

Toad Workshop Ljubljana 2020



Eero Mattila

Principal Systems Consultant

**Performance Management in
Oracle Enterprise and Standard Editions**

Quest™

Who am I?

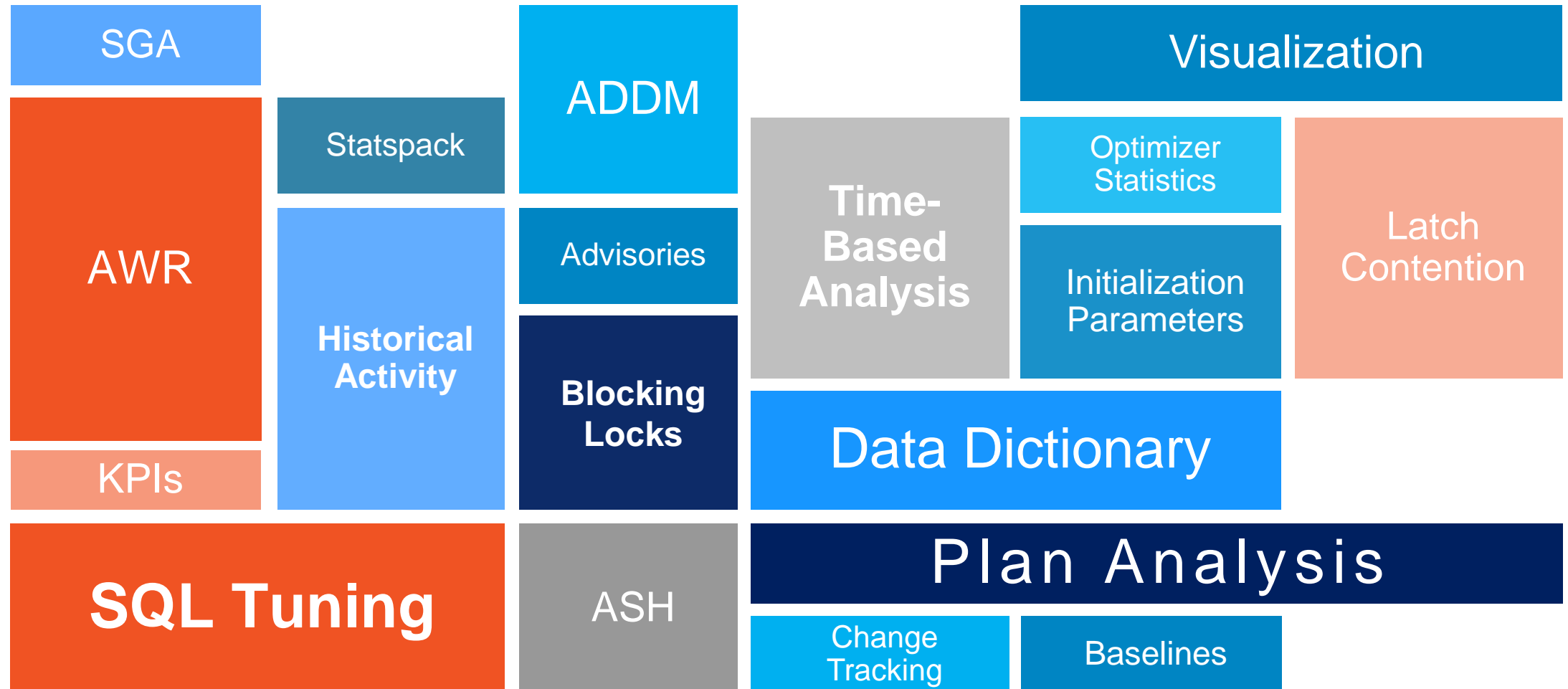
- Started with Oracle V6 in 1991
 - DBA, Forms/Reports Developer
- Oracle Germany 1995
 - RDBMS, Forms, Reports, Designer
- Quest Software since 2005
 - DB Administration and Development – Toad, Spotlight, SQL Optimizer
 - Database Replication – SharePlex
 - Performance Monitoring – Foglight

Agenda

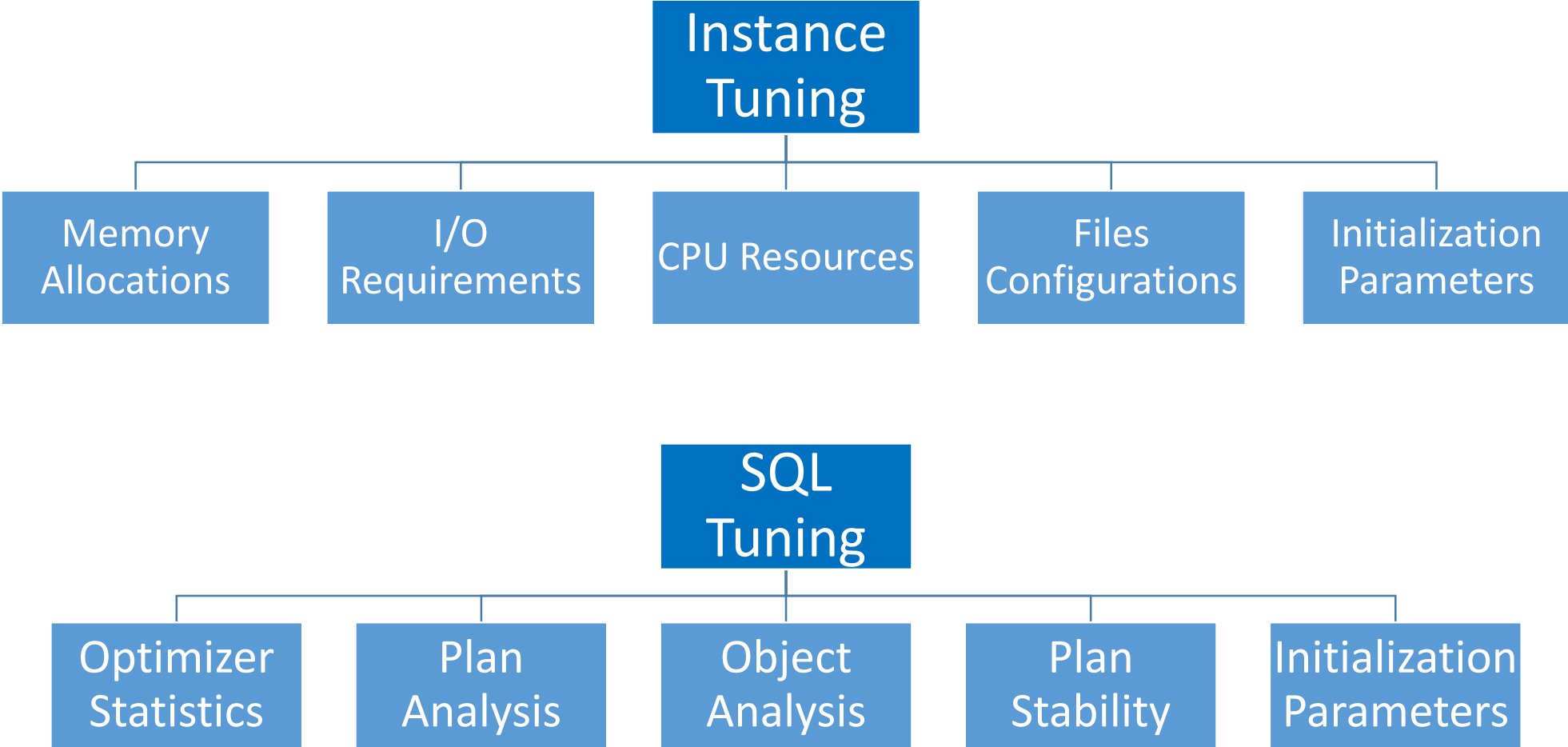


- ✓ **Introduction to Oracle database performance tuning**
- ✓ **Common challenges when using standard editions**
- ✓ **Toad DBA Edition w/ Spotlight**
- ✓ **Foglight for Databases**
- ✓ **Q&A**

Introduction to Performance Tuning



Common Performance Diagnostics Approaches



Instance Tuning - Common Things to Consider

- **Initialization Parameters** - (such as STATISTICS_LEVEL)
- **Memory structures** - ASMM, AMM, min size for buffer pools
- **Files configuration** - redo logs, undo tablespace
- **Hardware configuration** - memory, storage characteristics, network
- **ASM configuration** - disk groups and failure groups
- **Connection management** - both app level and DB level

SQL Tuning - Common Things to Consider

- **Bad SQLs** - using literals instead of bind variables
- **Object analysis** - Indexes, partitions, materialized views
- **Optimizer statistics** - last analyzed, dynamic statistics (OPTIMIZER_DYNAMIC_SAMPLING)
- **Oracle features** which may impact execution plans
 - SQL Profiles (10g)
 - Adaptive Cursor Sharing (11gR2)
 - Cardinality Feedback (11gR2)
 - Dynamic Statistics (12cR1)
 - Adaptive Plans (12cR1)
- **Plan stability** - SQL plan management (11gR1)

Simplified Performance Tuning Methodology

1. Determine the most significant bottleneck – i.e. best tuning opportunities
2. Improve/Fix it
3. Repeat it until performance is good

Pini Dibask's Blog Post:

<http://oracledbpro.blogspot.com/2015/09/simple-performance-tuning-methodology.html>

How can we find them with enterprise edition & diagnostics pack?

WORKLOAD REPOSITORY report for

✓ AWR

✓ ADDM

✓ ASH

✓ ASH Analytics

✓ SQL Tuning Advisory

DB Name	DB Id	Instance	Inst num	Startup Time	Release	RAC
ORCL11	4088687252	orcl11	1	16-Aug-15 10:08	11.2.0.3.0	NO

Host CPU (CPUs: 4 Cores: 4 Sockets: 2)

Host Name	Platform	CPUs	Cores	Sockets
ISRVMPALW06	Microsoft Windows x86 64-bit	4	4	1

Load Average Begin	Load Average End	%User	%System	%WIO	%Idle
		24.9	4.9		70.2

	Snap Id	Snap Time	Sessions	Cur
Begin Snap:	9253	30-Aug-15 15:00:02	91	
End Snap:	9254	30-Aug-15 16:00:37	92	
Elapsed:		60.59 (mins)		
DB Time:		179.46 (mins)		

Instance CPU

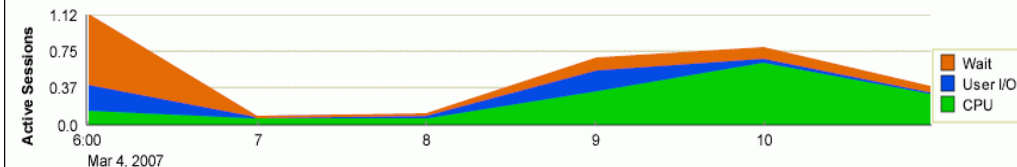
%Total CPU	%Busy CPU	%DB time waiting for CPU (Resource Manager)
1.3	4.4	0.0

Memory Statistics

	Begin	End
	8,191.6	8,191.6
	1,024.0	1,024.0
	781.3	784.8
	22.04	22.08

Database Activity

The icon selected below the graph identifies the ADDM analysis period. Click on a different icon to select a different analysis period.



TIP For an explanation of the icons and symbols used in this page, see the [Icon Key](#)

ADDM Performance Analysis

Task Name **ADDM:3216269853_1_56**

Time Range

[Filters](#) [View Snapshots](#) [View Report](#)

Task Owner **SYS**

Average Active Sessions **0.8**

Period Start Time **Mar 4, 2007 11:00:07 PM PST**

Period Duration **60.3 (minutes)**

Impact (%)	Finding	Occurrences (last 24 hrs)
100	Top SQL by DB Time	11 of 19
77.5	CPU Usage	12 of 19
9.5	Hard Parse	8 of 19
4.9	"Scheduler" Wait Class	12 of 19
1.4	I/O Throughput	11 of 19

Performance Tuning Challenges for DBAs

Feature	No Diagnostics Pack	Diagnostics Pack	Diagnostics & Tuning Packs
Statspack	✓	✓	✓
Dictionary Views	✓* (most of them...)	✓	✓
AWR	✗	✓	✓
ASH	✗	✓	✓
ASH Analytics	✗	✓	✓
ADDM/Compare Period ADDM	✗	✓	✓
SQL Tuning Advisor	✗	✗	✓
SQL Access Advisor	✗	✗	✓

So what are our options when diagnostics pack is not available?

- **Statspack** - available since Oracle 8i (1998)
- **Dictionary Views (below are my favorites...)**
 - DB statistics – V\$SESSTAT, V\$SYSSTAT
 - SQL statistics – V\$SQLAREA, V\$SQL
 - Execution plan information – V\$SQL_PLAN, V\$SQL_SHARED_CURSOR
 - Object level statistics – V\$SEGMENT_STATISTICS
 - File level statistics – V\$FILESTAT
 - Wait events – V\$SESSION, V\$SESSION_WAIT, V\$SYSTEM_EVENT
 - Time model statistics – V\$SYS_TIME_MODEL, V\$SES_TIME_MODEL

Statspack

- Oracle 8i and above
- "Poor man's AWR"
- Text based report - no easy navigation, no visualization
- Less comprehensive than AWR

Database	DB Id	Instance	Inst Num	Startup Time	Release	RAC
-----	-----	-----	-----	-----	-----	-----
	2915477242	prm11	1	15-Sep-18 01:37	11.2.0.3.0	NO
Host Name	Platform		CPUs	Cores	Sockets	Memory (G)
-----	-----		-----	-----	-----	-----
	ISRVMN879	Microsoft Windows x86	4	4	2	8.0
Snapshot	Snap Id	Snap Time	Sessions	Curs/Sess	Comment	
-----	-----	-----	-----	-----	-----	
Begin Snap:	1	04-Oct-18 16:38:36	92	2.3		
End Snap:	3	04-Oct-18 16:39:52	90	2.4		
Elapsed:	1.27 (mins)	Av Act Sess:	0.2			
DB time:	0.19 (mins)	DB CPU:	0.09 (mins)			
Cache Sizes	Begin		End			
-----	-----		-----			
Buffer Cache:	368M	Std Block Size:		8K		
Shared Pool:	592M	Log Buffer:		7,400K		
Load Profile	Per Second		Per Transaction	Per Exec	Per Call	
-----	-----		-----	-----	-----	
DB time(s):	0.2		0.9	0.00	0.01	
DB CPU(s):	0.1		0.4	0.00	0.01	
Redo size:	94,708.5		553,680.3			
Logical reads:	446.2		2,608.3			
Block changes:	182.8		1,068.5			
Physical reads:	18.0		105.2			
Physical writes:	3.8		22.0			
User calls:	13.4		78.3			
Parses:	4.8		28.0			
Hard parses:	1.9		10.9			
W/A MB processed:	1.6		9.6			
Logons:	0.1		0.5			
Executes:	31.6		184.9			
Rollbacks:	0.0		0.0			
Transactions:	0.2					
Instance Efficiency Indicators						

Buffer Nowait %:	99.99	Redo NoWait %:	100.00			
Buffer Hit %:	99.84	Optimal W/A Exec %:	100.00			
Library Hit %:	90.57	Soft Parse %:	61.26			
Execute to Parse %:	84.85	Latch Hit %:	100.00			
Parse CPU to Parse Elapsed %:	57.38	% Non-Parse CPU:	94.06			
Shared Pool Statistics						
		Begin	End			
		-----	-----			
Memory Usage %:	90.31	90.63				
% SQL with executions>1:	72.31	74.63				
% Memory for SQL w/exec>1:	86.45	87.60				

Using Dictionary Views - Example

2 • `SELECT *`
3 `FROM v$segment_statistics`
4 `WHERE object_name = 'ORDERS'`

Data Grid

Messages Data Grid Trace DBMS Output Query Viewer Explain Plan Script Output

	OWNER	OBJECT_NAME	SUBOBJECT_NAME	TABLESPACE_NAME	TS#	OBJ#	DATAOBJ#	OBJECT_TYPE	STATISTIC_NAME	STATISTIC#	VALUE
▶	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	logical reads	0	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	buffer busy waits	1	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	gc buffer busy	2	0
•	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	db block changes	3	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical reads	4	4
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical writes	5	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical read requests	6	4
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical write requests	7	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical reads direct	8	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	physical writes direct	9	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	optimized physical reads	11	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	gc cr blocks received	12	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	gc current blocks received	13	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	ITL waits	14	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	row lock waits	15	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	space used	17	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	space allocated	18	0
	SALES	ORDERS	OL200112	USERS	4	357057	357057	TABLE PARTITION	segment scans	20	1
	SALES	ORDERS	OL200111	USERS	4	357056	357056	TABLE PARTITION	logical reads	0	0
	SALES	ORDERS	OL200111	USERS	4	357056	357056	TABLE PARTITION	buffer busy waits	1	0

1 • `SELECT * FROM v$sys_time_model`

Data Grid

Messages Data Grid Trace DBMS Output Query Viewer Explain Plan

	STAT_ID	STAT_NAME	VALUE
	3649082374	DB time	168205069903
	2748282437	DB CPU	61760658004
	4157170894	background elapsed time	141748129666
	2451517896	background cpu time	15588771771
	4127043053	sequence load elapsed time	882877
	1431595225	parse time elapsed	2005849591
	372226525	hard parse elapsed time	1440327126
	2821698184	sql execute elapsed time	97699603724
	1990024365	connection management call elapsed time	607789686
	1824284809	failed parse elapsed time	534275694
	4125607023	failed parse (out of shared memory) elapsed time	0
	3138706091	hard parse (sharing criteria) elapsed time	450439555
	268357648	hard parse (bind mismatch) elapsed time	9147827
	2643905994	PL/SQL execution elapsed time	2449991737
	290749718	inbound PL/SQL rpc elapsed time	0
	1311180441	PL/SQL compilation elapsed time	89323501
	751169994	Java execution elapsed time	0
	1159091985	repeated bind elapsed time	9978508
▶	2411117902	RMAN cpu time (backup/restore)	0

What are the challenges with these approaches?

- **Statspack**

- Text format - less intuitive to navigate compared to AWR html format
- Not as feature-rich as AWR

- **Dictionary Views**

- Limited history
- Cumulative statistics - since startup

But most importantly... no visualization

What do these three things have in common?



They all provide a clearer perspective -

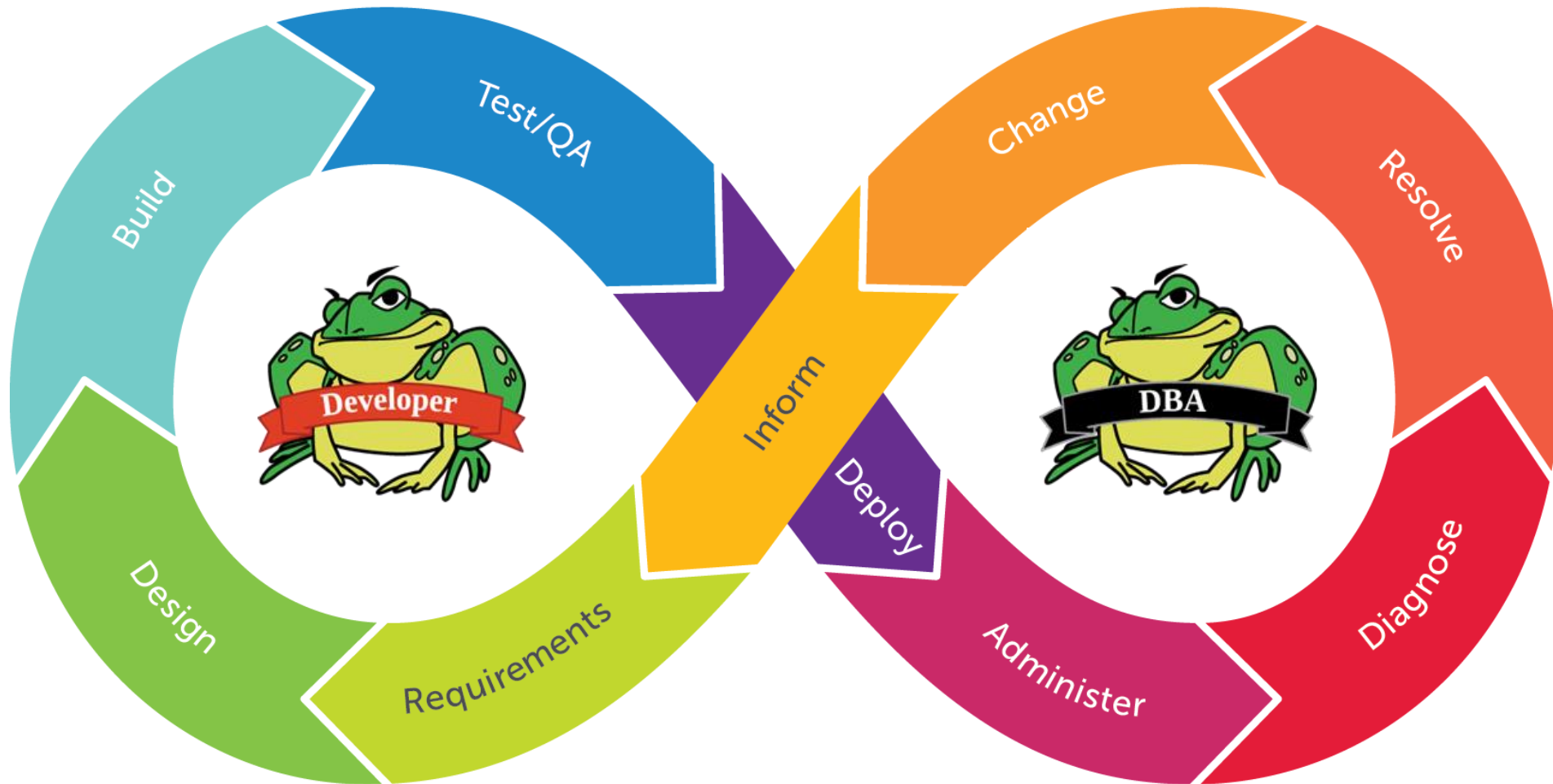
You can't fix what you don't see!

What if you could...

*Gain complete **visibility** into the **health**
and **performance** of your database
environments*



Toad for Oracle supports the database application lifecycle



Toad for Oracle Editions for DBAs



Toad for Oracle DB Admin Module:

Maintain database health and security, while minimizing the impact of changes. Perform database compare and synch, and schema compare across two databases simultaneously. Also, conduct database health checks (see Oracle options, packs, RAC and Exadata), objects and data.

What's included:

- Any Edition of Toad for Oracle is **required**
- Configurable database health checks

Toad DBA Edition for Oracle:

Assure maintenance, performance and change with a proactive approach through this complete comprehensive database administration toolset. Simplify administration tasks, and predict, diagnose and resolve database performance issues before end users are impacted.

What's included:

- Toad for Oracle Xpert Edition+DB Admin Module
- Spotlight™ on Oracle, Spotlight on MySQL, Spotlight on Unix/Linux, Windows, and Oracle Data Guard
- Toad Data Modeler
- Benchmark Factory (Oracle Edition)

Toad DBA Edition for Oracle – RAC Edition:

Simplify Oracle RAC administration and solve RAC performance bottlenecks quickly. Provides complete visibility of Oracle RAC architecture for effective real-time diagnostics and RAC scalability testing. Provides visibility into Oracle RAC at node, cluster and interconnect levels.

What's included:

- Toad for Oracle Xpert Edition + DB Admin Module
- Spotlight on Oracle RAC, Spotlight on MySQL, Spotlight on Unix/Linux, Windows, and Oracle Data Guard
- Toad Data Modeler
- Benchmark Factory (Oracle Edition)

Toad DBA Edition for Oracle – Exadata Edition:

Provides unique insight into Exadata. It's the only solution that shows a complete view of the entire environment. It allows you to identify and diagnose potential performance issues and bottlenecks before end users are impacted. This edition also provides checks to ensure optimal Exadata configuration.

What's included:

- Toad for Oracle Xpert Edition + DB Admin Module
- Spotlight on Oracle Exadata, Spotlight on MySQL, Spotlight on Unix/Linux, Windows, and Oracle Data Guard
- Toad Data Modeler
- Benchmark Factory (Oracle Edition)

Performance Management

- Highly visual representation of Oracle database architecture
- Detailed diagnostics screens enable full visibility of problems, together with immediate resolution





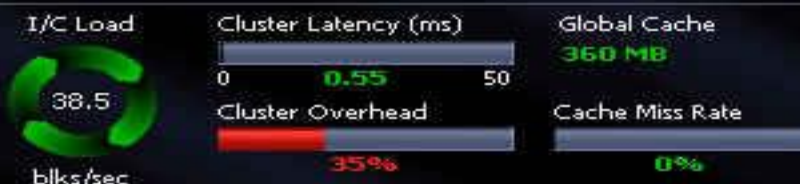
sim_RVDB.xml

Live connections

- cl92
- sim_RVDB.xml
- sim_RVDB.xml DE...
- sim_RVDB.xml DE...
- sim_RVDB.xml DE...

Interconnect and Global Status

Interconnect/GCS



Global Database



Instances

Balance

4%

Availability

100%

1 - 6

7 - 9

DEV101

CPU : 9%

Logical Reads

1.6E4

Sessions: 7

DEV102

CPU : 5%

Logical Reads

1346

Sessions: 5

DEV103

CPU : 6%

Logical Reads

41.1

Sessions: 6

DEV101

CPU : 7%

Logical Reads

3019

Sessions: 6

DEV102

CPU : 5%

Logical Reads

1041

Sessions: 5

DEV103

CPU : 6%

Logical Reads

45.7

Sessions: 6

IO Subsystem

ASM

Activity

0

IO/sec

Data Files

Phys Reads

445

IO/sec

Phys Writes

212

IO/sec

Single block re...
Service Time (ms)

0 to 20, current 4.24

Files: 16
Tablespaces: 13
Total: 8.03 GB

Redo Logs

Redo Writes

33.4

IO/sec

Groups (Avg/Max): 18/18
Size (MB) (Avg/Max): 10/10

Archive Logs

Total Size:
Minimum F...
Time to fail...
N/A (if not pur...

Performance Management – SQL Optimizer

- Proactively scan, identify and classify problematic application SQL directly from the source code

The screenshot displays the Quest SQL Optimizer 8.8.1 for Oracle interface. The main window shows a 'Job List' for the group 'QUEST_PERF programs (QUEST_PERF@ORCL)'. The list contains 20 rows of scanned SQL statements, each with a status of 'Completed' and a 'Date Scanned' of 9/1/2010. The 'Status' column includes a green checkmark icon. The 'User Comments' column is empty.

Path/Owner	Name of File/Object	Status	Set Schema	Elimi...	Probl...	Comp...	Simpl...	Inval...	Date Scanned	User Comments
QUEST_PERF	FUNC_FDATE	Completed	QUEST_PERF		0	4	3	0	7/9/2013 9:16...	
QUEST_PERF	FUNC_SAMPLE_7	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:45...	
QUEST_PERF	FUNC_SAMPLE_EMP...	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:45...	
QUEST_PERF	FUNC_SAMPLE_MA...	Completed	QUEST_PERF		0	0	1	0	9/1/2010 2:46...	
QUEST_PERF	FUNC_SAMPLE_NA...	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:46...	
QUEST_PERF	GET_AVG_JOB_SAL...	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:47...	
QUEST_PERF	EXAMPLE_PROC_IN...	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:47...	
QUEST_PERF	PROCEDURE_AX	Completed	QUEST_PERF		2	1	2	0	9/1/2010 2:47...	
QUEST_PERF	PROCEDURE_BX	Completed	QUEST_PERF		1	1	2	0	9/1/2010 2:49...	
QUEST_PERF	PROCEDURE_CX	Completed	QUEST_PERF		3	0	1	0	9/1/2010 2:50...	
QUEST_PERF	PROCEDURE_DX	Completed	QUEST_PERF		1	0	2	0	9/1/2010 2:51...	
QUEST_PERF	PROCEDURE_EX	Completed	QUEST_PERF		2	0	0	0	9/1/2010 2:51...	
QUEST_PERF	PROCEDURE_FX	Completed	QUEST_PERF		1	0	0	0	9/1/2010 2:52...	
QUEST_PERF	PROCEDURE_VIEW	Completed	QUEST_PERF		1	0	1	0	9/1/2010 2:52...	
QUEST_PERF	P_SAMPLE_DELETE	Completed	QUEST_PERF		2	0	3	0	9/1/2010 2:53...	
QUEST_PERF	P_SAMPLE_INSERT	Completed	QUEST_PERF		5	0	0	0	9/1/2010 2:55...	
QUEST_PERF	P_SAMPLE_SELECT	Completed	QUEST_PERF		3	1	1	0	9/1/2010 2:55...	
QUEST_PERF	P_SAMPLE_UPDATE	Completed	QUEST_PERF		1	0	3	0	9/1/2010 2:56...	
QUEST_PERF	P_SECOND_5	Completed	QUEST_PERF		3	0	0	0	9/1/2010 2:57...	
QUEST_PERF	P_SECOND_50	Completed	QUEST_PERF		1	0	1	0	9/1/2010 2:58...	
QUEST_PERF	P_SECOND_70	Completed	QUEST_PERF		0	1	0	0	9/1/2010 2:58...	

The bottom panel shows the 'SQL List' and 'SQL Text' tabs. The 'SQL List' tab displays a table with 4 rows of SQL statements, all classified as 'Problematic'.

SQL Name	SQL Classi...	Conversion	Cost
SQL 1	Problematic		3687
SQL 2	Problematic		5317
SQL 3	Problematic		3196
SQL 4	Problematic		690

The 'SQL Text' tab displays the SQL statement for SQL 1:

```
select emp_dept,
       max(emp_salary),
       min(sal_salary)
from v_emp_details,
     emp_sal_hist
where emp_id = sal_emp_id
      and emp_id in (select emp_id
                     from emp_small
                     where emp_dept like 'D%')
group by emp_dept
```

The 'SQL Information' tab shows the classification details for SQL 1:

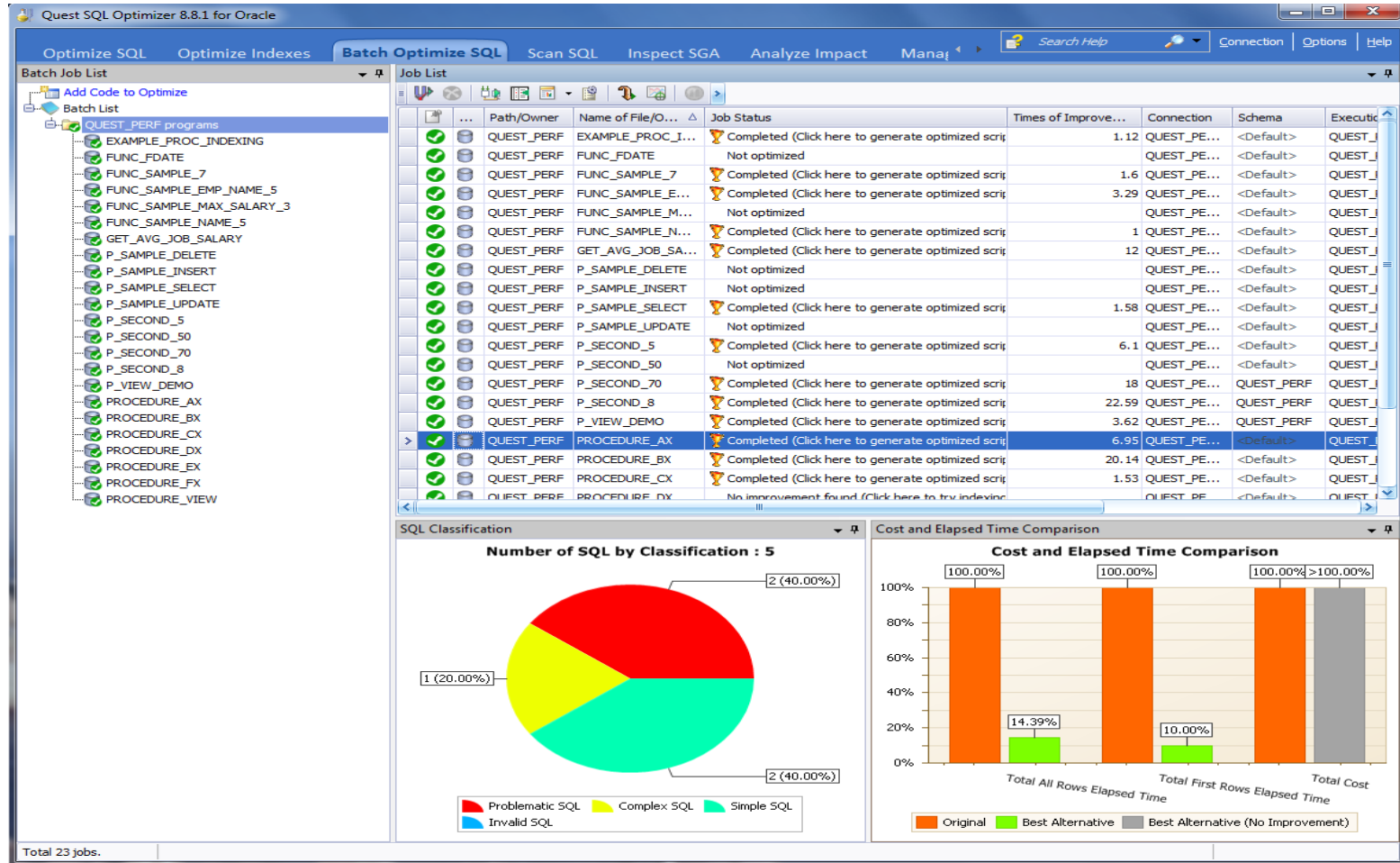
Classification	Conversion	PA Repository
SQL Name: SQL 1	SQL 1	
SQL Classification: Problematic		
Classification Rules:		

The 'Classification Rules Detail' section lists the rules that triggered the classification:

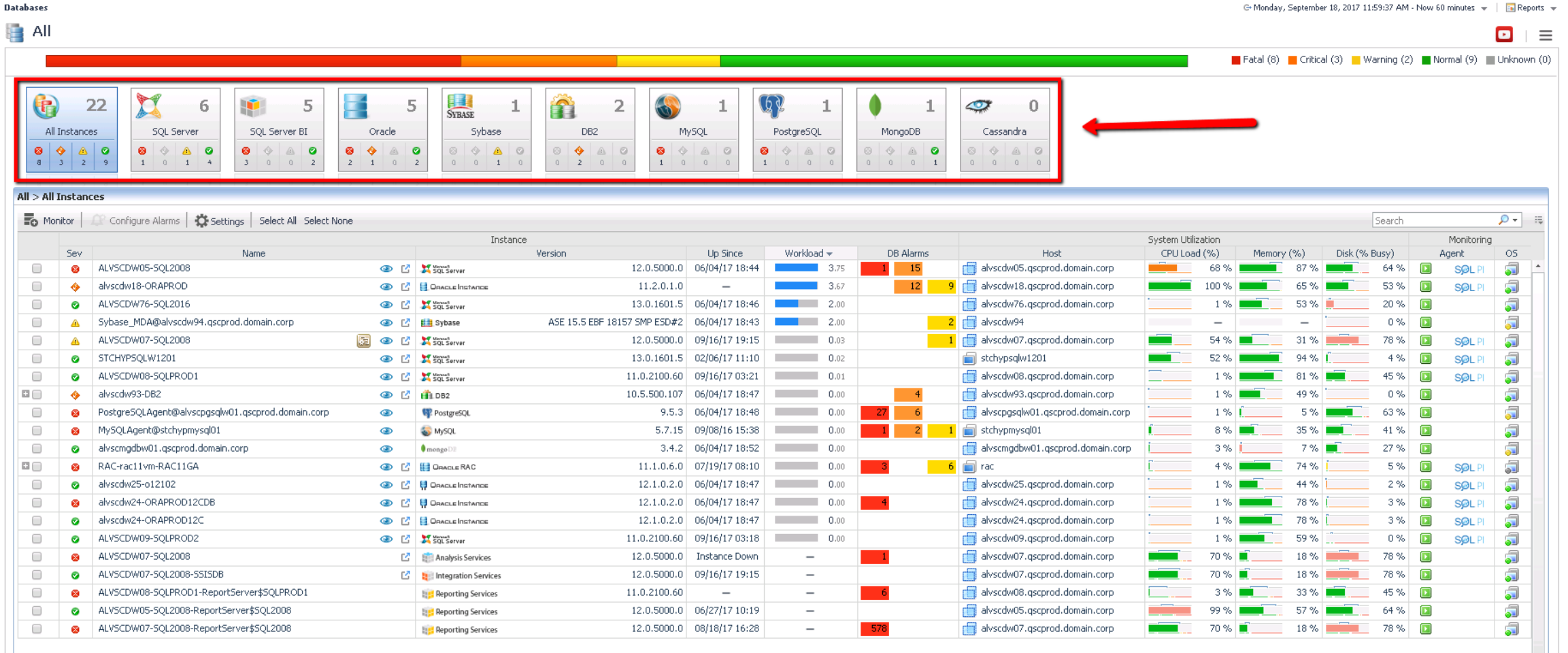
- TABLE ACCESS FULL (Count >= 2) - Excluding DUAL
- 2 or more TABLE ACCESS FULL operations
- TABLE ACCESS FULL (BYTES >= 4 MBytes) - Excluding DUAL
- TABLE ACCESS FULL operation where the Est rows in the step is 4 MBytes or larger

Performance Management – SQL Optimizer

- Simple and automatic re-writing of SQL statements
- Optimized code is returned directly to the Toad Editor



Foglight for Databases: Cross Platform Visibility



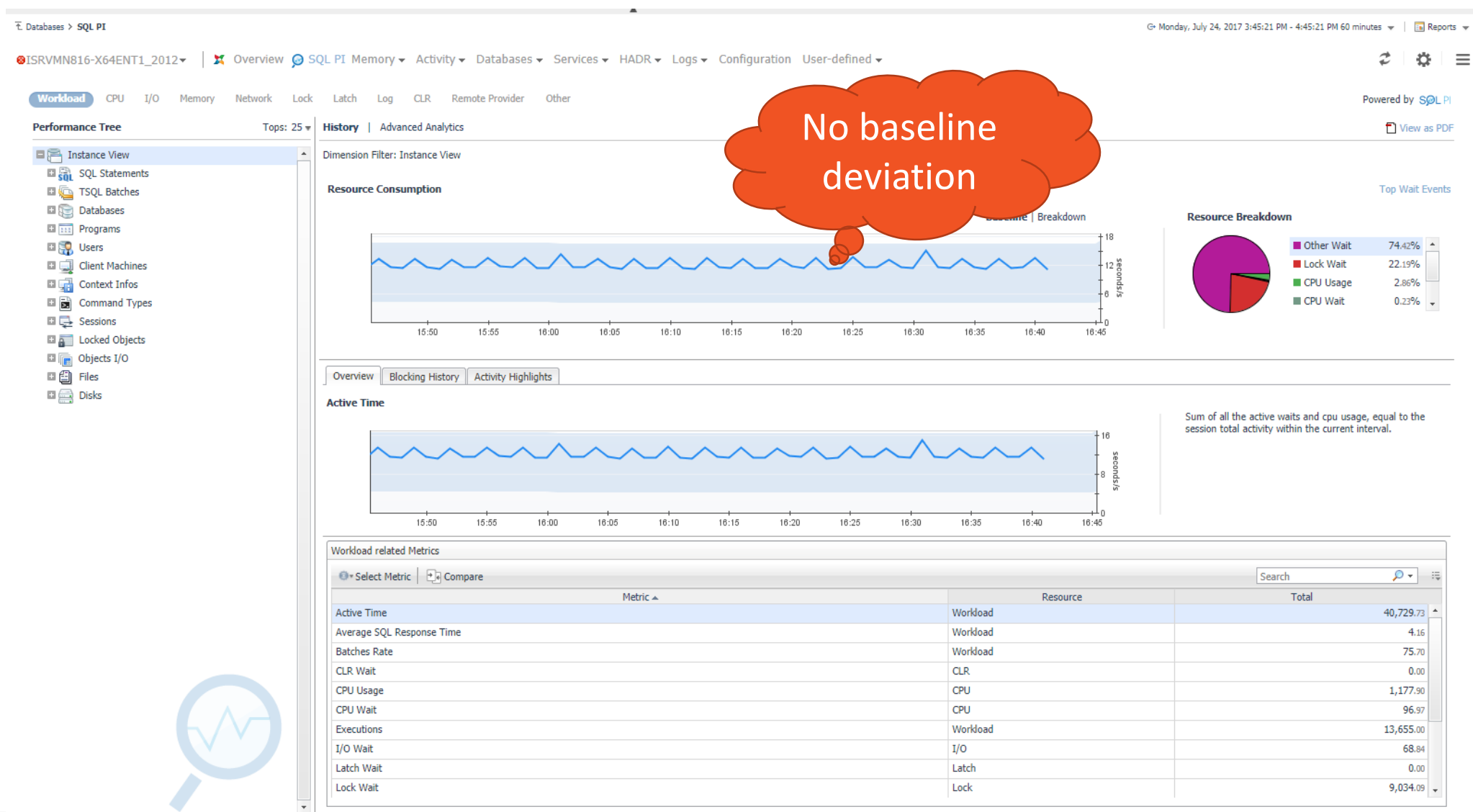
Get Performance Diagnostics Without Spending a Fortune!

- ✓ Foglight for Oracle - the Oracle Diagnostics pack alternative
- ✓ Analysis tool-set for deep visibility into database performance
- ✓ Supports all editions and configurations
- ✓ Remote ("agentless") collector



Supports all database editions and configurations!

Baseline visualization in Foglight



No baseline deviation



Deep-dive Multi-Dimensional workload analysis

The alternative to OEM Active Session History Analytics

The screenshot displays the SQL Performance Center Performance Tree. The left pane shows the 'Instance View' with a comprehensive set of dimensions including SQL Statements, PL/SQL Blocks, Programs, OS Users, DB Users, Machines, Actions, Modules, Client Info, Command Types, Services, Consumer Groups, Databases, Sessions, Locked Objects, Files, Disks, and Objects I/O. The right pane shows a drilldown view of the 'SQL Statements' node, listing specific SQL queries such as 'update orders2 set id = 8327' and several 'SELECT COUNT(CNT) FROM' queries. Two red callout bubbles are overlaid on the image: one on the left pane labeled 'Comprehensive set of dimensions' and one on the right pane labeled 'Drilldown, slice, and dice operations'.

Performance Tree

Tops: 25

Instance View

- SQL Statements
- PL/SQL Blocks
- Programs
- OS Users
- DB Users
- Machines
- Actions
- Modules
- Client Info
- Command Types
- Services
- Consumer Groups
- Databases
- Sessions
- Locked Objects
- Files
- Disks
- Objects I/O

Comprehensive set of dimensions

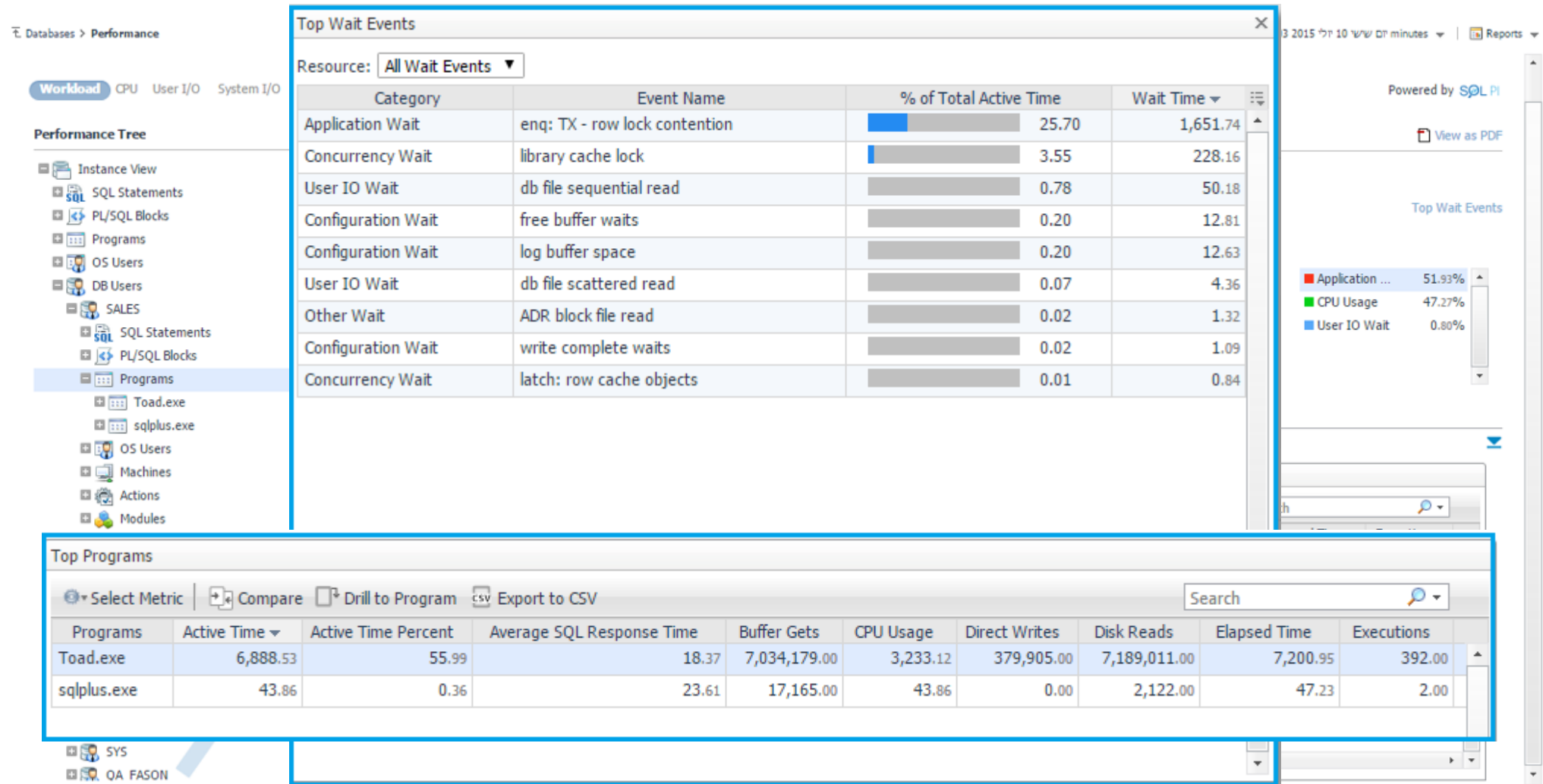
Performance Tree

Tops: 25

- DB Users
- SALES
- SQL Statements
- PL/SQL Blocks
- Programs
- Toad.exe
- SQL Statements
 - update orders2 set id = 8327
 - SELECT COUNT(CNT) FROM (SELECT /*+ full(orders) full(order_line...USTOMER_ID=CUSTOMER.CUSTOMER_ID GROUP BY FIRST_NAME,PRIORITY)
 - SELECT COUNT(CNT) FROM (SELECT /*+ full(orders) full(order_line...IER_ID=PRODUCT.SUPPLIER_ID GROUP BY NAME, ITEM_TYPE,PRIORITY)
 - SELECT COUNT(CNT) FROM (SELECT /*+ full(orders) full(order_line...CITY_CODE=CUSTOMER.CITY GROUP BY CITY , FIRST_NAME, PRIORITY)
 - SELECT COUNT(CNT) FROM (SELECT /*+ full(orders) full(order_line...STOMER_ID < 6000 AND ORDERS.CUSTOMER_ID=CUSTOMER.CUSTOMER_ID)
- PL/SQL Blocks
- OS Users
- Machines

Drilldown, slice, and dice operations

Investigate Multi-Dimensional wait-events & statistics



Track Changes & Correlate them with your workload

Filter by Execution Plan Changes

Correlate between changes to the actual workload

See the SQL statements which have new plans

The screenshot displays the Oracle SQL Performance Insight (SQL PI) interface. The top navigation bar includes 'Databases > SQL PI' and a list of tabs: Overview, SQL PI, Activity, Storage, Configuration, and Alert Log. The 'Workload' tab is selected, showing a 'Performance Tree' on the left with categories like Instance View, SQL Statements, PL/SQL Blocks, Programs, OS Users, DB Users, Machines, Actions, Modules, Client Info, Command Types, Services, Consumer Groups, Sessions, Client PIDs, Locked Objects, Files, Disks, and Objects I/O. The main area shows 'Advanced Analytics' with a 'Resource Consumption' graph and a 'Change Tracking' table. A 'Categories' panel on the right lists various categories with their counts. An 'Analyze Plan' dialog box is open, prompting the user to further investigate the plan using Plan Analysis. The 'Change Tracking' table lists changes, including new plans found on 8/2/17.

Category	Count
Execution Plan	81
Oracle Configuration	60
Oracle Schema	857
System Configuration	6
User Defined	0

Date	Change
8/2/17 5:08 PM	New plan(s) found
8/2/17 3:53 PM	New plan(s) found
8/2/17 2:38 PM	New plan(s) found
8/2/17 2:38 PM	New plan(s) found

Compare between different Execution Plans

Execution Plan Comparison

21:22 09/07/15 Analyze Plan 21:22 09/07/15 Analyze Plan

Operation	Cost	Bytes	Cardinality
Cost Changed 0 SELECT STATEMENT	8,653	10	1
1 SORT AGGREGATE	0	10	1
Cost Changed 2 HASH JOIN	8,653	29,937,670	2,993,767
Operation Changed 3 TABLE ACCESS FULL SALES.DEPT	1,189	4,999,755	999,951
Operation Changed 4 TABLE ACCESS FULL SALES.EMP	4,152	14,999,270	2,999,854

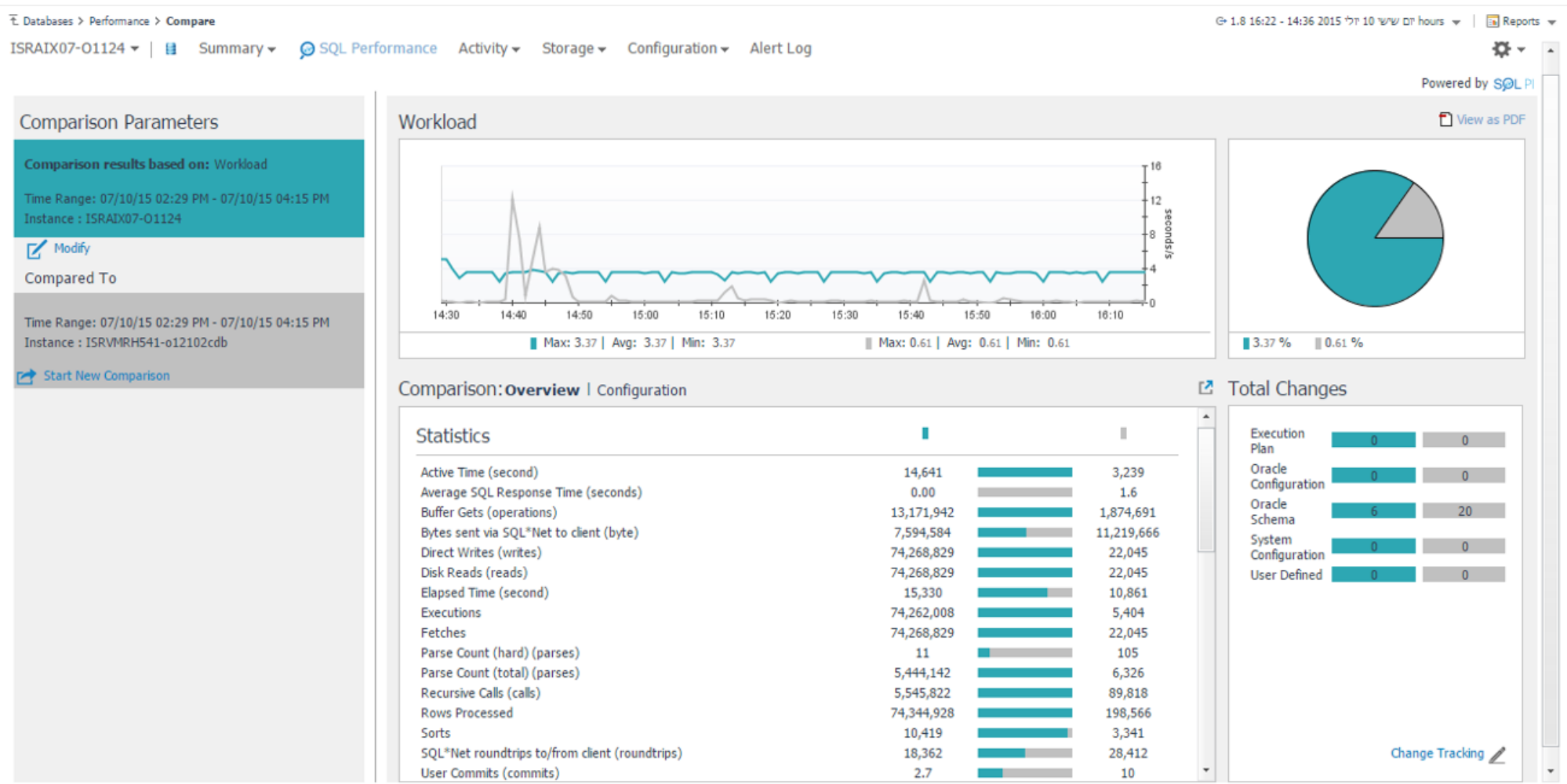
Operation	Cost	Bytes	Cardinality
Cost Changed 0 SELECT STATEMENT	2,909	10	1
1 SORT AGGREGATE	0	10	1
Cost Changed 2 HASH JOIN	2,909	9,999,510	999,951
Operation Changed 3 INDEX FAST FULL SCAN SYS.EMP_IDX	628	4,999,755	999,951
Operation Changed 4 INDEX FAST FULL SCAN SYS.DEPT_IDX	628	4,999,755	999,951

SQL Text

```
SELECT count(*)
FROM emp JOIN DEPT ON (emp.dept_id = dept.id)
WHERE dept.id > 50
```

Close

Compare different Instances or Dimensions



Compare different Instances or Dimensions

Databases > Performance > Compare

ISRAIX07-O1124 | Summary | **SQL Performance** | Activity | Storage | Configuration | Alert Log

PowerTools SQL PI

Comparison Parameters

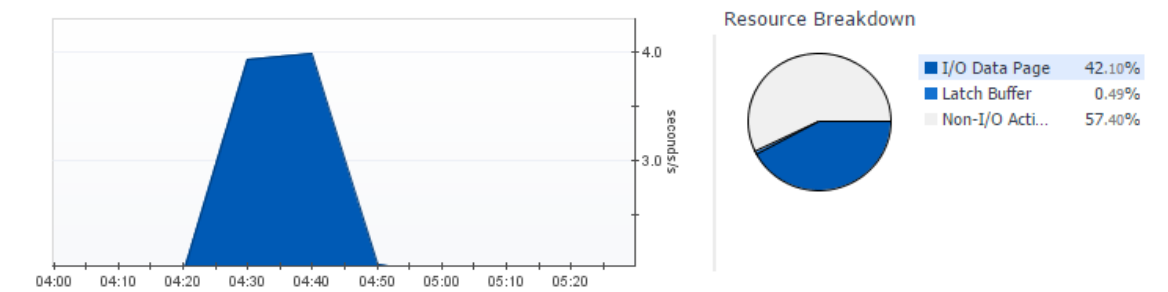
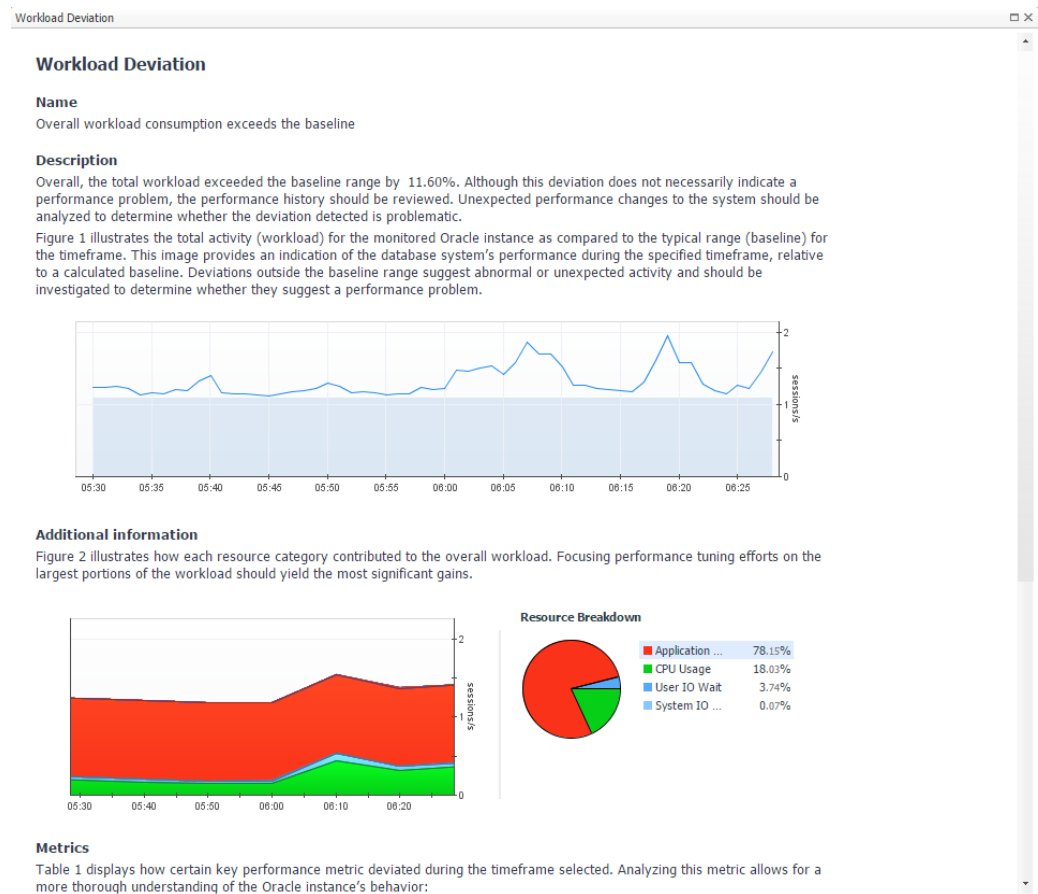
Comparison: Overview | **Configuration**

View: * | # | [icon]

Configuration

audit_file_dest	/oravl01/oracle/admin/O1124/ad...	≠	/oravl01/oracle/admin/o12102cd...
audit_sys_operations	FALSE	≠	TRUE
background_dump_dest	/oravl01/oracle/diag/rdbms/o11...	≠	/oravl01/oracle/product/12.1.0...
backup_tape_io_slaves	FALSE	≠	TRUE
blank_trimming	FALSE	≠	TRUE
compatible	11.2.0.4.0	≠	12.1.0.2.0
control_files	/oravl01/oracle/oradata/O1124/...	≠	/oravl01/oracle/oradata/O12102...
core_dump_dest	/oravl01/oracle/diag/rdbms/o11...	≠	/oravl01/oracle/diag/rdbms/o12...
cpu_count	8	≠	4
db_cache_size	0	≠	234881024
db_file_multiblock_read_count	30	≠	34
db_name	O1124	≠	o12102cd
db_securefile	PERMITTED	≠	PREFERRED
db_unique_name	O1124	≠	o12102cdb
dg_broker_config_file1	/oravl01/oracle/product/11.2.0...	≠	/oravl01/oracle/product/12.1.0...
dg_broker_config_file2	/oravl01/oracle/product/11.2.0...	≠	/oravl01/oracle/product/12.1.0...
dispatchers	(PROTOCOL=TCP) (SERVICE=O1124X...	≠	(PROTOCOL=TCP) (SERVICE=o12102...
dml_locks	4080	≠	5384

Advisories



Use the following information to determine the root cause of the I/O contention observed during the analysis period.

Table 1: Top SQL Statements producing I/O Contention

	SQL Statements	Number of Executions	Average Active Time (seconds)	% of Active Time	I/O Wait (%)	I/O Data Page Wait (%)
	insert into #Quest_fragmented_indexes select si . name as IndexN... = @1 and si . type < > @2 order by AverageFragmentation desc	-	0.00	39.87	81.02	80.36
	insert into #Dell_fragmented_indexes select si . name as IndexNa... = @1 and si . type < > @2 order by AverageFragmentation desc	-	0.00	12.86	18.81	18.41

Click on a SQL Statement to find out more about the operation of the specific statement.
Press "Analyze Plan" to see information about the execution plan of the statement.
Press "Tune" to launch SQL Tuning for the statement, maintaining database context (if it is installed).

Table2: Top missing indexes

Each SQL Server maintains a list of the indexes considered to be missing from each database. The list is updated whenever a query is executed. The table below lists indexes identified in the analyzed timeframe, which are ranked based on the relative improvement they would yield if they existed. The relative impact of each index is based on <average cost of the query(s)> * <average cost-improvement of the query(s)> * <Number of seeks+scans that could have utilized the index>.

	Database	Schema	Object name	Relative Impact	Avg User Impact (%)	Avg Cost	User Seeks	User Scans	Noticed Missing	Statement
	SpotlightPlaybackDatabase_batchen	dbo	spotlight_playback_alarms	88,268.17	15.04	167.68	35.00	0.00	2/9/17 2:20 AM	CREATE INDEX missi...
	SpotlightPlaybackDatabase_batchen	dbo	spotlight_playback_alarms	87,798.66	14.96	167.68	35.00	0.00	2/9/17 2:20 AM	CREATE INDEX missi...

Click on the statement to get the full command for creating the index.
Note: Adding indexes adds overhead of maintaining that index. Before creating an index, ensure the overhead is negligible compared to the improvement. You can use the Database Tuning Advisor (DTA) for that.

What our customers say?

QUEST FOGLIGHT CUSTOMER TESTIMONIAL

Foglight has helped me in my role as an IT professional by...

“Using Foglight with Oracle Standard Edition to decrease OEM license costs and still provide vision into system health and upcoming problems.”

— Sezgin Cebi, Senior IT Deputy Manager, Kaptan Demir Celik

Source: Sezgin Cebi, Senior IT Deputy Manager, Kaptan Demir Celik

Validated Published: Aug. 3, 2017 TVID: 4C6-F55-E7C

Quest



QUEST FOGLIGHT CUSTOMER TESTIMONIAL

Foglight has helped me in my role as an IT professional by...

“Providing me with a centralized view of my entire distributed database environment.”

— Database Administrator, Medium Enterprise Environmental Services & Equipment Company

Source: Database Administrator, Medium Enterprise Environmental Services & Equipment Company

Validated Published: Aug. 3, 2017 TVID: 8F2-EA8-03A

Quest



QUEST FOGLIGHT CUSTOMER TESTIMONIAL

Foglight has helped me in my role as an IT professional by...

“Allowing me to stand out as a performance tuning specialist through understanding our performance metrics.”

— IT Administrator, Large Enterprise Transportation Services Company

Source: IT Administrator, Large Enterprise Transportation Services Company

Validated Published: Aug. 3, 2017 TVID: E9F-682-8D5

Quest



Additional Resources

Foglight for Oracle Website

<https://www.quest.com/products/foglight-for-oracle/>

Foglight for Oracle Product Overview Video

<https://youtu.be/J9oIUZx3YaQ>

Questions?



QuestTM

Thank you

QuestTM