at end of SQL execution
- SQL Performance Analyzer (SPA) does not change the state of the database

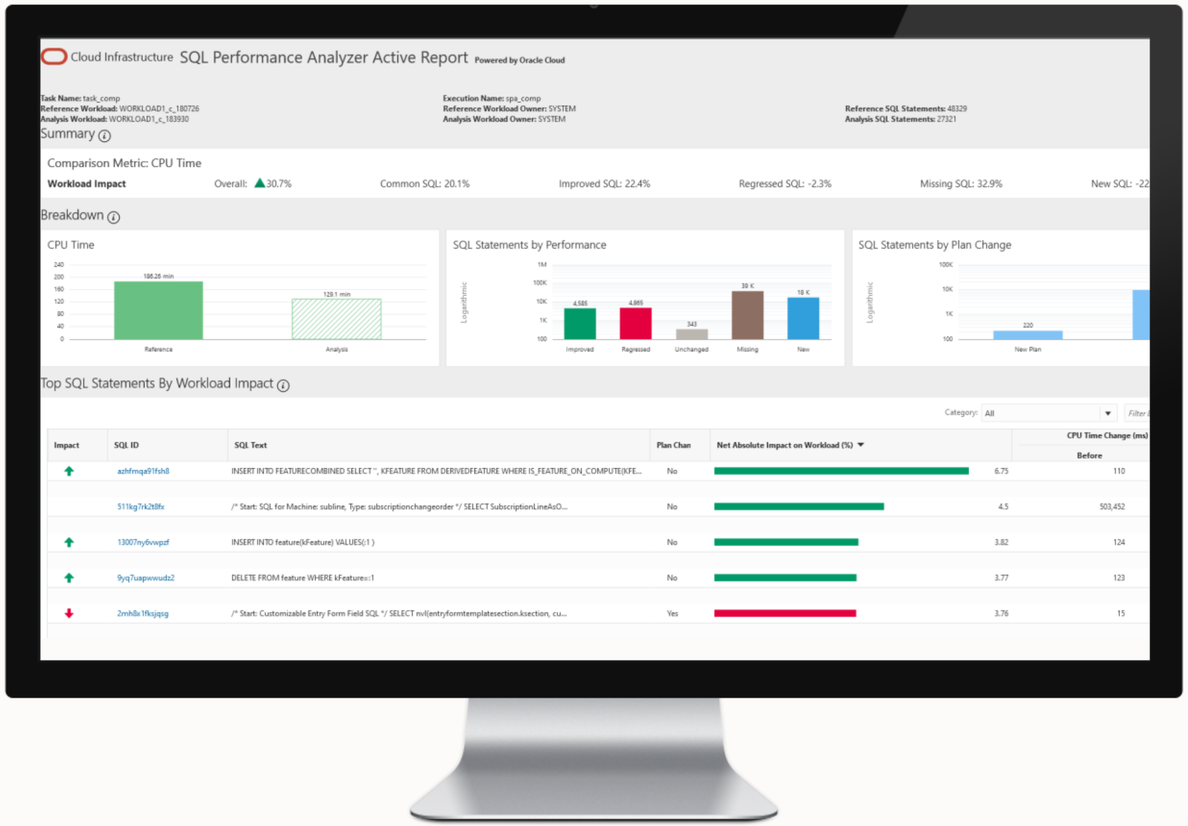
SQL Performance Analyzer (SPA)

Key capabilities

- Build different SQL trials (experiments) of SQL statements performance by local/remote test execution or explain plan
- Identifies potential regressions before actual change and helps in fixing regressions
- Modernized and optimized workflows with baked-in best practices
- Generates comprehensive workload report for the assessed change as well as granular SQL level view by performance metrics

Benefits

- Proactively address potential issues to maintain SLAs
- Increases business agility to adopt to new technologies
- Low overhead capture of SQL workload to STS on database system
- Remediate the regressed SQL



SQL Performance Analyzer

SPA: Overall steps

Capture SQL Tuning Set(s)

Pre-change trial

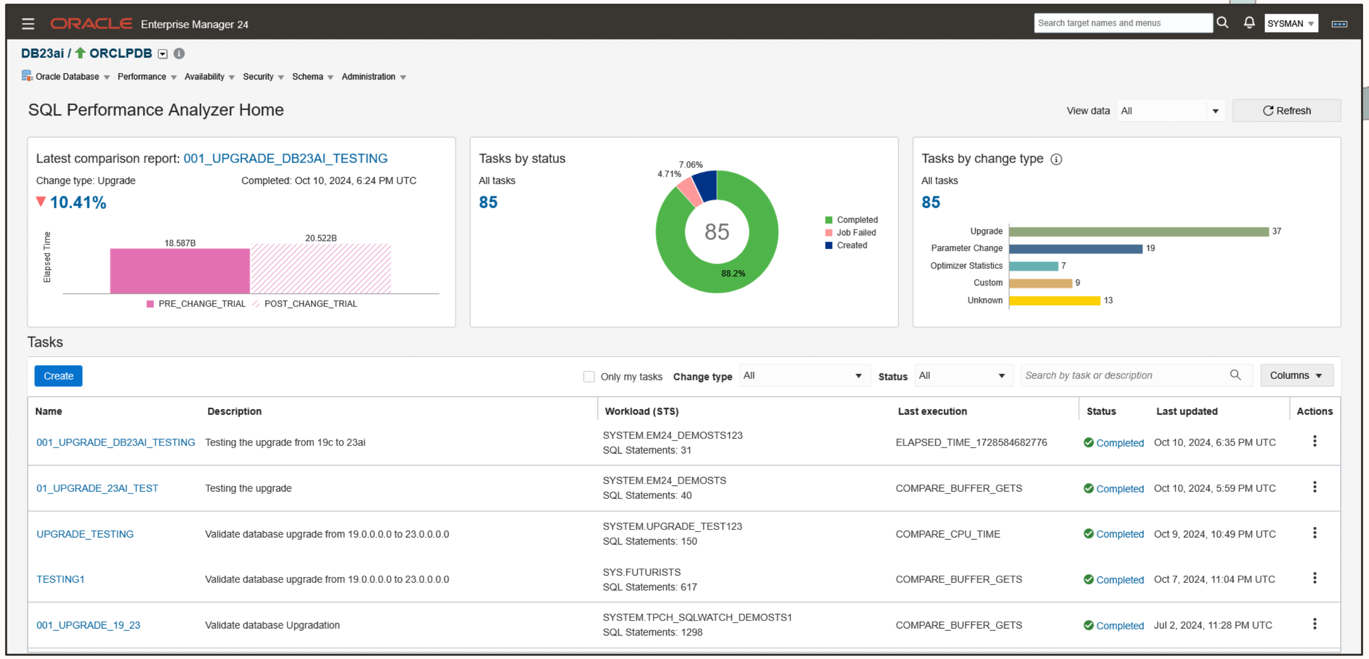
Change: Upgrade to DB 23ai

Post-change trial

Compare performance

Analyze report

Tuning changes



DEMO

Best practices

SQL Performance Analyzer

Use as few initialization parameters as possible

- Stick to default settings

Establish baseline on same release, e.g., DB19c before upgrade to 23ai

- Understand workload behavior, overhead of any artifacts introduced by SPA, DB Replay

Always make one change at a time, this will help understand causality

Save workloads, results and tuning artifacts

- STS, workload capture/replay, SQL Profiles, export baselines, results, reports, AWR, etc.
- Tests systems are transient

Best practices

SQL Performance Analyzer

1. Use hybrid mode – explain plan trials to reduce workload to a small subset
2. Any SQL execution plan changes – improved or regressed performance, test additionally since faster HW or other such factors may mask underlying issues
3. Use multiple comparison metrics, start with Elapsed Time
4. Check for #executions, #rows processed, hash value (if checking for correctness) in addition to execution plan for individual SQL statements
5. Use CONVERT_STS trial option as last resort, pay attention to #executions in each trial, it is better to compare multi-executions to multi-executions trial
6. Ensure con_id & dbid is mapped correctly to the destinations (documentation)

UNITED

United Airlines Case Study

EM Tech Forum, 2024

United Airlines: Challenges

500+
Prod/Non-Prod
environments

- Mission critical databases
- Operations & revenue impact
- Response time/ latencies play a big role in defining a successful operation
- 1000s of SQL statement with sub-sub second response system
- Use EM 13.5 RU 21, Diagnostics/Tuning, Real Application Testing (SPA) features
- Database 19c RU19, RU23

United Airlines: Challenges

Upgrade/Migration

- How to predict the workload behavior after an upgrade?
- Will the SQL performance regress or improve?
- How to convince application team without key metrics or data about performance improvement?
- How to test with the full dataset for multiple peak periods?

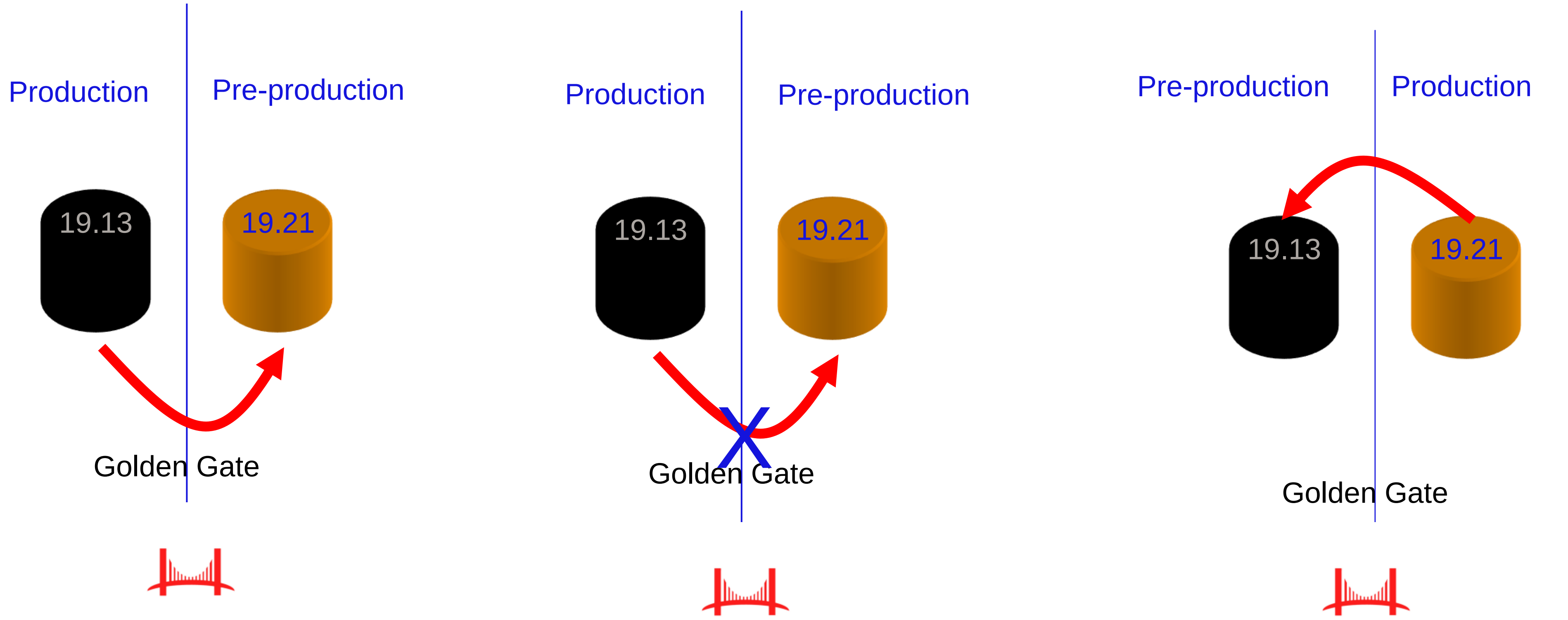
Workload profiles

- Numerous complex queries
- Different workload at different times of the day
  - Peak travel hour
  - Rush hour
  - Slow at noon
- Scheduled workloads
  - Maintenance jobs
  - Backups/purges

SPA at United

- ✓ With introduction of SPA, it has made our lives easier as it helps us to assess and predict the change due to the upgrades/migrations
- ✓ At United, it is our Go-to tool for performance predictability
- ✓ We use and stress on using SPA on mission critical databases during upgrades and platform migrations
- ✓ Now we can confidently present results to application team to get their buy in for upgrades/migrations

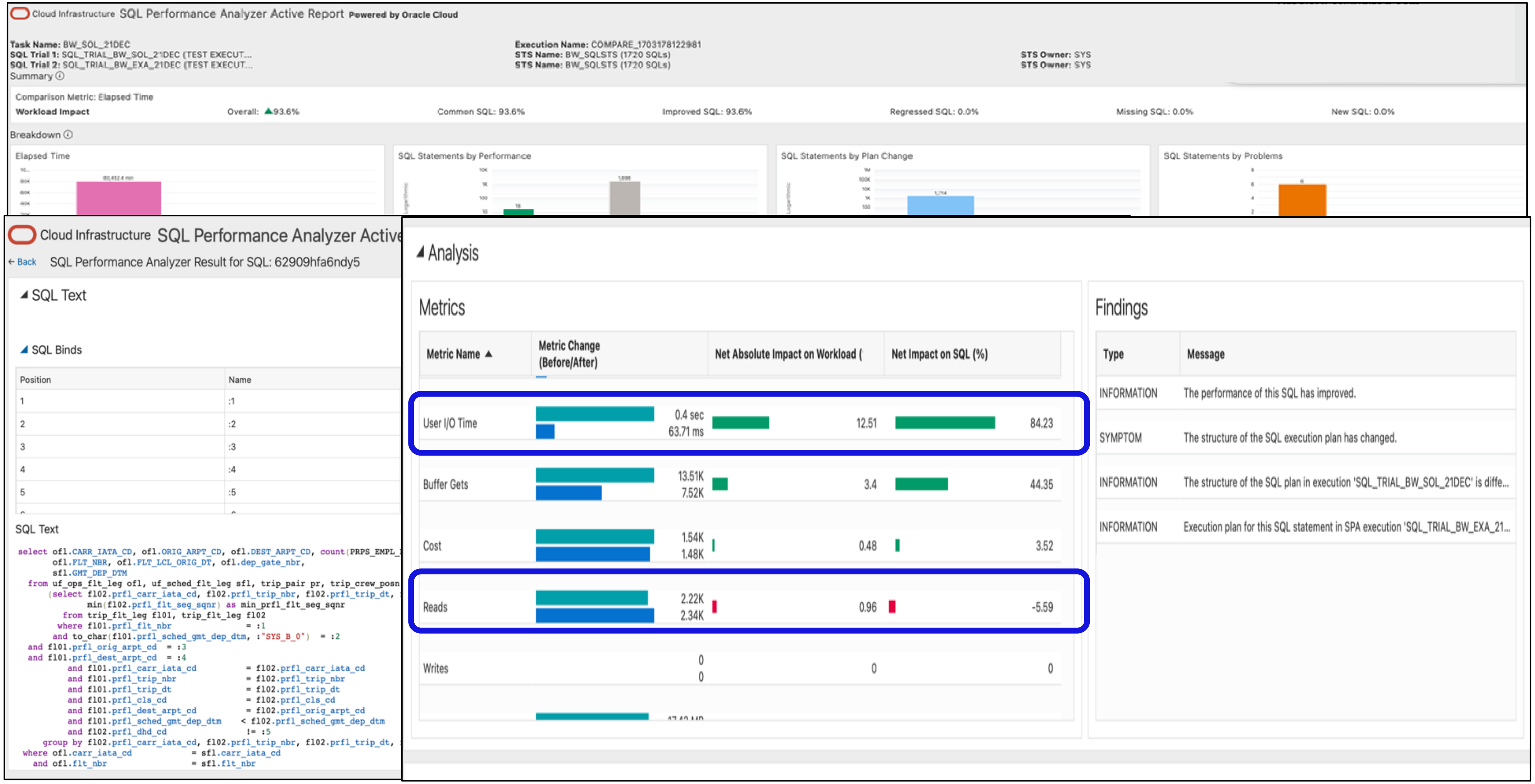
General testing strategy for migration, patching, and upgrades



1. Workload replication using Golden Gate, capture workload and move it to pre-production
2. Pre-production STS, compare performance to Production
3. Setup fallback after successful migration/upgrade

Strategy to be used for DB23ai and other changes to test with latest dataset and multiple production workloads

SPA Analysis



United Airlines – SPA Best Practices

- ✓ Use incremental cursor cache capture of STS to avoid overhead on the production systems
- ✓ Run the workload in production during maintenance window
- ✓ Don’t just focus on activities during peak time but also on nightly activity where various maintenance jobs run
- ✓ Successful timely completion of these could also be crucial to next day operation
- ✓ While running in test, make sure it has comparable configurations like prod in terms of SGA/PGA or optimizer setting and other parameters
- ✓ Use several runs and several metrics to compare like elapse time, CPU time (Pre & Post)

"With introduction of the Oracle SQL Performance Analyzer, it has made our lives easier as it helps us to assess and predict the change due to the upgrades/migrations"

Sunil Kalra

Principal Database Administrator

Q&A
Learn More

Web: oracle.com/enterprisemanager

Videos: youtube.com/OracleEnterpriseMgr

Blogs: blogs.oracle.com/observability

Docs: http://docs.oracle.com/en/enterprise-manager

Try it now

Oracle Cloud Free Tier

Always Free

Services you can use for unlimited time

+

30-Day Free Trial

Free credits you can use for more services

www.oracle.com/cloud/free

Appendices