



# Ensure Performance Stability When Upgrading Oracle Databases

Roy Swonger  
Vice President  
Database Upgrade and Utilities  
Oracle Corporation



**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, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

\$> whoami



22 years w/Oracle

Previously with DEC Rdb

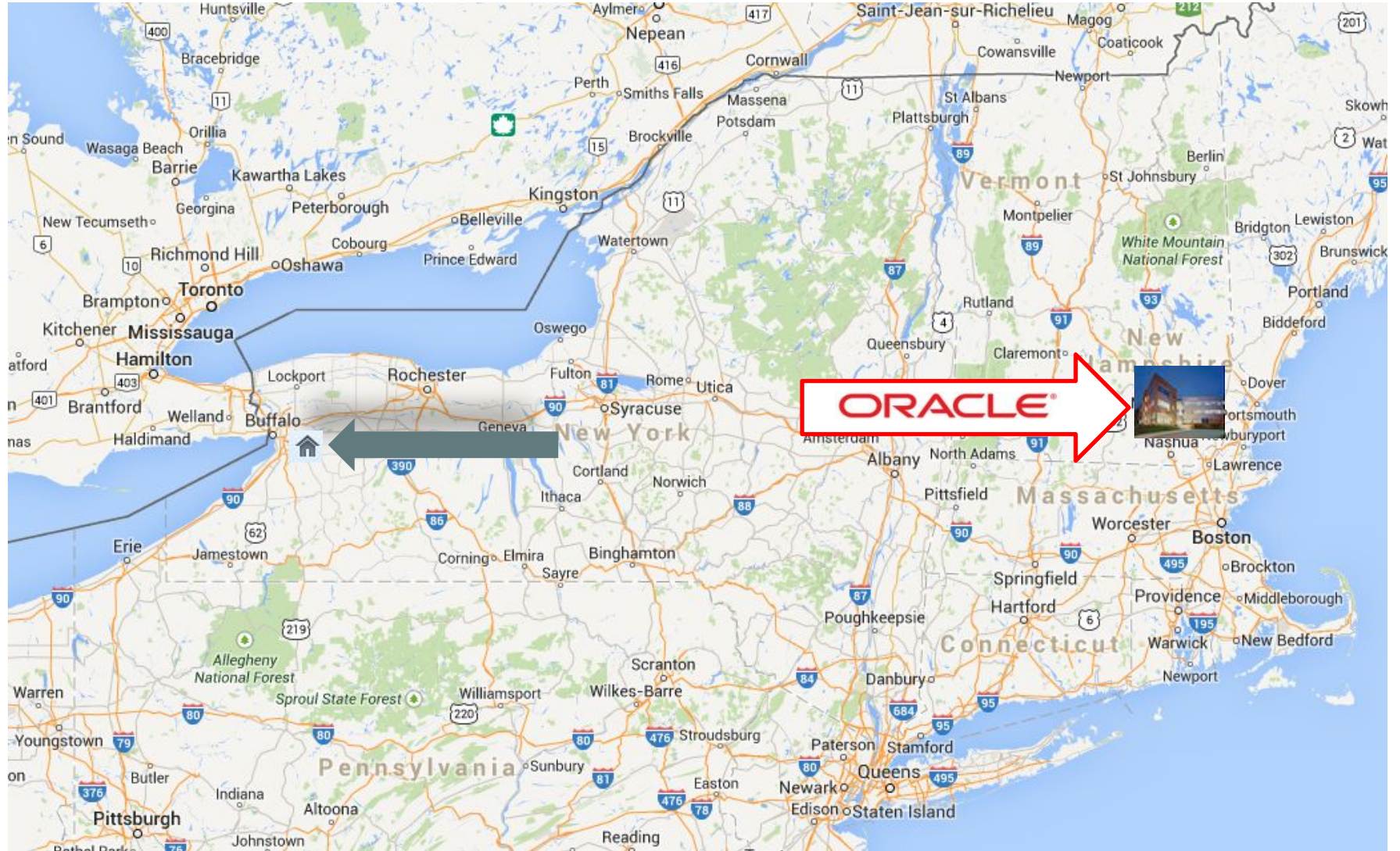
13+ years managing

Data Pump

Database Upgrade

SQL\*Loader

Transportable Tablespaces






# Database Upgrade Blog - Slides

▪ <https://MikeDietrichDE.com/>

## Upgrade your Database – NOW!

Mike Dietrich's Oracle Database Upgrade Blog



Blog Slides Hands-On Lab Workshops Papers / Docs Videos Scripts Links About

### Slides Download Center

Edit

This page will be refreshed to a more user-friendly look&feel soon.

#### Comprehensive

- [Upgrade, Migrate & Consolidate to Oracle Database 12.2 & Cloud](#)  
*Updated: 26-FEB-2017*
- [Upgrade, Migrate & Consolidate to Oracle Database 12c](#)  
*Refreshed 3-DEC-2016*
- [Why you need to upgrade NOW!](#)

#### Recent Posts

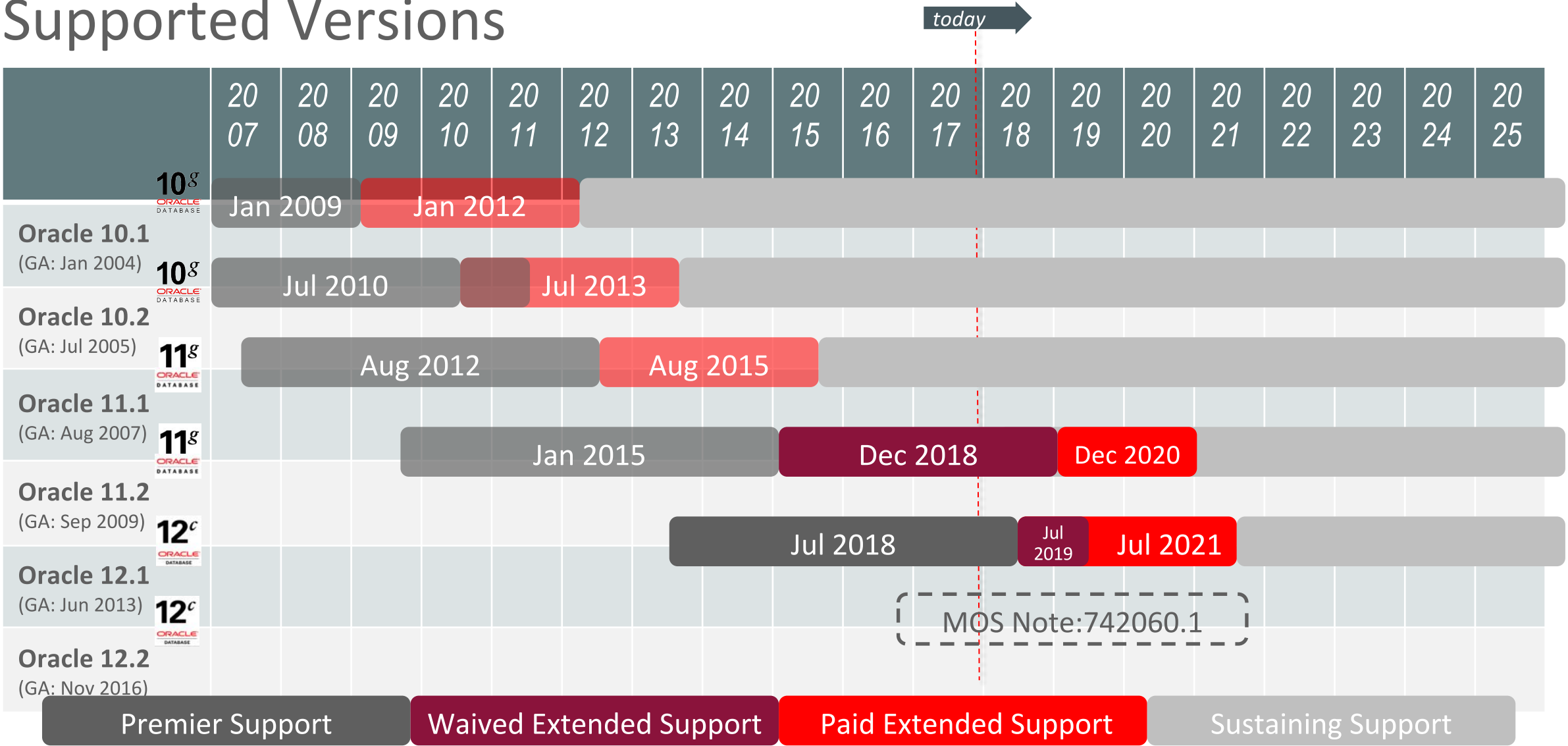
- Hands-On Lab available: Upgrade to Oracle Database 12.2.0.1
- Collaborate Conference 2017 – Upgrade "Performance" Talk + Oracle Database 12.2 Hands-On Lab
- Multiple hop upgrades? Execute the matching preupgrade scripts for each hop
- Oracle Database 12.2.0.1 for Windows available



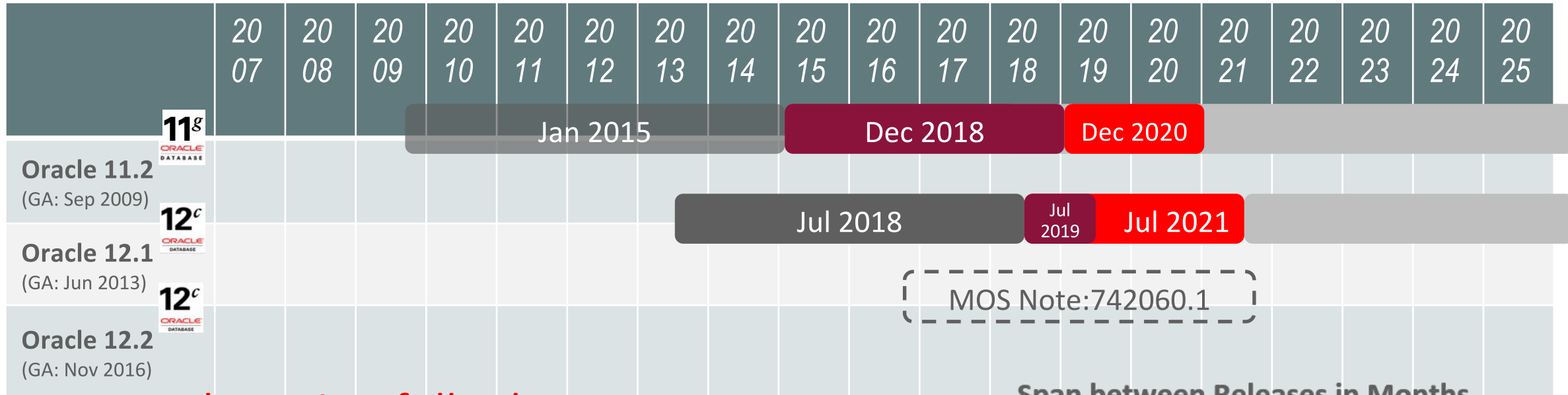


# Preface

# Supported Versions

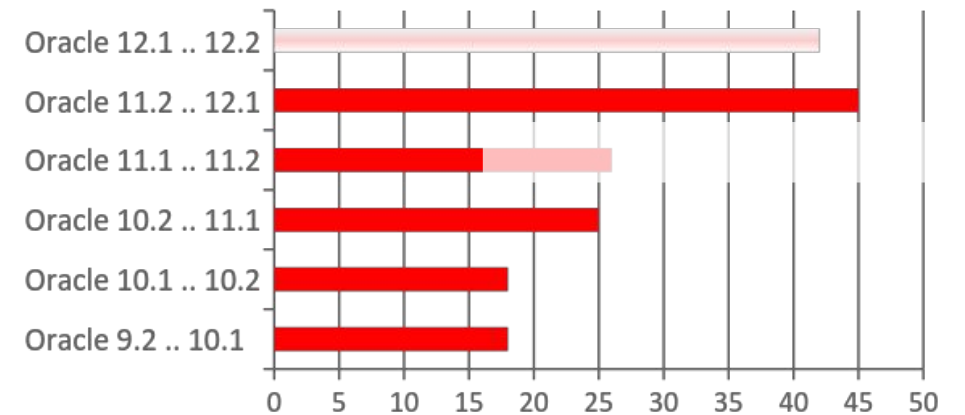


# Oracle Database 12.1.0.2 or 12.2.0.1?



- Every release is a full release
- Every release will get a significant number of new features and changes
- There's no such thing as **THE 2<sup>nd</sup> release** anymore!

Span between Releases in Months







# Your biggest fear when you upgrade/migrate?

Let me guess ...



# Unpredictable Surprises

# Paradigm





*"The entire upgrade project of our 300 Oracle databases at Die Mobiliar is running very well.*

*When we catch issues, we fix them before going live."*

Paolo Kreth, Group Manager Databases, Die Mobiliar

# Before Upgrade

# Preserve Performance Figures

- On production:

- Check your current AWR settings:

```
SQL> select * from DBA_HIST_WR_CONTROL;
```

- Adjust retention to fit at least **40 days** of snapshots taken **every 30 minutes**:

```
SQL> exec DBMS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS (57600, 30);
```

- Trend analysis:

```
?/rdbms/admin/awrinfo.sql
```

```
?/rdbms/admin/utlsyxsz.sql
```

- AWR export:

```
?/rdbms/admin/awrextr.sql
```

- If Diagnostic Pack is not licensed (e.g. Standard Edition): [MOS Note: 94224.1-Statspack Complete Reference](#)



# Helpful Scripts

- Directly from Oracle's Optimizer PM
  - [https://github.com/oracle/dw-vldb/tree/master/upgrading\\_to\\_12c](https://github.com/oracle/dw-vldb/tree/master/upgrading_to_12c)

Branch: master ▾ dw-vldb / upgrading\_to\_12c /

Create new file Find file History

nigelbayliss 12.1.2 Latest commit 1337e4d

..	
dpump_copy	upgr12cR1_1
duplicate	12.1.2
show_stats	upgr12cR1_1
README.md	upgr12cR1_1

**show\_stats**

These scripts demonstrate how to view extended statistics, histograms, SQL plan directives and column usage information. They also demonstrate how you can see which histograms and extended statistics have been created automatically.

**duplicate**

These scripts query a database schema to spool scripts that can be used to create matching histograms and extended statistics on another database.

**dpump\_copy**

These scripts demonstrate how easy it is to use Data Pump to copy all relevant statistics from one database schema to another.

## The Optimizer and Upgrading to Oracle Database 12c

If you are upgrading to Oracle Database 12c, you need to be aware that the Oracle Optimizer stores certain metadata to support its adaptive features. This can affect the way a DBA manages statistics during testing and upgrade. Certain optimizer statistics are created in response to the optimizer metadata to improve cardinality estimates over time. Specifically, these are *histograms* (in response to column usage information) and *column group statistics* (in response to SQL plan directives).

You need to be aware of changes to optimizer statistics and metadata so that you can manage the optimizer successfully during an upgrade.

# Now we have to patch a bit ...

I know, it's hard work and no fun at all – but it's important, necessary and **essential**

# Download Patch Sets, PSUs, BPs, CPUs – Quick Reference

- [MOS Note:1454618.1 - Quick Reference to Patch Numbers for Database PSU, SPU\(CPU\), Bundle Patches and Patch Sets](#)

PSU, SPU(CPU), Bundle Patches

12.1.0.2				
Description	PSU	GI PSU	Proactive Bundle Patch	Bundle Patch (Windows 32bit & 64bit)
JAN2017	<a href="#">24732082</a> (12.1.0.2.170117)	<a href="#">24732082</a> (12.1.0.2.170117)	<a href="#">24968615</a> (12.1.0.2.170117)	<a href="#">25115951</a> (12.1.0.2.170117)
OCT2016	<a href="#">24006101</a> (12.1.0.2.161018)	<a href="#">24412235</a> (12.1.0.2.161018)	<a href="#">24448103</a> (12.1.0.2.161018)	<a href="#">24591642</a> (12.1.0.2.161018)
JUL2016	<a href="#">23054246</a> (12.1.0.2.160719)	<a href="#">23273629</a> (12.1.0.2.160719)	<a href="#">23273686</a> (12.1.0.2.160719)	<a href="#">23530387</a> (12.1.0.2.160719)
APR2016	<a href="#">22291349</a> (12.1.0.2.160419)	<a href="#">22646084</a> (12.1.0.2.160419)	<a href="#">22899531</a> (12.1.0.2.160419)	<a href="#">22809813</a> (12.1.0.2.160419)
JAN2016	<a href="#">22100559</a> (12.1.0.2.160119)	<a href="#">22191349</a> (12.1.0.2.160119)	<a href="#">22100559</a> (12.1.0.2.160119)	<a href="#">22310559</a> (12.1.0.2.160119)



# Download Patch Sets, PSUs, BPs, CPUs – Assistant

- Assistant: Download Reference for Oracle Database/GI PSU, SPU(CPU), Bundle Patches, Patchsets and Base Releases ([Doc ID 2118136.2](#))

★ Assistant: Download Reference for Oracle Database/GI PSU, SPU(CPU), Bundle Patches, Patchsets and Base Releases (Doc ID 2118136.2) To Bottom

**Selection(s)**

**What would you like to download?**

- Oracle Database Base Releases
- Oracle Database Patchsets
- Oracle Database Release Updates (RU) and Release Update Revisions (RUR)
- Oracle Database PSU, SPU(CPU), Bundle Patches
- OJVM PSU/Bundle Patches
- Latest Available Microsoft Windows Patches

**Additional Clarification**

Please select an option above to begin clarifying the issue you are experiencing.

**Solution(s)**

Possible Solutions will appear once you make your selection.

# Upgrade Information / Alerts

- START HERE → CLICK on the RELEASE LINK [MOS Note:161818.1](#)

Oracle Database Releases Status Summary

Release <i>(Click for Details)</i>	Current Patch Set <i>(Click for Availability and Known Issues)</i>	Next Patch Set	Premier Support Ends	Extended Support Ends	Notes
<a href="#">12.1.0.X</a>	<a href="#">12.1.0.2</a>	None	-	-	Base release 12.1.0.2 is the available for 2 - see <a href="#">Note:</a>
<a href="#">11.2.0.X</a>	<a href="#">11.2.0.4</a>	None	Jan-2015	<b>Dec-2020</b> Extended Support fees are waived from Jan-2015 to 31-May-2017 See <a href="#">Note:1067455.1</a>  Patching for 11.2.0.1 ended on 13/Sep/2011 Patching for 11.2.0.2 ended on 31/Oct/2013 Patching for 11.2.0.3 ended on 27/Aug/2015 - See <a href="#">Note:742060.1</a>	Base release 11.2.0.4 is the 11.2 Patch Set <a href="#">Note:118978</a>
<a href="#">11.1.0.X</a>	<a href="#">11.1.0.7</a>	None	Aug-2012	<b>Aug-2015</b> Patching for 11.1.0.7 ended on 31/Aug/2015 for most platforms. <b>Limited Extended Support</b> available for HPUX-Itanium - see <a href="#">Note:1307745.1</a>	Base release 11.1.0.7 is the
				<b>Jul-2013</b> Limited Extended Support ended on 31-July-2015 for most	10.2.0.5 is the

- Then CLICK on "Availability and Known Issues" for your patch set

## Base Release

This section gives links to information for each Oracle 12c Release 1 release.

Release	Comments
12.1.0.2	Oracle 12c Release 1 (12.1.0.2)  <b>Availability and Known issues for 12.1.0.2</b> <b>List of fixes included in 12.1.0.2</b>
	<a href="#">Note:1683799.1</a> <a href="#">Note:1683802.1</a>
	Oracle 12c Release 1 Base Release

# Upgrade Information / Alerts

- Alerts and Known Issues with 12.1.0.2? [MOS Note:1683799.1](https://mos.cis.oracle.com/1683799.1)

## General Alerts / Issues

Bug/Doc	Fixed in PSU/Bundle	Description	Updated
<a href="#">Note:2058461.1*</a>		Corruption during Recovery after upgrading to 12c for Compressed Tables	20/Jul/2016
<a href="#">Note 1608167.1*</a>		ORA-600 [kdsgrp1] ORA-1555 / ORA-600 [ktbdchk1: bad dscn] due to Invalid Commit SCN in INDEX block	10/Aug/2016
<a href="#">21608238+</a>		Wrong results with rowsets (enabled by default)	29/Jun/2016
<a href="#">20144308+</a>		ORA-27086 or ORA-1182 RMAN May Overwrite a SOURCE Database File during TTS, TSPITR, etc when OMF is used in SOURCE. ORA-1578 ORA-1122 in SOURCE afterwards	10/May/2016
<a href="#">20369110</a>		ORA-600[9999] / Cannot enable more than 8 kernel options (such as uniaud , olap, lbac etc..)	14/Jul/2015
<a href="#">20881450+</a>	12.1.0.2.DBBP:160119	Wrong results or Assorted dumps and errors querying HCC tables with OLTP blocks	08/Jun/2016
<a href="#">Note 1944645.1*</a>	12.1.0.2.3	ORA-600 [kdblckerror]..[6266] corruption with self-referenced chained row. ORA-600 [kdsgrp1] / Wrong Results / ORA-8102	01/Mar/2016
<a href="#">Note:1957710.1*P</a>	12.1.0.2.160419	12c Hang: LGWR waiting for 'lgwr any worker group' or ORA-600 [kcrfrgv_nextlwn_scn] ORA-600 [krr_process_read_error_2] on IBM AIX / HPIA	02/May/2016



# Upgrade Information / Alerts

- Alerts and **new issues** with 12.1.0.2? [MOS Note:1683799.1](#)

## **Issues introduced in 12.1.0.2**

This section lists bugs **introduced** in 12.1.0.2 (if any). Such issues may be either serious or trivial but the aim is to list them all to help customers assess the risk of applying the Patch Set on top of 12.1.0.1

<b>Bug/Doc</b>	<b>Fixed in PSU/Bundle</b>	<b>Description</b>	<b>Updated</b>
<a href="#">Note:2058461.1*</a>		Corruption during Recovery after upgrading to 12c for Compressed Tables - Superseded	06/Apr/2016
<a href="#">24385983</a>		False ORA-4020 if fix for bug 12608451 is present	03/Oct/2016
<a href="#">23238932</a>		Query fails with ORA-979 when select list subquery has CONNECT BY clause and fix 16053273 present	13/Dec/2016
<a href="#">23073214</a>		ORA-600[kghstack_free1][kzaxpsqbn: ksmals: 1] or spin with AUDIT_TRAIL=XML,EXTENDED	05/Dec/2016
<a href="#">22062517</a>		wrong result with fix for 17833448 enabled	31/Aug/2016
<a href="#">21971099</a>		12c wrong cardinality from SQL analytic windows functions	28/Mar/2016
<a href="#">21826068</a>		Wrong Results when _optimizer_aggr_groupby_elim=true	11/Jan/2017
<a href="#">21498770D</a>		automatic incremental statistics job taking more time with fix 16851194	03/Jan/2017
<a href="#">21482099</a>		ORA-7445 [opitca] or ORA-932 errors from aggregate GROUP BY elimination - superseded	11/Jan/2017
<a href="#">21128593</a>		Updating the master table at the end of Data Pump job is Slow	05/Jan/2017

# Important **Optimizer** Issues and Fixes

- Things to consider before upgrade to Oracle Database 12.1.0.2 to **avoid Poor Performance** or **Wrong Results**: [MOS Note:2034610.1](https://www.oracle.com/technetwork/middleware/optimizer/2034610-1.html)

No PSU	PSU 1	2	3	4	5	160119	160419	160719	161018	170117	170418	170718	Bugs Fixed
													<a href="#">Document 19855835.8</a> Upgrade slow when reorganizing large stats history tables <b>NB: Only applicable for upgrades from 11.2.0.3 or below. Apply before running the 12.1.0.2 upgrade script. There is no benefit to applying it later on.</b>
													<a href="#">Document 20879889.8</a> Open cursor leak from DML on table with a materialized view log
													<a href="#">Document 20476175.8</a> High VERSION_COUNT (in V\$SQLAREA) for query with OPT_PARAM('_fix_control') hint
													<a href="#">Document 18650065.8</a> Wrong Results on Query with Subquery Using OR EXISTS or Null Accepting Semijoin
													<a href="#">Document 20636003.8</a> Slow Parsing caused by Dynamic Sampling (DS_SVC) queries (side effects possible ORA-12751/ ORA-29771)
													<a href="#">Document 20807398.8</a> ORA-600 [kgl-hash-collision] with fix to bug 20465582 installed
													<a href="#">Document 21826068.8</a> Wrong Results when _optimizer_aggr_groupby_elim=true
													<a href="#">Document 23147905.8</a> Wrong Results with filtering on an aggregation expression
													<a href="#">Document 23665623.8</a> ORA-7445 kkeics with fix for bug 21091518 (supersedes <a href="#">Document 21091518.8</a> Extend fix of bug 18304693 to Partition Views)
													<a href="#">Document 24739928.8</a> ORA-7445 [kglMutexCleanupAll] with fix for Bug 13542050 (supersedes <a href="#">Document 13542050.8</a> A mutex related hang with holder around 65534 (0xffff))
													<a href="#">Document 18430870.8</a> Adaptive Plan and Left Join Give Wrong Result
													<a href="#">Document 19174639.8</a> Plan regression in 11.2.0.4 - OJPPD not occurring when expected
													<a href="#">Document 21171382.8</a> Enh: AUTO_STAT_EXTENSIONS preference on DBMS_STATS
													<a href="#">Document 22864303.8</a> SELECT query failing with ORA-979 after upgrade from 12.1.0.1 from 12.1.0.2
													<a href="#">Document 23321926.8</a> Wrong results with Partial Join Evaluation or Group By Key Reduction column replacement
													<a href="#">Document 20804826.8</a> Wrong results or ORA-600 [qescopyorigrowsetopns:cnt] with shared binds across views





# Important SQL Plan Management Issues and Fixes

- Patches to Consider for 12.1.0.2 to Avoid Problems with SQL Plan Management (SPM): [MOS Note:2035898.1](#)

No PSU	PSU 1	2	3	4	5	160119	160419	160719	161018	170117	170418	170718	Bugs Fixed	
<a href="#">Patch 18747342</a> for 12.1.0.2.0		<a href="#">Patch 18747342</a> for 12.1.0.2.160119												<a href="#">Document 18747342.8</a> Plan reproduction fails for SQL statement with a [NOT] EXISTS select list subquery
<a href="#">Patch 18961555</a> for 12.1.0.2.0														<a href="#">Document 18961555.8</a> Static PL/SQL baseline reproduction broken by fix for bug 18020394
<a href="#">Patch 22511835</a> for 12.1.0.2.0		<i>Included in PSU 160119 and above</i>												<a href="#">Document 19141838.8</a> ORA-600 [qksanGetTextStr:1] from SQL Plan Management after Upgrade to 12.1
		<a href="#">Patch 22324460</a> for 12.1.0.2.160119												<a href="#">Document 22324460.8</a> ORA-600 [qksanGetTextStr:1] After Unpacking SQL Plan Baselines
<a href="#">Patch 20476175</a> for 12.1.0.2.0		<i>Included in PSU 5 and above</i>												<a href="#">Document 20476175.8</a> High VERSION_COUNT (in V\$SQLAREA) for query with OPT_PARAM('_fix_control') hint
<a href="#">Patch 21075138</a> for 12.1.0.2.0	<a href="#">Patch 21075138</a> for 12.1.0.2.3		<a href="#">Patch 21075138</a> for 12.1.0.2.160719		<a href="#">Patch 21075138</a> for 12.1.0.2.161018								<a href="#">Document 21075138.8</a> SPM does not reproduce plan with SORT UNIQUE	
<a href="#">Patch 21463894</a> for 12.1.0.2.0														<a href="#">Document 21463894.8</a> Failure to reproduce plan with fix for bug 20978266 (supersedes <a href="#">Document 20978266.8</a> SQL not using plan in plan baselines and plans showing as not reproducible)
<a href="#">Patch 20877664</a> for 12.1.0.2.0		<i>Included in PSU 160119 and above</i>												<a href="#">Document 20877664.8</a> SQL Plan Management Slow with High Shared Pool Allocations





# Complexity and non-Standard

There are no Golden Magic Rules for every configuration



# Parameter Recommendations

- General guidelines:
  - The fewer parameters you have in your `spfile`, the better
  - **Remove outdated parameters, underscores, events ...**
    - Unless directed by applications (e.g. E-Business Suite, Siebel, SAP etc.) to set them
    - [MOS Note:396009.1 DB Init Parameters for EBS R12](#)

The image shows a collage of Oracle documentation pages. On the left is the 'SAP Database Guide: Oracle' with a sidebar menu. The main content area is titled 'Database Parameters' and lists OS-specific paths for UNIX and Windows. Below this, it discusses the 'init.ora' file and provides 'Guidelines for Configuring Settings in the init.ora File'. On the right, there is a 'MOS Note' titled 'Interoperability Notes EBS 12.0 and 12.1 with Database 11gR2 (Doc ID 1058763.1)' dated December 2013. The MOS note content includes a title, a date, and a list of parameters with their default values and recommendations.

**SAP Database Guide: Oracle**

- New Features
- Getting Started with Oracle and the SAP System
- Database Security
- Computing Center Management System
- Specification of SAP Tables in the ABAP Dictionary
- Oracle Database Storage Parameters in the ABAP Dictionary
- Support on SAP Service Marketplace
- Database System Configuration

## Database Parameters

The SAP system for the Oracle database comes with a standard profile that you can access the original parameters, if necessary.

- UNIX  
OS> <ORACLE\_HOME>/dbs/init<DBSID>.ora
- Windows  
OS> <ORACLE\_HOME>\database\init<DBSID>.ora

For example (UNIX):  
/oracle/C11/dbs/initC11.ora

This profile contains the default parameter settings recommended for you to use. You can also access the original parameters, if necessary.

**Interoperability Notes EBS 12.0 and 12.1 with Database 11gR2 (Doc ID 1058763.1)**  
Modified: 04-Dec-2013 Type: WHITE PAPER

### Interoperability Notes Oracle E-Business Suite Release 12.0 or 12.1 with Oracle Database 11g Release 2 (11.2.0)

December 2013

#### Guidelines for Configuring Settings in the init.ora File

The init.ora file contains parameters that have a major effect on the performance of Siebel Business Applications using Oracle Database. Use the following settings as guidelines for your initial configuration. Your final settings will vary depending on the hardware configuration, the number of users, and the size of the database. In the init.ora file, default parameter values are provided for small, medium, and large database deployments. Unless the configuration parameters are explicitly set, the default values will be used. For more information on the effects of these parameters on database performance and system resource utilization, see Oracle Database documentation.

This topic is part of [Configuring an Oracle Database for Siebel Business Applications](#).

Brief descriptions follow for several parameters for which you might have to adjust values:

- **CURSOR\_SHARING**. This parameter is set to EXACT by default and must not be changed.  
**CAUTION:** Changing this value might lead to failure of some Siebel Server components.
- **DB\_FILE\_MULTIBLOCK\_READ\_COUNT**. The database buffer cache parameter dictates the number of data blocks that are read from storage (such as a NetApp Filer), then set the value to 8 to reduce potential network traffic problems.
- **MEMORY\_TARGET**. This parameter specifies the Oracle system-wide usable memory. A common recommendation is to set it to a value of 15 GB.
- **NLS\_DATE\_FORMAT**. Set this parameter as needed. The default setting is DD-MON-YY. For information about the supported format, see Oracle Database documentation.

# Optimizer Adaptive Features

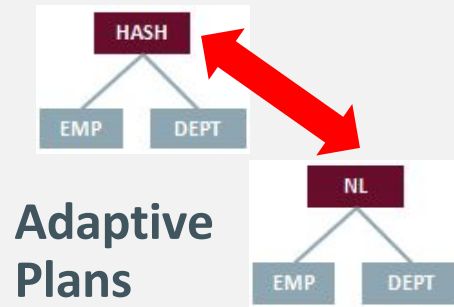
Oracle  
12.1

optimizer\_adaptive\_features

Oracle  
12.2

optimizer\_adaptive\_plans

Default: **TRUE**



optimizer\_adaptive\_statistics

Default: **FALSE**



**Conservative**

**Default**

**Progressive**

optimizer\_adaptive\_plans

FALSE

TRUE

TRUE

optimizer\_adaptive\_statistics

FALSE

FALSE

TRUE

# Optimizer **Adaptive** Features in **Oracle Database 12.1**

- [MOS Note: 2187449.1](#)  
[Recommendations for Adaptive Features in Oracle Database 12.1](#)
- Recommended approach:
  - **Patch 22652097**: introduces `OPTIMIZER_ADAPTIVE_PLANS` and `OPTIMIZER_ADAPTIVE_STATISTICS`
  - **Patch 21171382**: disables automatic creation of extended statistics
    - No need to tweak `OPTIMIZER_DYNAMIC_SAMPLING` anymore
    - Override with preference `AUTO_STATS_EXTENSIONS` set to ON
  - `SQL> alter system reset OPTIMIZER_ADAPTIVE_FEATURES;`
- For more information please see:
  - [https://blogs.oracle.com/UPGRADE/entry/optimizer\\_adaptive\\_features\\_obsolete\\_in](https://blogs.oracle.com/UPGRADE/entry/optimizer_adaptive_features_obsolete_in)
  - [https://blogs.oracle.com/optimizer/entry/optimizer\\_adaptive\\_features\\_in\\_the](https://blogs.oracle.com/optimizer/entry/optimizer_adaptive_features_in_the)

# New for 12.1.0.2 DBBP and 12.2 RU: Fix Control Persistence

- Automatic Fix Control Persistence (FCP) for Database Proactive Bundle Patch ([Doc ID 2147007.1](#))

## SCOPE

This document is intended for DBA's that are responsible for RDBMS software patching.

## DETAILS

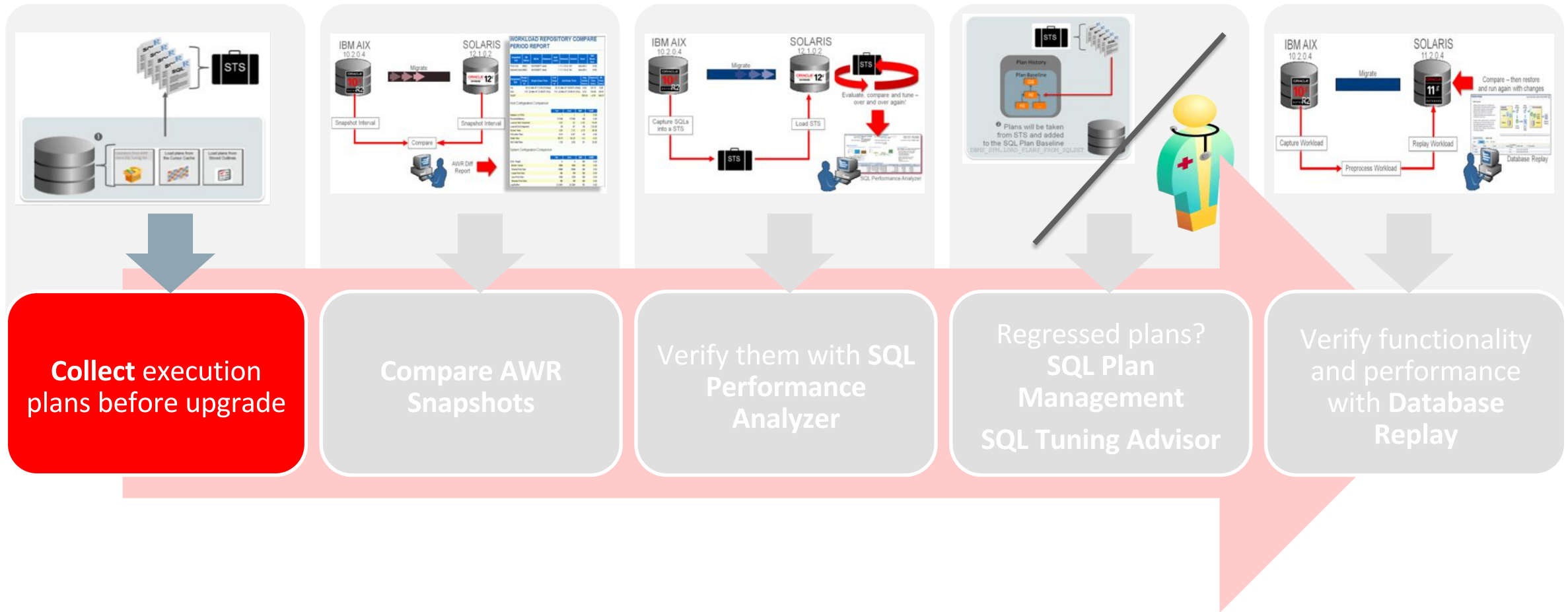
### Managing "installed but disabled" module bug fixes

Beginning with the April 2017 DB Proactive Bundle Patch (BP) for RDBMS 12.1.0.2 and beyond (including 12.2.0.1) the new "DI" bug fixes, which cause an execution plan change. At the successful conclusion of the patching event, none of the new, installed The status of any module bug fixes (which cause an execution plan change) that were in an "enabled" state prior to starting th

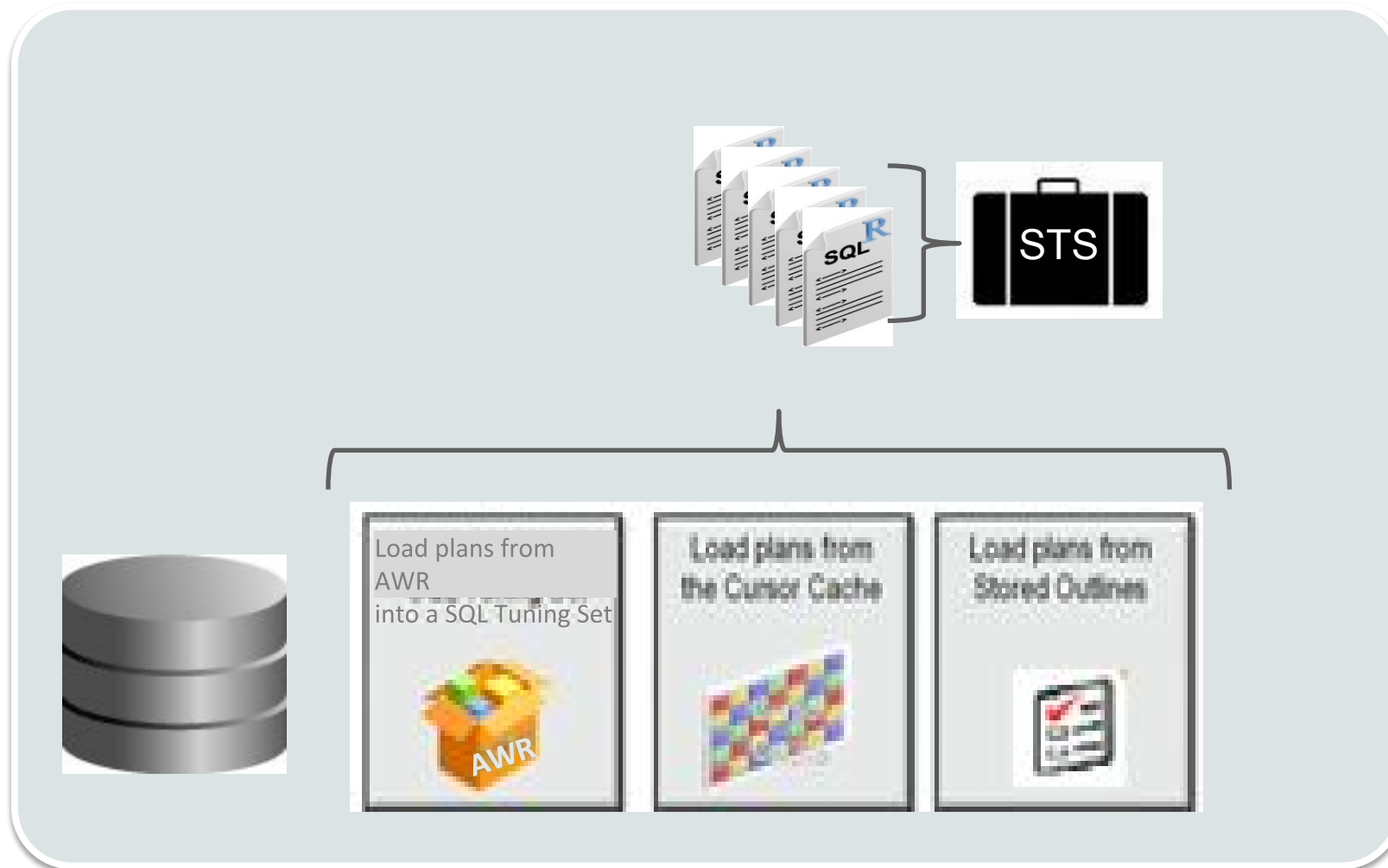


# Testing

# Testing Tools – Hand-in-Hand



# Collect Plans on Production



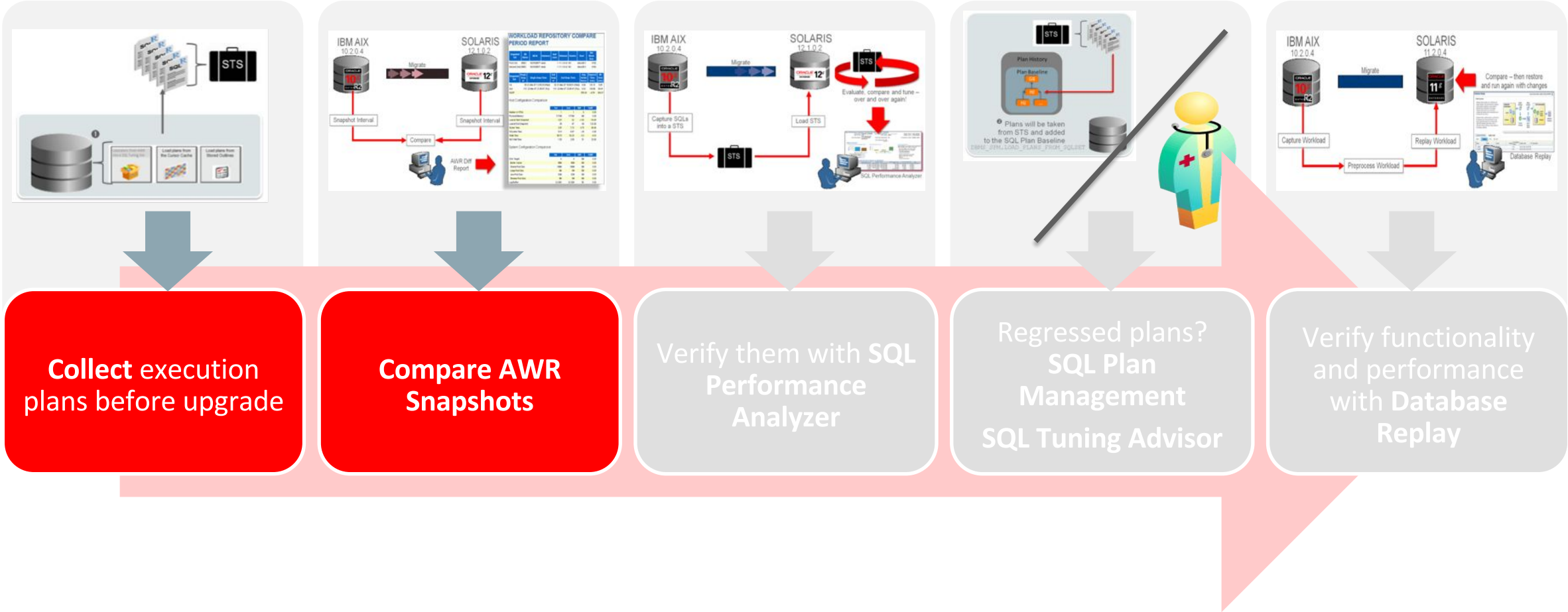


# Database Licensing Information User Manual

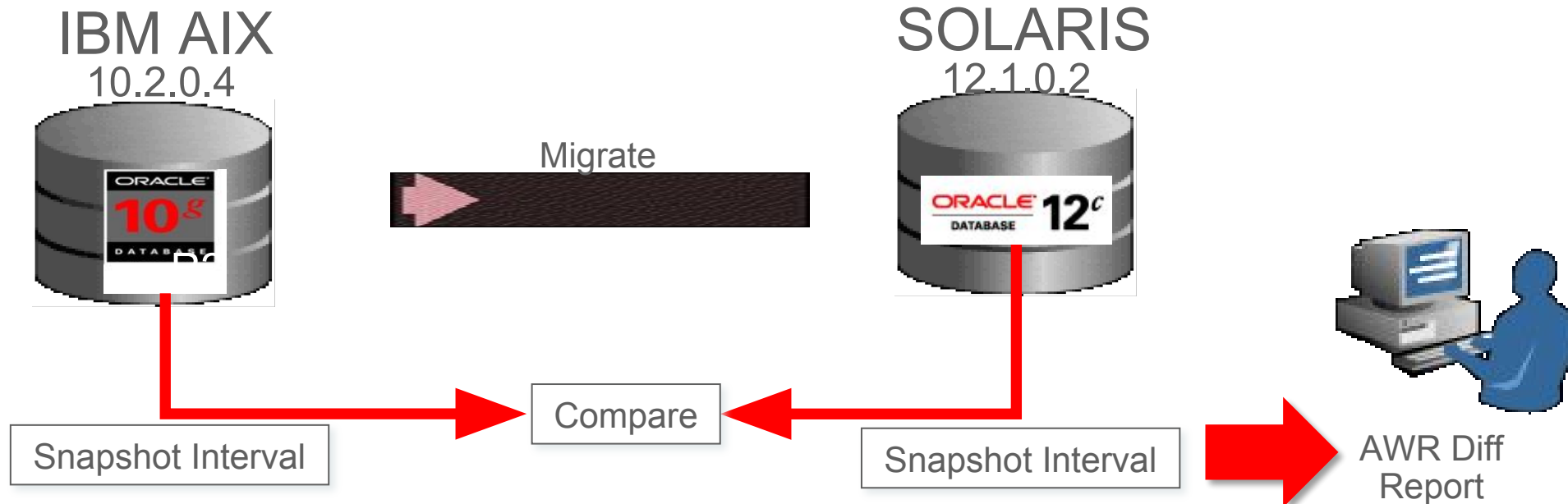
**Table 1-1 Feature Availability for Oracle Database Editions**

Feature/Option	SE1	SE/SE2	EE	Notes
<b>Consolidation</b>	—	—	—	—
SQL Tuning Sets	N	N	Y	SQL Tuning Sets can also be accessed by way of database server APIs and command-line interfaces. The following subprograms, part of the <code>DBMS_SQLTUNE</code> package, provide an interface to manage SQL Tuning Sets and are part of Oracle Database Enterprise Edition: <code>ADD_SQLSET_REFERENCE</code> , <code>CAPTURE_CURSOR_CACHE_SQLSET</code> , <code>CREATE_SQLSET</code> , <code>CREATE_STGTAB_SQLSET</code> , <code>DELETE_SQLSET</code> , <code>DROP_SQLSET</code> , <code>LOAD_SQLSET</code> , <code>PACK_STGTAB_SQLSET</code> , <code>REMOVE_SQLSET_REFERENCE</code> , <code>SELECT_CURSOR_CACHE</code> , <code>SELECT_SQLSET</code> , <code>SELECT_WORKLOAD_REPOSITORY</code> , <code>UNPACK_STGTAB_SQLSET</code> , <code>UPDATE_SQLSET</code> .

# Testing Tools – Hand-in-Hand



# Use the Right Tools: AWR Diff Reports



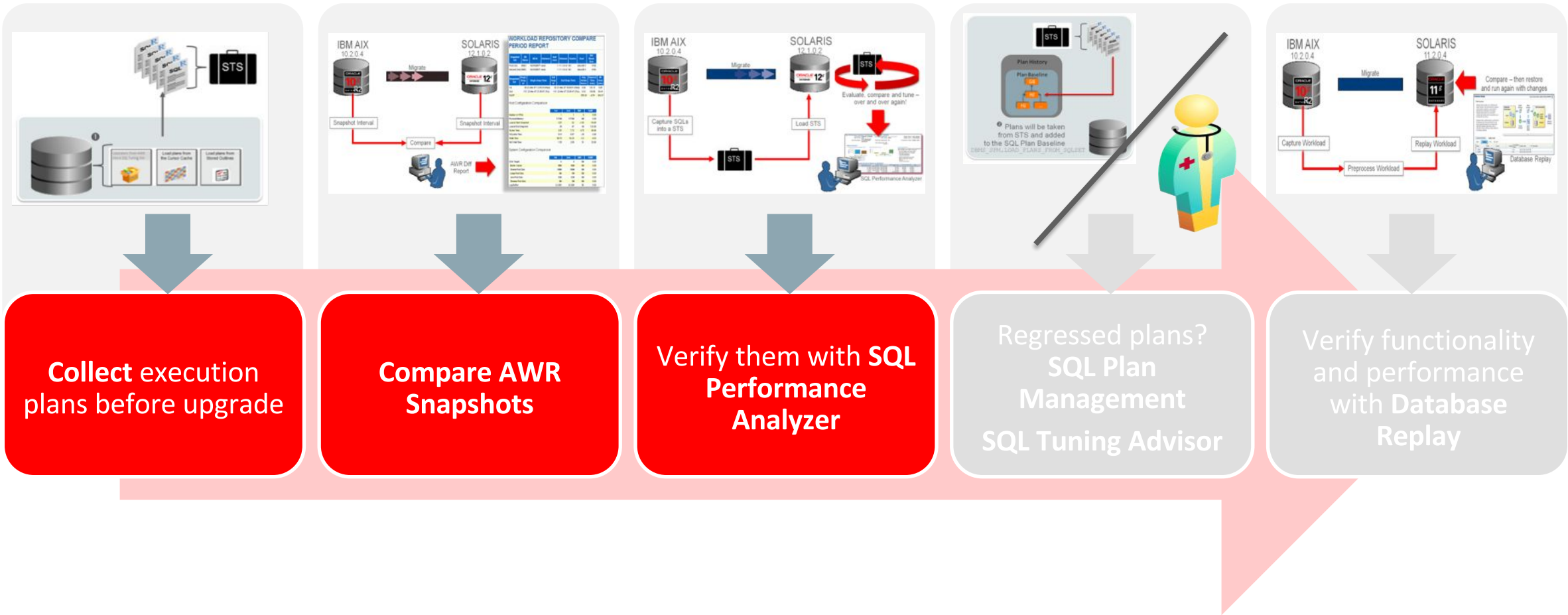
## Top Timed Events

- Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes

1st						2nd					
Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time	Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time
CPU time			68,289.05		43.73	db file sequential read	User I/O	22,193,998	114,919.21	5.18	23.17
db file sequential read	User I/O	6,686,953	37,737.81	5.64	24.17	enq: SS - contention	Configuration	3,913	98,997.90	25,299.74	19.96
gc buffer busy	Cluster	12,508,244	23,886.55	1.91	15.30	CPU time			73,786.55		14.88
TCP Socket (KGAS)	Network	680,629	12,514.65	18.39	8.01	row cache lock	Concurrency	73,940	48,472.30	655.56	9.77
db file scattered read	User I/O	1,572,296	4,271.68	2.72	2.74	reliable message	Other	41,148	47,600.87	1,156.82	9.60



# Testing Tools – Hand-in-Hand



# Use the Right Tools: SQL Performance Analyzer

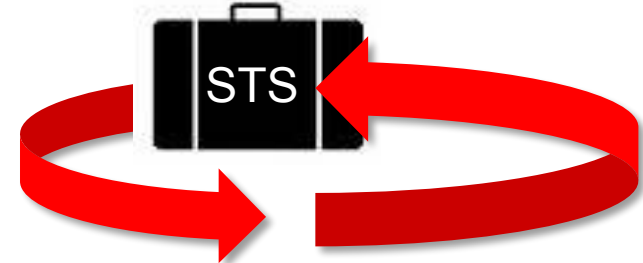
IBM AIX  
10.2.0.4



Migrate



SOLARIS  
12.1.0.2

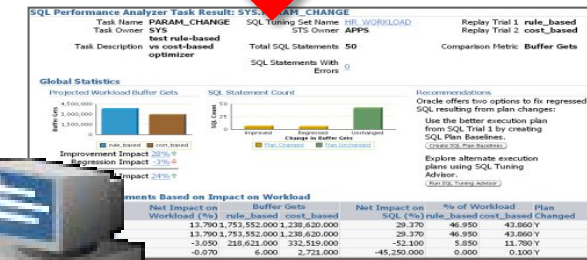
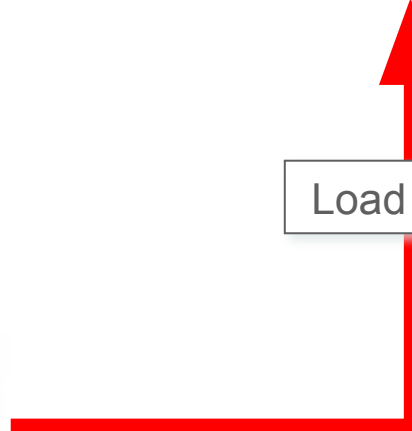


Evaluate, compare and tune –  
over and over again!

Capture SQLs  
into a STS



Load STS



SQL Performance Analyzer



# Use the Right Tools: SPA Regressed Report

## Projected Workload Change Impact:

Overall Impact : 0%  
 Improvement Impact : 0%  
 Regression Impact : 0%

## SQL Statement Count

SQL Category	SQL Count	Plan Change Count
Overall	3552	1593
Improved	658	410
Regressed	99	45
Unchanged	1754	1128
with Errors		

## SQL Statement

object_id	sql
7277	a304c09gqxxf3
8144	f26

## Execution Plan Before Change:

Plan Hash Value : Unknown

Id	Operation	Name	Impact on SQL	% Before
0	SELECT STATEMENT			
1	COUNT STOPKEY			
2	VIEW			
3	SORT ORDER BY STOPKEY			
4	NESTED LOOPS			
5	TABLE ACCESS BY INDEX ROWID	BUCHUNGEN	1000%	
6	INDEX RANGE SCAN	I_EID_BUCHUNGEN	550%	
7	TABLE ACCESS BY INDEX ROWID	PERSON		
8	INDEX UNIQUE SCAN	SYS_C0010236		

## Execution Plan After Change:

Plan Id : 27959  
 Plan Hash Value : 4020578872

Id	Operation	Name
0	SELECT STATEMENT	
* 1	COUNT STOPKEY	
2	VIEW	
* 3	SORT ORDER BY STOPKEY	
4	NESTED LOOPS	
5	NESTED LOOPS	
* 6	TABLE ACCESS BY INDEX ROWID	BUCHUNGEN
* 7	INDEX RANGE SCAN	I_EID_BUCHUNGEN
* 8	INDEX UNIQUE SCAN	SYS_C0012673
* 9	TABLE ACCESS BY INDEX ROWID	PERSON

## SQL Details:

Object ID : 7277  
 Schema Name : UHRZS006  
 SQL ID : a304c09gqxxf3  
 Execution Frequency : 1


SQL Text : select a,b,c from ( select ware a,kommentar b, p.nachname c from and eid = 349905 and kommentar like 'show\_user%' and b.ware = 60

## Execution Statistics:

Stat Name	Impact on Workload	Value Before	Value After	Impact on SQL	% Workload Before	% Workload After
elapsed_time	-1,61%	,022	1,206	-5381,82%	,03%	4,08%
parse_time			,001			,02%
cpu_time	-,4%	,02	,22	-1000%	,04%	1,3%
buffer_gets	-,01%	1721	1802	-4,71%	,2%	,28%
cost			7			0%
reads	-20,66%	0	2215	-221500%	0%	7,09%
writes	0%	0	0	0%	0%	0%
rows		36	33			

# Use the Right Tools: Use SPA in the Cloud to test the Upgrade

AWR  
(Diagnostics Pack)  
or SQL  
Tuning Set  
On Premises



SQL  
Performance  
Analyzer  
--- it's  
included



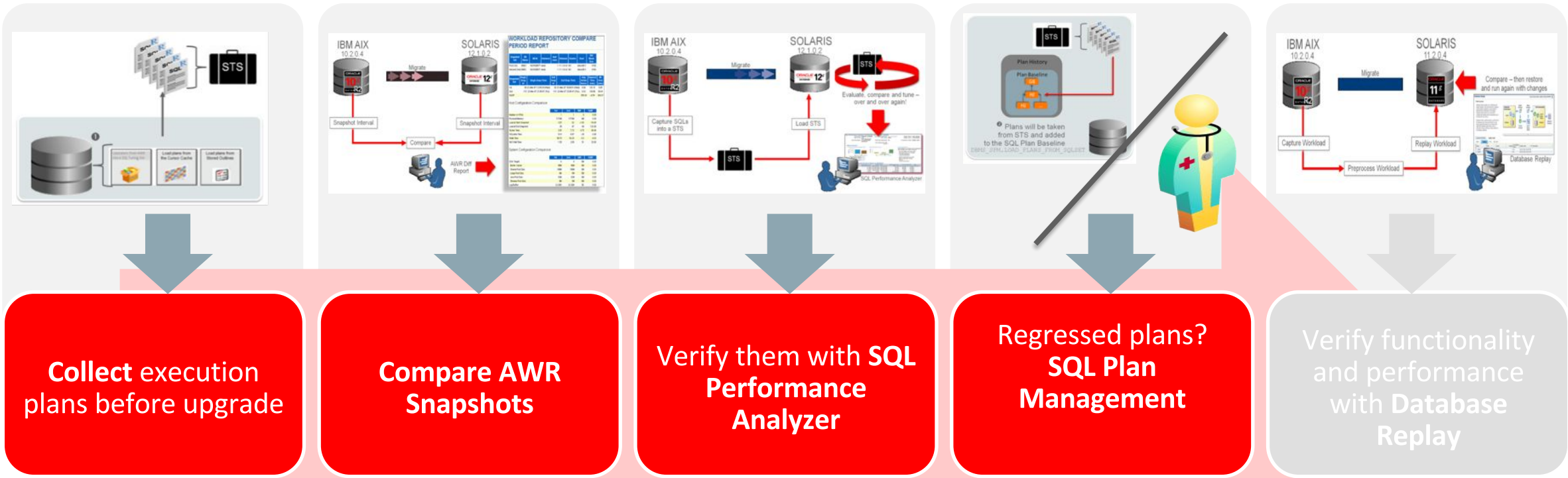
Oracle-Cloud

EE High Performance

EE Extreme Performance

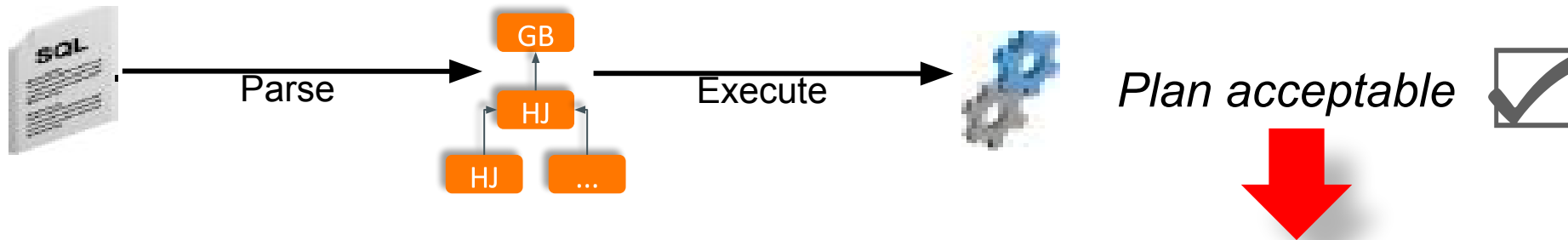


# Testing Tools – Hand-in-Hand

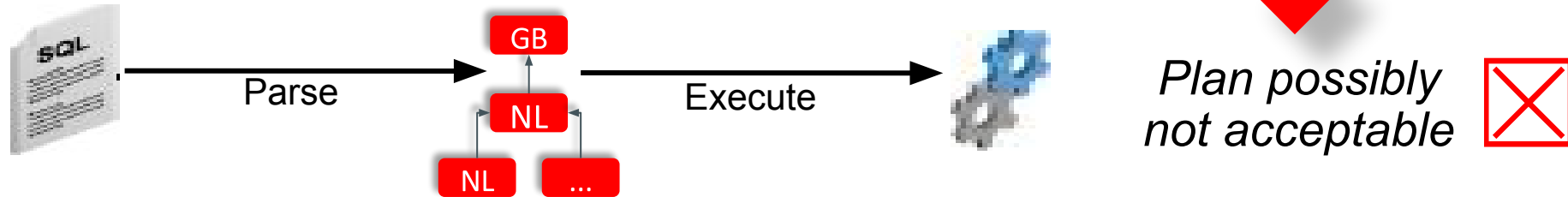


# Typical situation **after a change**

- Challenging to "freeze" execution plans



Conditions change:  
New statistics, parameter change, upgrade

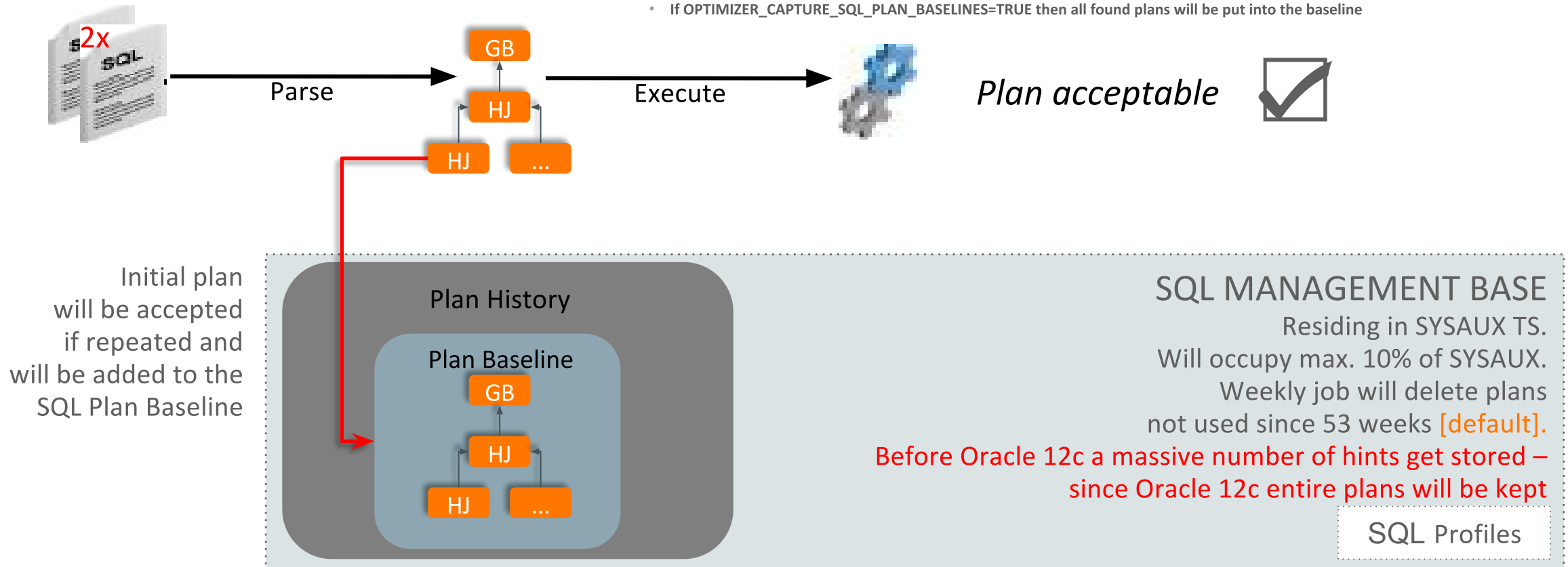


# SQL Plan Management - Mechanism

## Phase 1 – Baseline Capture

– Set `OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES=TRUE`

• If `OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES=TRUE` then all found plans will be put into the baseline

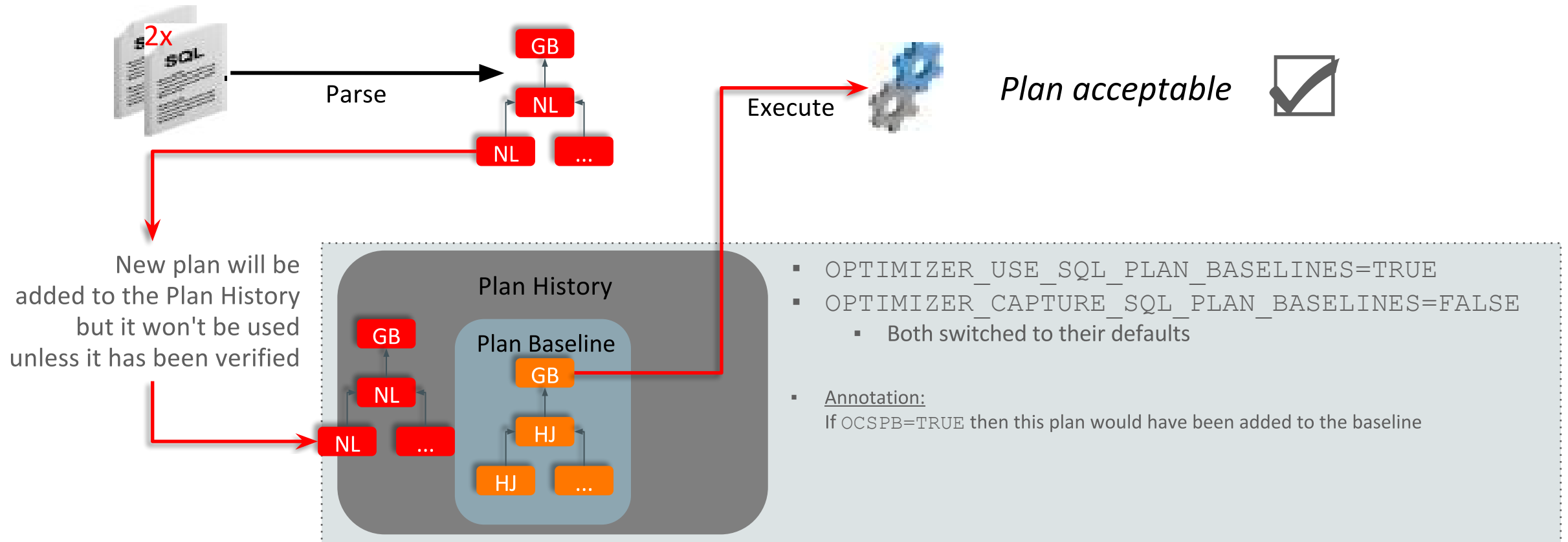




# SQL Plan Management - Mechanism

## Phase 2 - Selection

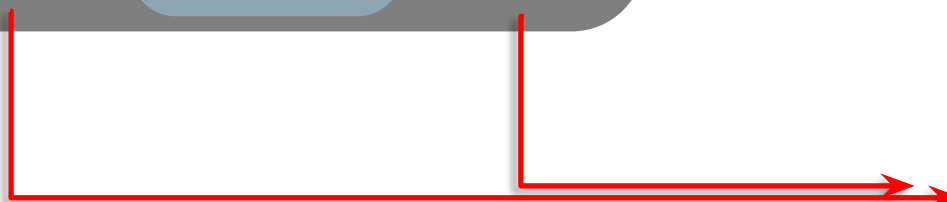
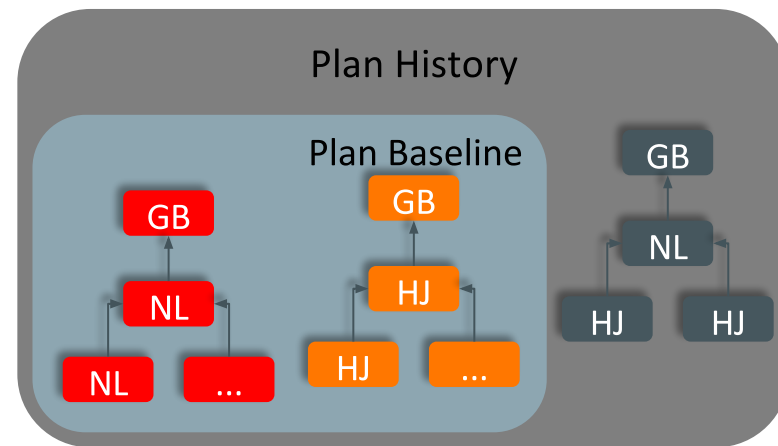
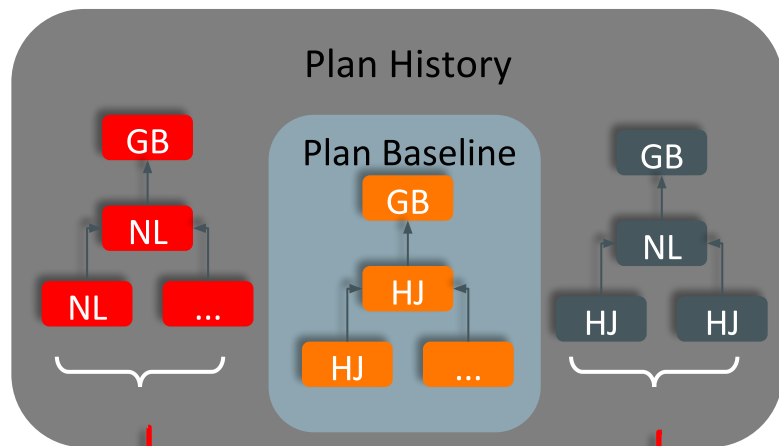
- Same statement parsed again but a different plan will be created



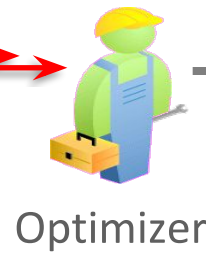
# SQL Plan Management - Mechanism

## Phase 3 - Evolution

– Since Oracle 12.1.0.2:



New  
`SYS_AUTO_SPM_EVOLVE_TASK`  
job as part of the *Automatic SQL*

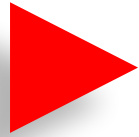


Optimizer

# SQL Plan Management

- Configure **SQL Plan Management (SPM)**

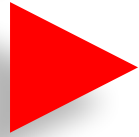
- Check settings:



```
SQL> select PARAMETER_NAME, PARAMETER_VALUE  
from DBA_SQL_MANAGEMENT_CONFIG;
```

- Change retention:

- Default: 53 weeks

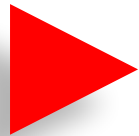


```
SQL> exec  
DBMS_SPM.CONFIGURE ('plan_retention_weeks', 5);
```

- Change space consumption:

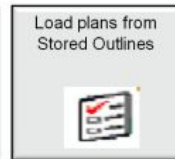
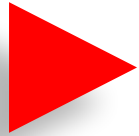
- Default: 10% of SYSAUX

- Plans will be stored in a LOB

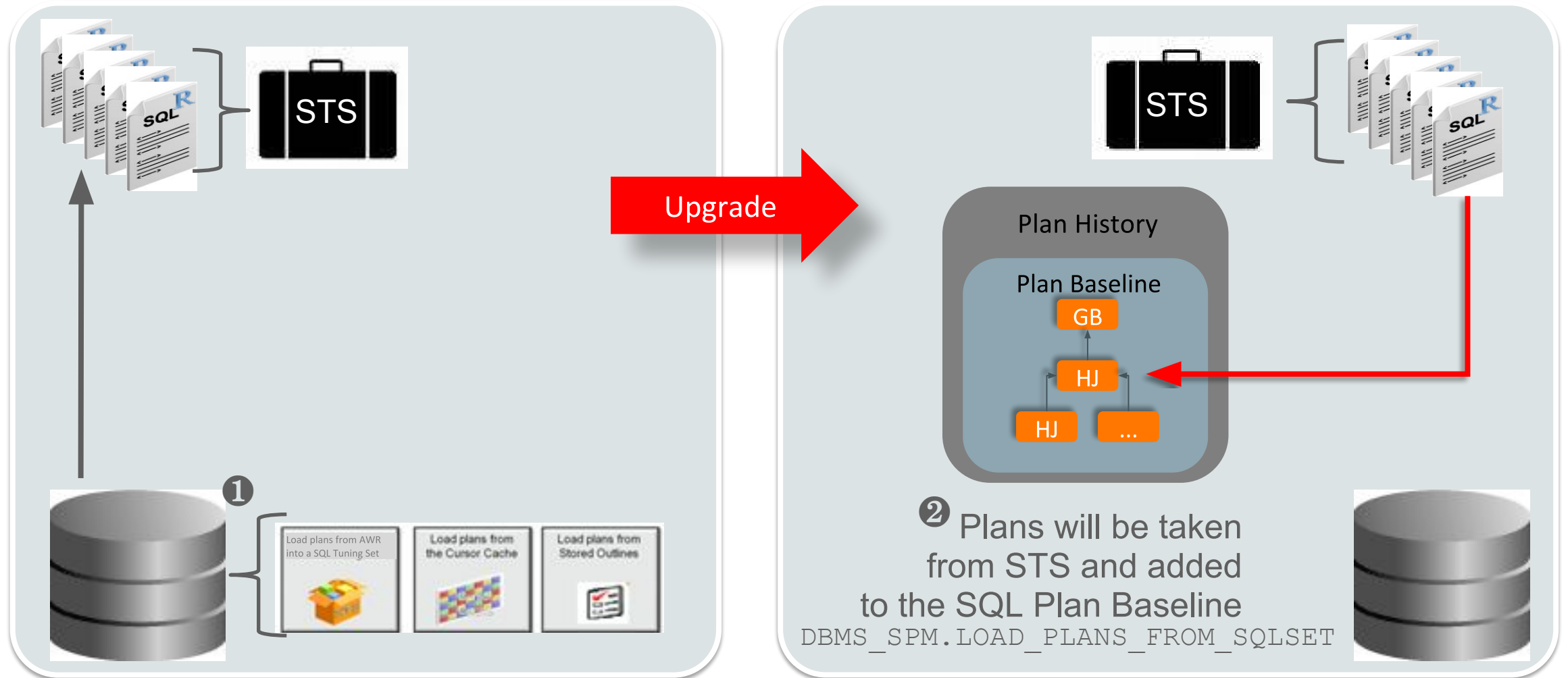


```
SQL> exec  
DBMS_SPM.CONFIGURE ('space_budget_percent', 5);
```

- Sources to load plans from:



# SPM – Plan Stability using Plan Capture



# SPM – Plan Stability using Plan Capture

- Upgrade scenario example: Capturing from AWR

```
exec DBMS_SQLTUNE.CREATE_SQLSET('MD_STS','SYS')
/
DECLARE
  cur sys_refcursor;
BEGIN
open cur for
  select value(p) from
table(dbms_sqltune.select_workload_repository(
  begin_snap => 4711,
  end_snap => 4788,
  basic_filter => 'parsing_schema_name not in
                  (''DBSNMP'', ''SYS'', ''ORACLE_OCM'')',
  ranking_measure1 => 'elapsed_time',
  result_limit => 250)) p;
  dbms_sqltune.load_sqlset('MD_STS', cur);
close cur;
END;
/
```

# SPM – Plan Stability using **Plan Capture**

- Upgrade scenario example: **"Fix" a plan in the baseline**
  - Prevent the plan from being aged out after `plan_retention_weeks` of non-use

```
DECLARE
  my_plans PLS_INTEGER;
BEGIN
  my_plans := DBMS_SPM.LOAD_PLANS_FROM_SQLSET (
    sqlset_name => 'MD_STS',
    basic_filter => 'sql_id="b25h7qc53gowp"',
    fixed => 'YES');
END;
/
```



# SPM – Plan Stability using **Plan Capture**

- [MOS Note: 789888.1](#)  
[How to Load SQL Plans into SQL Plan Management \(SPM\) from the Automatic Workload Repository \(AWR\)](#)

## ☆ [How to Load SQL Plans into SQL Plan Management \(SPM\) from the Automatic Workload Repository \(AWR\) \(Doc ID 789888.1\)](#)

### In this Document

[Goal](#)

[Solution](#)

[1. Populate the STS using DBMS\\_SQLTUNE.SELECT\\_WORKLOAD\\_REPOSITORY](#)

[Example 1](#)

[Example 2](#)

[Example 3](#)

[Verify how many sqls got loaded in the STS.](#)

[Verify the sql statements and its sql\\_id in the STS](#)

[Verify the execution Plan of a SQL\\_ID in the STS for an user sql](#)

[Verify the Plan baseline to check how many plans before](#)

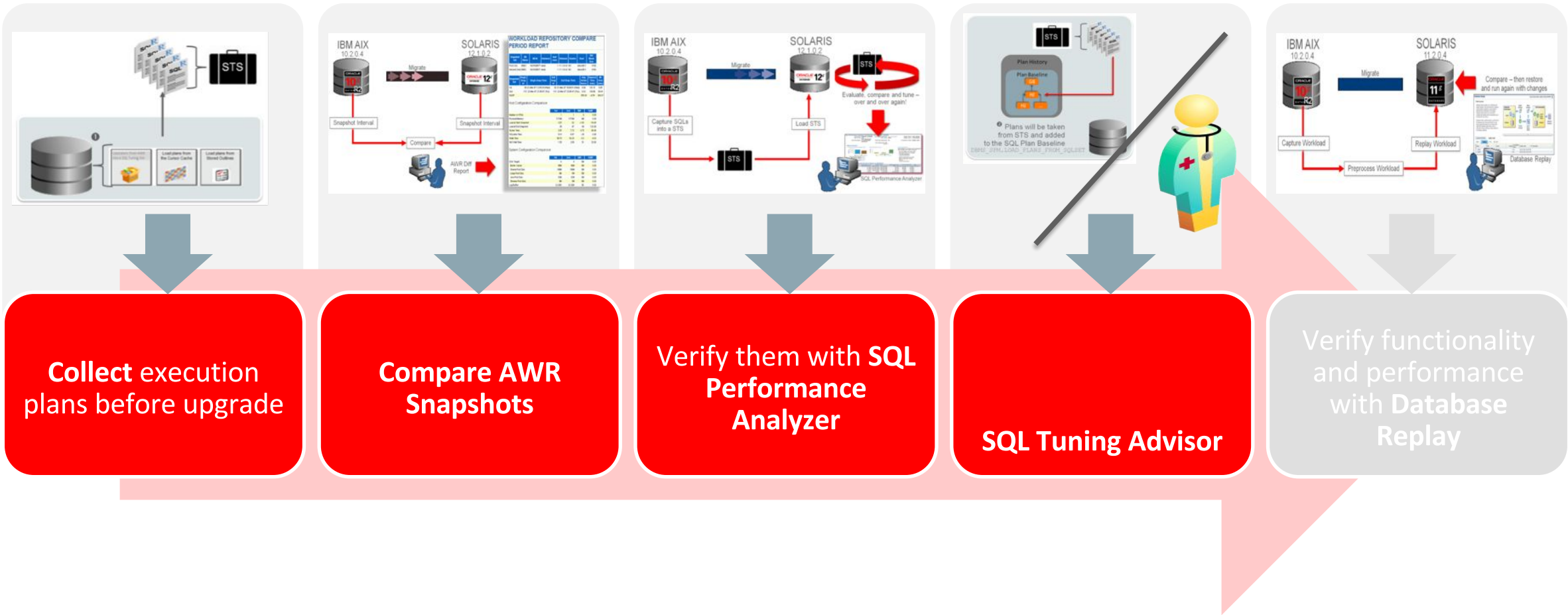
[2. Load the Sql Plan Baseline from STS](#)

[Verify the Plan baseline to check how many plans in plan baseline](#)

[Patches](#)

[References](#)

# Testing Tools – Hand-in-Hand



**Collect** execution plans before upgrade

**Compare** AWR Snapshots

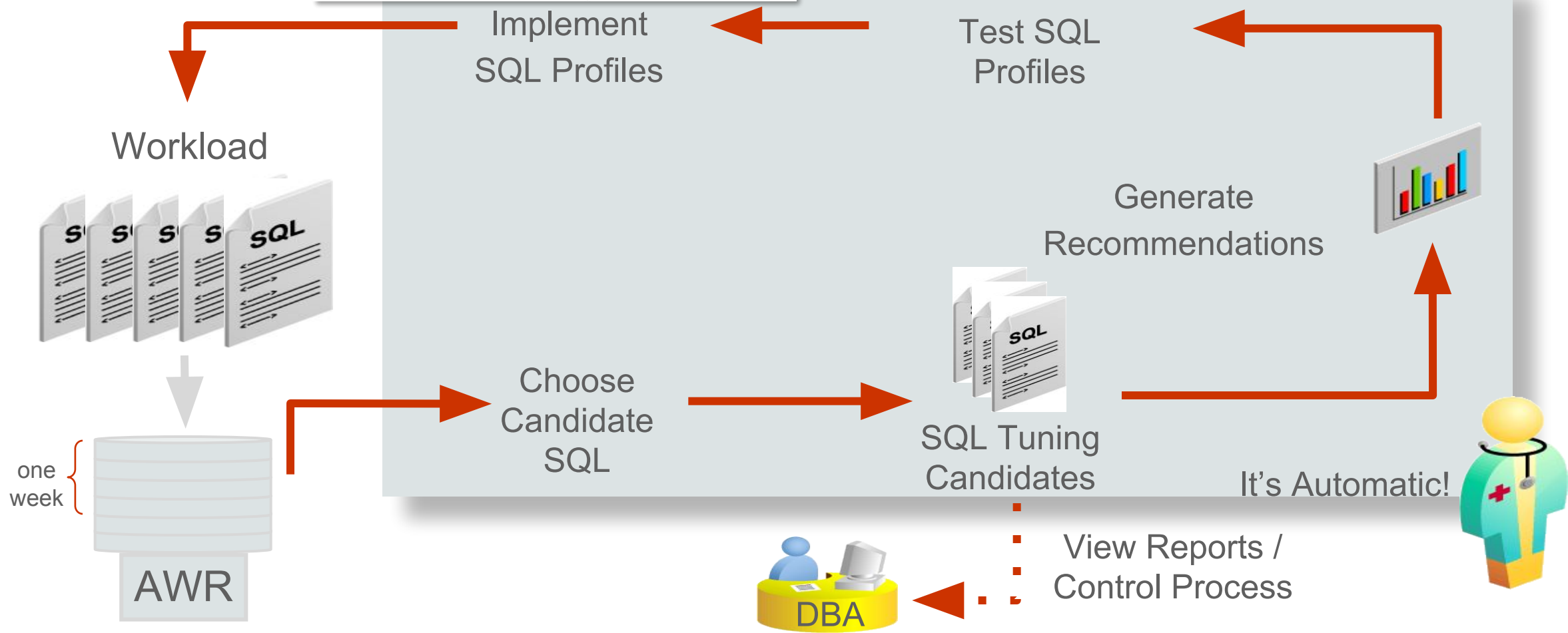
**Verify** them with **SQL Performance Analyzer**

**SQL Tuning Advisor**

**Verify** functionality and performance with **Database Replay**

# SQL Tuning Automation since Oracle Database 11g

Maximum Time Spent Per SQL During Tuning (sec)	1200
Automatic Implementation of SQL Profiles	<input checked="" type="radio"/> Yes <input type="radio"/> No
Maximum SQL Profiles Implemented Per Execution	20
Maximum SQL Profiles Implemented (Overall)	10000



# SQL Profiles

- SQL Profiles provide more information to the optimizer
  - Part of the Tuning Pack
  - Interface: GUI in EMCC or CLI with DBMS\_SQLTUNE
  - Don't change the application code
  - Persistent, transportable ([MOS Note:751068.1](#)), can be tested without risk
- See also [AskTom about SQL Profiles](#)



# SQL Tuning Example

1. Problem to be solved
2. Select candidate statement to tune
3. Get SQL statement details
4. Create and run SQL Tuning task
5. View report results

# SQL Tuning Example

## 1. Problem to be solved

We should be trying to understand why the export of statistics to the stats table took 1.5 hrs, but the import took 48 hours to complete. Instead, it appears that the SR engineer wants to create some type of work-around situation.

I would think we would want to trace the import stats table process to determine why it is taking so long. The stats table that was created only contains 2.8 GBs of data, which should be able to be loaded in mins....

Your help to keep this on track is appreciated...



# SQL Tuning Example

## 2. Select candidate statement to tune

- Generate an AWR Report

### SQL ordered by Elapsed Time

- Resources reported for PL/SQL code includes the resources used by all SQL statements called by the code.
- % Total DB Time is the Elapsed Time of the SQL statement divided into the Total Database Time multiplied by 100
- %Total - Elapsed Time as a percentage of Total DB time
- %CPU - CPU Time as a percentage of Elapsed Time
- %IO - User I/O Time as a percentage of Elapsed Time
- Captured SQL account for 100.3% of Total DB Time (s): 3,679
- Captured PL/SQL account for 99.0% of Total DB Time (s): 3,679

Elapsed Time (s)	Executions	Elapsed Time per Exec (s)	%Total	%CPU	%IO	SQL Id	SQL Module	SQL Text
3,606.81	0		98.03	99.28	0.02	<u>f344p5b5rrn81</u>	SQL*Plus	BEGIN DBMS_STATS_IMPORT_DATABA...
1,772.44	74	23.95	48.17	99.73	0.00	<u>f4k19gvr3nu38</u>	SQL*Plus	insert into sys.dbms_stats_id...
889.66	74	11.75	23.64	99.63	0.00	<u>mrkzynzfv5v1</u>	SQL Plus	insert into sys.dbms_stats_id...
792.05	68	11.65	21.53	99.87	0.00	<u>7c6w10f79j6g3</u>	SQL*Plus	insert into sys.dbms_stats_id...
65.28	4	16.32	1.77	40.00	69.02	<u>bm6v0v6m643m0</u>	salplus@edwdevdbadm01.humana.com (TNS V1-V3)	select owner . sum(bvtes)/1024...

# SQL Tuning Example

## 3. Get SQL Statement Details

```
DBMS_XPLAN.BUILD_PLAN_XML(TABLE_NAME=>gv$sql_plan, PLAN_TAG=>plan, FILTER_PRED=>:B35, FORMAT=> -  
PROJECTION +ALIAS +ADAPTIVE') ELSE NULL END XPLAN_XML FROM DUAL) V1) CONST_VIEW
```

```
dnak3w997p17j update tabpart$ set dataobj# = :1, part# = :2, ts# = :3, file# = :4, block# = :5, pctfree$ = :6, pctused$ = :7, initrans = :8, maxtrans = :9, flags  
= :10, analyzetime = :11, samplesize = :12, rowcnt = :13, blkcnt = :14, empcnt = :15, avgspc = :16, chncnt = :17, avgrln = :18 where obj# =  
:19
```

```
f344p5b5rm81 BEGIN DBMS_STATS.IMPORT DATABASE STATS(stattab => 'STATS'); END;
```

```
f4k19gvr3nu38 insert into sys.dbms_stats_id_map_tab (c5, c1, c2, cn) select distinct s.c5, s.c1, s.c2, d.partition_name cn from "SYSTEM"."STATS" s,  
(select u.name table_owner, op.name table_name, op.subname partition_name, tp.part# partition_position from user$ u, obj$ op, (select  
obj#, part# from tabpartv$ union all select obj#, part# from tabcompartv$ ) tp where u.user# = op.owner# and op.type# = 19 and op.obj# =  
tp.obj# ) d where s.c5 = :1 and s.c1 = :2 and s.type in ('T', 'C', 'E', 'P', 'H', 'B', 't', 'c', 'M', 'U', 'G', 'L') and s.n13 is not null and s.c2 is not null  
and s.c3 is null and s.c5 = d.table_owner and s.c1 = d.table_name and s.n13 = d.partition_position and s.c2 != d.partition_name and  
s.statid is null
```

```
fcj8q52nqgfc5 update indcompart$ set part# = :1, subpartcnt = :2, flags = :3, defts# = :4, defpctfree = :5, defnitrans = :6, defmaxtrans = :7, definiexts =  
:8, defextsize = :9, defminexts = :10, defmaxexts = :11, defextpct = :12, deflists = :13, defgroups = :14, defbufpool = :15, deflogging = :16,  
analyzetime = :17, samplesize = :18, rowcnt = :19, blevel = :20, leafcnt = :21, distkey = :22, lblkkey = :23, dblkkey = :24, clufac = :25,  
spare2 = :26, spare3 = :27, defmaxsize = :28 where obj# = :29
```

```
fdzqjimpvd6hvy SELECT O.DATAOBJ# FROM SYS.OBJ$ O WHERE O.OBJ# = :B1
```

# SQL Tuning Example

## 4. Create and run SQL Tuning task

```
VARIABLE stmt_task VARCHAR2(64);  
  
EXEC :stmt_task := DBMS_SQLTUNE.CREATE_TUNING_TASK(sql_id => 'f4k19gvr3nu38');  
EXEC DBMS_SQLTUNE.EXECUTE_TUNING_TASK(:stmt_task);  
  
SET LONG 10000  
SET LONGCHUNKSIZE 10000  
SET LINESIZE 100  
SPOOL sql_tune_f4k19gvr3nu38.txt  
SELECT DBMS_SQLTUNE.REPORT_TUNING_TASK( :stmt_task ) FROM DUAL;  
SPOOL OFF;
```



# SQL Tuning Example

## 5. View report results

```
DBMS_SQLTUNE.REPORT_TUNING_TASK(:STMT_TASK)
```

```
Schema Name: SYS
```

```
SQL ID      : f4k19gvr3nu38
```

```
SQL Text    :      insert into sys.dbms_stats_id_map_tab (c5, c1, c2, cn)
                select distinct s.c5, s.c1, s.c2, d.partition_name cn
                from "SYSTEM"."STATS" s,
                (select u.name table_owner, op.name table_name,
                    op.subname partition_name,
                    tp.part# partition_position
                from user$ u, obj$ op,
                (select obj#, part# from tabpartv$
                 union all
                 select obj#, part# from tabcompartv$
                ) tp
                where u.user# = op.owner# and op.type# = 19 and op.obj# = tp.obj#
                ) d
                where s.c5 = :1 and s.c1 = :2
                and s.type in ('T','C','E','P','H','B','t','c','M','U','G','L')
                and s.n13 is not null and s.c2 is not null and s.c3 is null
                and s.c5 = d.table_owner and s.c1 = d.table_name
                and s.n13 = d.partition_position
                and s.c2 != d.partition_name and s.statid is null
```

```
Bind Variables :
```

```
1 - (VARCHAR2(32)):STG
```

```
2 - (VARCHAR2(32)):MTH_MBR_COV_PLAN_PCP_STG
```

# SQL Tuning Example

## 5. View report results

-----  
FINDINGS SECTION (8 findings)  
-----

1- Statistics Finding  
-----

Optimizer statistics for table "SYS"."TABPART\$" and its indices are stale.

Recommendation  
-----

- Consider collecting optimizer statistics for this table.  
execute dbms\_stats.gather\_table\_stats(ownname => 'SYS', tabname =>  
    'TABPART\$', estimate\_percent => DBMS\_STATS.AUTO\_SAMPLE\_SIZE,  
    method\_opt => 'FOR ALL COLUMNS SIZE AUTO');

Rationale  
-----

The optimizer requires up-to-date statistics for the table in order to select a good execution plan.



# Pending Statistics

## Switch on Pending Statistics:

```
SQL> exec  
DBMS_STATS.SET_GLOBAL_PREFS('PENDING','TRUE');
```



## Gather new Oracle 12c statistics as "pending":

```
SQL> exec  
DBMS_STATS.GATHER_TABLE_STATS('SYS','TABPART$');
```



## Verify critical statements using "pending" 12c statistics:

```
SQL> alter session set  
optimizer_use_pending_statistics=TRUE;
```



## Publish new statistics if everything is fine:

```
SQL> exec DBMS_STATS.PUBLISH_PENDING_STATS();
```

# SQL Tuning Example

## 5. View report results

6- SQL Profile Finding (see explain plans section below)

A potentially better execution plan was found for this statement.

Recommendation (estimated benefit: 67.2%)

- Consider accepting the recommended SQL profile.  
`execute dbms_sqltune.accept_sql_profile(task_name => 'TASK_21944',  
task_owner => 'SYS', replace => TRUE);`

Validation results

The SQL profile was tested by executing both its plan and the original plan and measuring their respective execution statistics. A plan may have been only partially executed if the other could be run to completion in less time.

	Original Plan	With SQL Profile	% Improved
Completion Status:	COMPLETE	COMPLETE	
Elapsed Time (s):	2.588553	.802211	69 %
CPU Time (s):	2.57261	.799878	68.9 %
User I/O Time (s):	.000557	0	100 %
Buffer Gets:	182336	59805	67.2 %
Physical Read Requests:	7	0	100 %
Physical Write Requests:	0	0	
Physical Read Bytes:	114688	0	100 %
Physical Write Bytes:	0	0	
Rows Processed:	0	0	
Fetches:	0	0	
Executions:	1	1	

# SQL Profile - evaluation

- SQL Profiles should be evaluated before making them available to every user:

```
exec
: p_name:=dbms_sqltune.accept_sql_profile
task_name=>'TASK_21944', name=>'XT_PROFILE',
category=>'TEST_ENV', FORCE_MATCH=>TRUE)
```

- Now evaluate the statement's profile in a limited user context

```
alter session set SQLTUNE_CATEGORY='TEST_ENV';
```

- If verification went fine, make it accessible to everybody

```
exec
dbms_sqltune.alter_sql_profile
(name=>'XT_PROFILE',
attribute_name=>'CATEGORY', value=>'DEFAULT')
```

# SQL Tuning Example

## 5. View report results

7- Index Finding (see explain plans section below)

---

The execution plan of this statement can be improved by creating one or more indices.

DBMS\_SQLTUNE.REPORT\_TUNING\_TASK(:STMT\_TASK)

---

Recommendation (estimated benefit: 88.23%)

---

- Consider running the Access Advisor to improve the physical schema design or creating the recommended index.

```
create index SYSTEM.IDX$$_55B80001 on SYSTEM.STATS("N13");
```

Rationale

---

Creating the recommended indices significantly improves the execution plan of this statement. However, it might be preferable to run "Access Advisor" using a representative SQL workload as opposed to a single statement. This will allow to get comprehensive index recommendations which takes into account index maintenance overhead and additional space consumption.



# SQL Tuning Example

## 5. View report results

### 8- Alternative Plan Finding

---

Some alternative execution plans for this statement were found by searching the system's real-time and historical performance data.

The following table lists these plans ranked by their average elapsed time. See section "ALTERNATIVE PLANS SECTION" for detailed information on each plan.

id	plan hash	last seen	elapsed (s)	origin	note
1	1434278210	2017-01-11/01:19:14	1.254	Cursor Cache	not reproducible
2	1201858690	2017-01-11/01:19:23	1.355	Cursor Cache	not reproducible
3	3167061724	2017-01-11/01:19:39	2.580	Cursor Cache	not reproducible
4	2779611207	2017-01-11/01:19:30	2.855	Cursor Cache	not reproducible
5	2908117100	2017-01-11/05:00:47	3.093	Cursor Cache	not reproducible

### Information

---

- All alternative plans other than the Original Plan could not be reproduced in the current environment.
- The plan with id 1 could not be reproduced in the current environment. For this reason, a SQL plan baseline cannot be created to instruct the Oracle optimizer to pick this plan in the future.
- The plan with id 2 could not be reproduced in the current environment. For this reason, a SQL plan baseline cannot be created to instruct the Oracle optimizer to pick this plan in the future.
- The plan with id 3 could not be reproduced in the current environment. For this reason, a SQL plan baseline cannot be created to instruct the



# SQL Tuning Example

## 6. Act on findings

- Follow 5 statistics recommendations to gather stats on 5 tables

Hi All,

Follow the action plan as below. The import\_database\_stats finish in **2hrs 11 min.**

```
CREATE INDEX STATS_IDX ON STATS(C1,C5,TYPE,STATID);
```

### 1- Statistics Finding

- Consider collecting optimizer statistics for this table.

```
execute dbms_stats.gather_table_stats(ownname => 'SYS', tabname =>
'TABPART$', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt => 'FOR ALL COLUMNS SIZE AUTO');
```

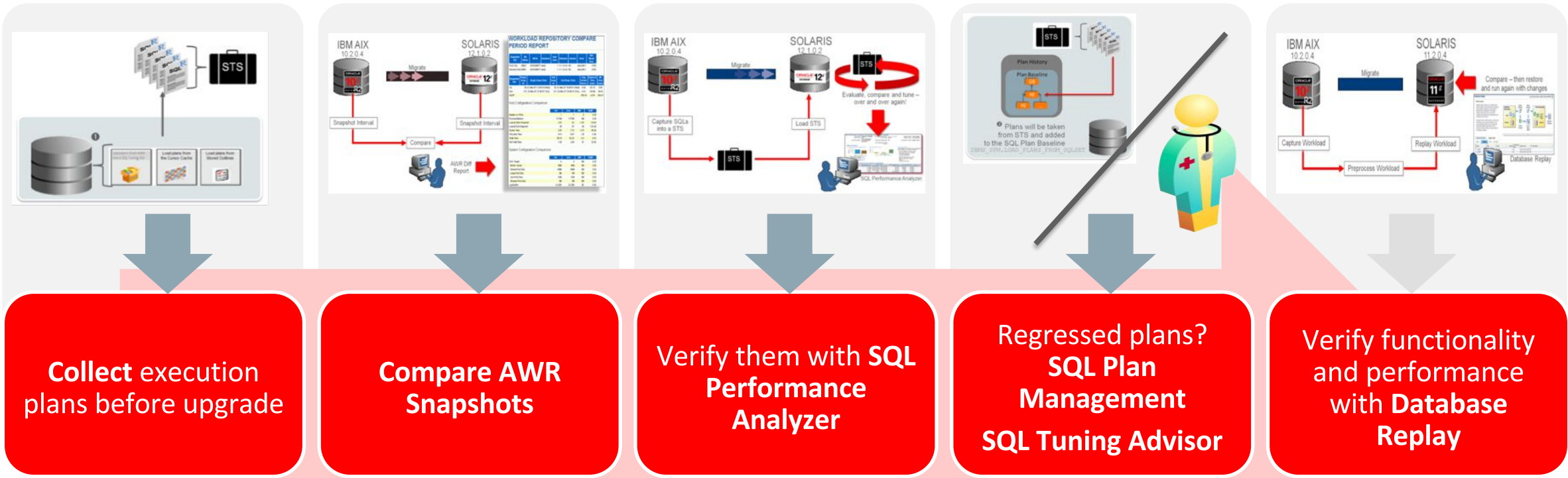
### 2- Statistics Finding

- Consider collecting optimizer statistics for this table.

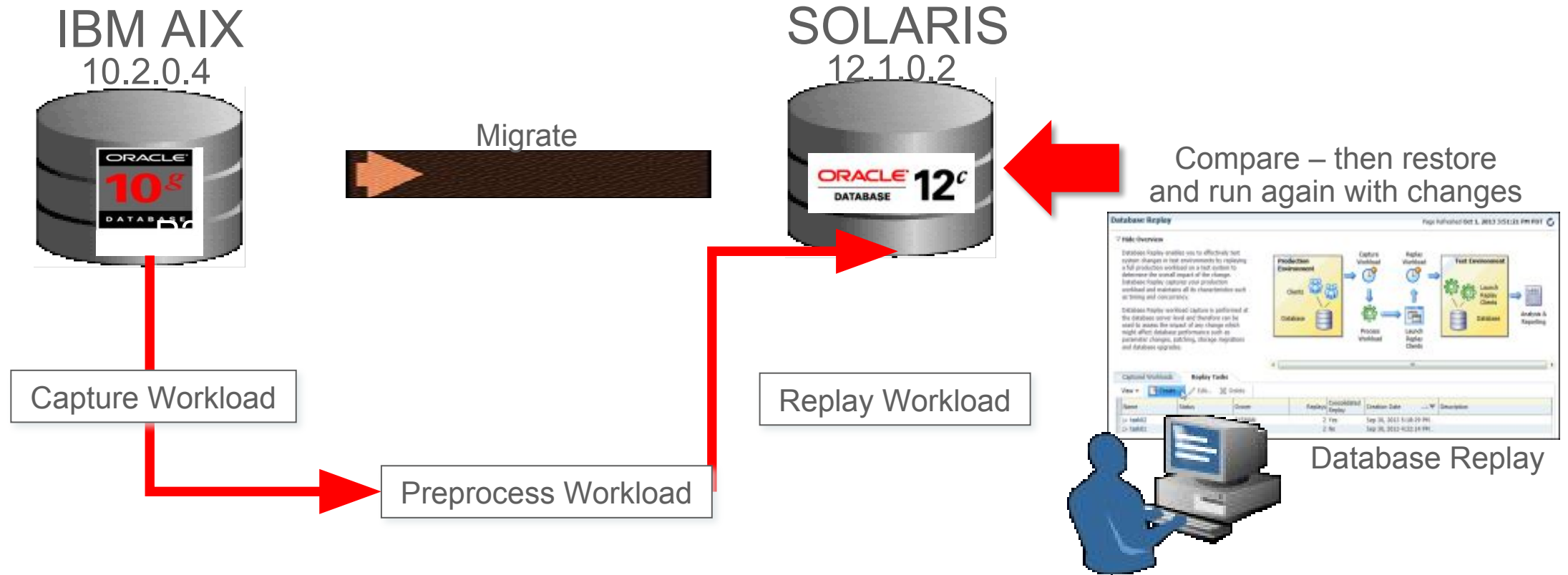
```
execute dbms_stats.gather_table_stats(ownname => 'SYS', tabname =>
'TABCOMPART$', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
```

- Result: **20x improvement!**

# Testing Tools – Hand-in-Hand



# Use the Right Tools: Database **Replay**





Now RELAX ... Stay Calm ...  
And open an SR with Oracle Support in case of real trouble

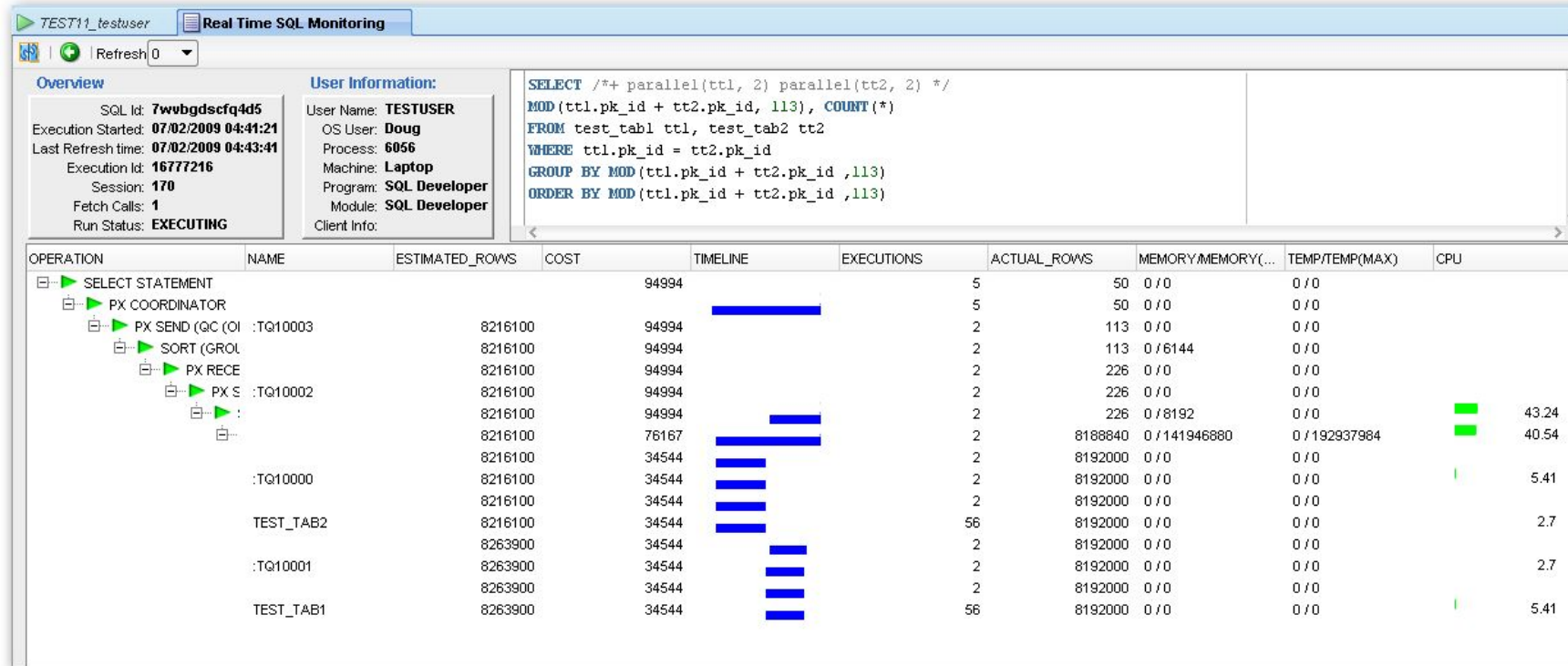
Picture taken from:

<https://aycw.wordpress.com/2016/06/10/young-peoples-attitudes-toward->



# SQL Real-Time Monitoring

- **The best way** to monitor the real execution plan
  - Part of Tuning Pack license – see also [MOS Note:1229904.1](#)



Source:

<http://oracledoug.com/serendipity/uploads/sqldev5.png>

- Or follow Database SQL Tuning Guide: [Reading Adaptive Query Plans](#)

# Performance Features



# Application Transparent Performance Features


- Free features:
  - SecureFile LOBs (11g)
  - Incremental Statistics (11g)
  - PL/SQL Native Compilation (since forever, but easier/cheaper since 11g)
  - New types of partitioning (practically every patch set)
  - New index features
- Options:
  - OLTP Compression (11g)

# Database Upgrade Blog - Slides

▪ <https://MikeDietrichDE.com/>

## Upgrade your Database – NOW!

Mike Dietrich's Oracle Database Upgrade Blog



Blog Slides Hands-On Lab Workshops Papers / Docs Videos Scripts Links About

### Slides Download Center

Edit

This page will be refreshed to a more user-friendly look&feel soon.

#### Comprehensive

- [Upgrade, Migrate & Consolidate to Oracle Database 12.2 & Cloud](#)  
*Updated: 26-FEB-2017*
- [Upgrade, Migrate & Consolidate to Oracle Database 12c](#)  
*Refreshed 3-DEC-2016*
- [Why you need to upgrade NOW!](#)

#### Recent Posts

- Hands-On Lab available: Upgrade to Oracle Database 12.2.0.1
- Collaborate Conference 2017 – Upgrade "Performance" Talk + Oracle Database 12.2 Hands-On Lab
- Multiple hop upgrades? Execute the matching preupgrade scripts for each hop
- Oracle Database 12.2.0.1 for Windows available



ORACLE®