

accenture operations

Accenture Enkitech Group

Smooth 12c Upgrades

High performance. Delivered.

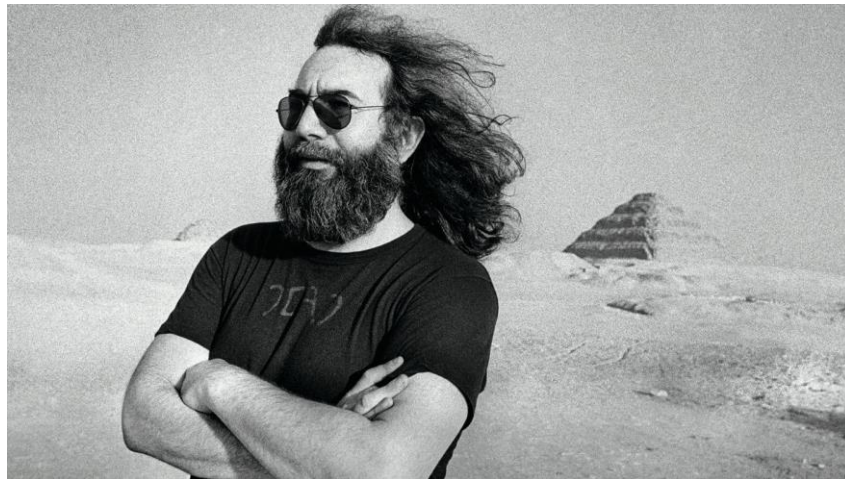
Andy Klock 2016

accenture operations

Strategy | Consulting | Digital | Technology | Operations

Smooth 12c Upgrades

Stuff that's hidden and murky and ambiguous is scary because you don't know what it does. -- Jerry Garcia



High performance. Delivered.

accenture operations

Andy Klock

Infrastructure Senior Principal / Oracle Guy

ORACLE[®]
Certified Professional

- Majored in Geology so I'm used to thinking in big numbers
- Got my first Oracle job in 2000 at Raytheon working for FEMA
- Was an Oracle Apps guy for 2 years in Boston
- Wrote financial software at Portware
- Technical Team Lead at Pythian
- Joined Enkitech (now AEG) in 2013
- Like all things Oracle
- Oracle OCP 8, 8i, 9i, 10g so, I guess I'm really behind the times



High performance. Delivered.

- Email: andy.klock@accenture.com
- Blog: checkyourlogsblog.com
- Twitter: [@andyklock](https://twitter.com/andyklock)

Strategy | Consulting | Digital | Technology | **Operations**

accenture>operations

Accenture Enkitech Group

- Kerry Osborne is still around
- Still doing lots of Exadata
- Still doing lots of Health Checks
- Still own a bunch Exadata's, BDAs, Exalytics's, ODAs, etc.
- Still a great place to work

High performance. Delivered.



Strategy | Consulting | Digital | Technology | [Operations](#)

Recent Upgrade / Migration

- Large North American Postal Operator Tracking System
- 2x 75+ TB Databases
- 11.2.0.3 -> 12.1.0.2 Apr PSU
- Migration to HP SuperDome X
- 144 cores, 6 TB RAM
- 2x Active Data Guard
- GoldenGate
- Custom PL/SQL Application
- Plus Informatica / Java / Hibernate



High performance. Delivered.

12c Upgrade Paths

- Upgrade or Migration?
- Multitenant?
- DBUA or Command-Line?
- Full Transportable Export/Import
- Transportable Tablespaces
- Rolling Upgrades?
- Data Guard?
- GoldenGate?
- WHERE DO WE GO?



12c Features

- Adaptive Query Optimizations
- Adaptive Plans
- Adaptive Statistics
- Automatic Re-optimization
- SQL Plan Directives
- Enhanced Statistics
- Sound good?
- Not yet.
- `optimizer_adaptive_features = FALSE !`



accenture operations

Performance Stability

- What can you do to prepare?
- Test
- Real Application Testing
- SQL Performance Analyzer
- Custom Testing
- ASH/AWR Reviews
- Are you ready?
- How can you be sure?



SQL Plan Management

- What are they?
- Canned Execution plans
- Sort of like old school Outlines
- Use hints to derive Plans
- In 12c the plans are actually stored in SPM



accenture[>]operations

SPM Options

- Option 1: Capture everything!
 - Just don't "Accept" the SQL Plan Baselines
- Option 2: Capture everything!
 - "Accept" but plan on spending the rest of your life evolving baselines
- Option 1 Wins



High performance. Delivered.

Option 1: Capture Everything!

- Problems with Option 1
- Thousands and thousands of baselines
- 11g Baselines aren't guaranteed to work
- 11g has lots of SPM bugs
- Plans aren't stored in 11g SPM
- Whelp.
- Now what?



Option 3! SQL Tuning Sets

- A set of SQL Statements
- Execution context
 - Objects
 - Module/Action
 - Bind variables
- Execution statistics
 - Elapsed time
 - CPU, buffer gets, executions, cost, etc.
- Executions Plans!



accenture operations

SQL Tuning Sets (STS)

- So what are they used for?
- SQL Performance Analyzer
- SQL Tuning Advisors
- SQL Access Advisors
- SQLT
- Doctors
- SQL Plan Baselines can be created from them!



accenture[>]operations

The Plan

- Capture PROD statements into STS
- Move STS to a TEST database
- Upgrade TEST
 - AWR is your friend
 - Test, test, test
- Identify Regressed SQL
- Create SQL Baselines if needed
- Rinse/Repeat



accenture[>]operations

Load PROD Statements into a STS

- Determine which statements are needed

- AWR
- Cursor Cache

- Parsing Schema Name

- Executions

- Scripts

- <https://github.com/andyklock/scripts/tree/master/sts>
- Majority are based off of Carlos Sierra's SPM scripts
- <https://github.com/carlos-sierra/cscripts/archive/master.zip>



High performance. Delivered.

accenture[>]operations

SQL Tuning Set Recipe

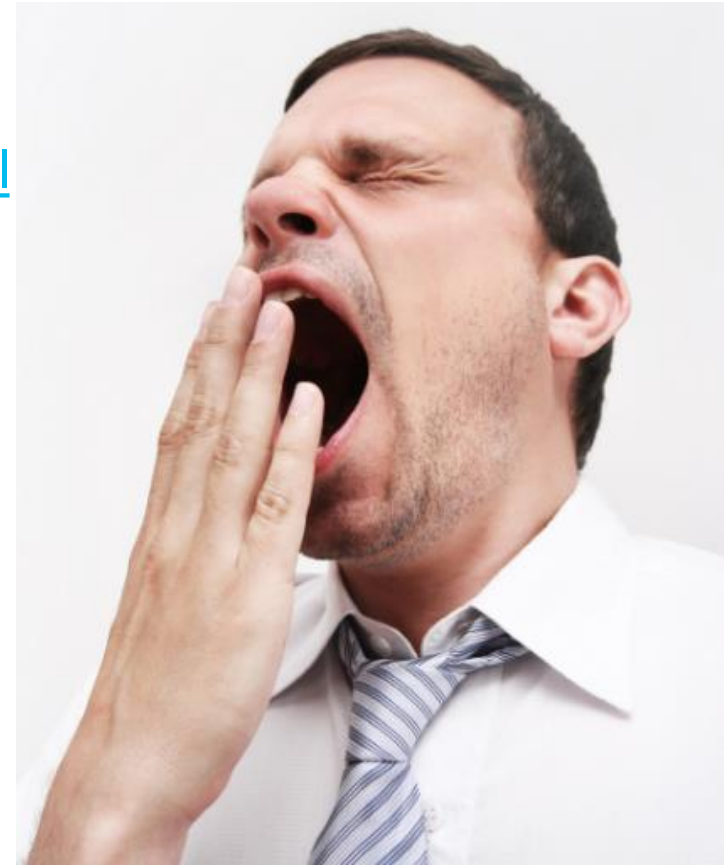
- Load statements from Cursor Cache
- Load statements from AWR
- Pack and export STS
- Import and unpack STS
- Create SQL Plan Baseline from STS



accenture[>]operations

SQL Tuning Set Recipe Scripts

- [create_sql_tuning_set_from_cursor_cache.sql](#)
- [create_sql_tuning_set_from_awr.sql](#)
- [pack_sql_tuning_sets.sql](#)
- [unpack_sql_tuning_sets.sql](#)
- [create_spb_from_sts.sql](#)
- [params.sql](#) <<< special configuration script
 - SQLSET_OWNER
 - SQLSET_TAG
 - PARSING_SCHEMAS



Load PROD Statements into a STS (cont.)

```
-- Totals for everything, by APP SCHEMA (renamed schemas to protect the innocent)
SQL> select parsing_schema_name, count(distinct sql_id) from gv$sql
      where parsing_schema_name in ('APP1', 'APP2', 'APP3', 'APP4') and
      last_active_time > sysdate-5
      group by parsing_schema_name order by 2,1;
```

PARSING_SCHEMA_NAME	COUNT(DISTINCTSQL_ID)
APP1	105
APP2	2715
APP3	79425
APP4	226309

```
-- Totals for everything executed over 50 times
SQL> select parsing_schema_name, count(distinct sql_id) from gv$sql
      where parsing_schema_name in ('APP1', 'APP2', 'APP3', 'APP4') and
      executions > 50 group by parsing_schema_name;
```

PARSING_SCHEMA_NAME	COUNT(DISTINCTSQL_ID)
APP1	14
APP2	33
APP3	207
APP4	2063

accenture operations

Load PROD Statements into a STS (cont.)

- From create_sql_tuning_set_from_cursor_cache.sql

```
OPEN sts_cur FOR
  SELECT VALUE(p)
    FROM TABLE(DBMS_SQLTUNE.select_cursor_cache(
      'parsing_schema_name in ('||l_parsing_schema_name_list||') and executions > 50',
      NULL, NULL, NULL, NULL, 1, NULL, 'ALL')) p;

SYS.DBMS_SQLTUNE.load_sqlset (
  sqlset_name      => l_sqlset_name,
  populate_cursor => sts_cur,
  sqlset_owner => '&sqlset_owner' );
DBMS_OUTPUT.put_line('loaded sqlset: '||l_sqlset_name);

CLOSE sts_cur;
```

accenture operations

Load PROD Statements into a STS (cont.)

- From create_sql_tuning_set_from_awr.sql

```
OPEN sts_cur FOR
  SELECT VALUE(p)
    FROM TABLE(DBMS_SQLTUNE.select_workload_repository(l_snap_begin, l_snap_end,
      'parsing_schema_name in ('||l_parsing_schema_name_list||') AND loaded_versions > 0',
      NULL, NULL, NULL, NULL, 1, NULL, 'ALL')) p;

SYS.DBMS_SQLTUNE.load_sqlset (
  sqlset_name      => l_sqlset_name,
  sqlset_owner     => '&&sqlset_owner',
  populate_cursor => sts_cur );
DBMS_OUTPUT.put_line('loaded sqlset: '||l_sqlset_name);

CLOSE sts_cur;
```

Load PROD Statements into a STS (cont.)

```
SQL> select SQLSET_NAME, SQLSET_OWNER, count(*) from dba_sqlset_statements  
group by SQLSET_NAME, SQLSET_OWNER order by 2,1;
```

SQLSET_NAME	SQLSET_OWNER	COUNT (*)
STS_11G_AWR	STS_OWNER	1430
STS_11G_CURCACHE_1	STS_OWNER	2522
STS_11G_CURCACHE_2	STS_OWNER	2457
STS_11G_CURCACHE_3	STS_OWNER	2344
STS_11G_CURCACHE_4	STS_OWNER	2025
STS_11G_CURCACHE_5	STS_OWNER	2020
STS_11G_CURCACHE_6	STS_OWNER	1772

accenture[>]operations

Moving to TEST

- PROD
 - [pack_sql_tuning_sets.sql](#)
 - Export/expdp
- TEST
 - Import/impdp
 - [unpack_sql_tuning_sets.sql](#)



accenture[>]operations

Test!

- Now what?
- Find regressed SQL
- How?
- AWR Top SQL
- [ashtop.sql](#) (Tanel Poder's) < best tool ever*
- [sql_performance_changed.sql](#) (Carlos Sierra's)
- [sqlset regress cur.sql](#)
- [unstable_plans.sql](#) (Kerry Osborne's)
- [awr_plan_change_all.sql](#)

* besides Snapper



Regressed SQL

```
SQL> @sqlset_regress_cur.sql
```

SQL_ID	V12C_PHV	EXECUTIONS	AVG_ETIME	V11G_PHV	EXECUTIONS	AVG_ETIME	D	DIFF_ETIME	PERFORMAN
6zjxqxdb9t6vd	890145704	3	318.897	2409362170	634767	1.630	*	-317.26637	Regressed
376fjswqu4cnt	1920954746	3	140.638	2259879957	3218	.328	*	-140.31039	Regressed
0wf3hdbcmfn9d	1566315087	3	161.397	1566315087	174	65.093		-96.304033	Regressed
f61rxd361b8g2	1566315087	3	113.604	1566315087	163	69.922		-43.682398	Regressed
gp1p36b8yskbv	3302961321	7	27.064	1218582728	8981	.682	*	-26.381721	Regressed
b2kxd8xwd3ymy	659280062	1	26.402	659280062	101	7.523		-18.879445	Regressed
ghfdrs7hrw252	2155093733	33	23.465	2603635391	85319	5.554	*	-17.911551	Regressed
0znx18d043bg3	2743575134	1	15.805	3971049314	107	.004	*	-15.801179	Regressed
d29prd53mzuyc	2371823249	8	14.128	3772892946	17002	.084	*	-14.044119	Regressed
a6yadgjjwulkm1	4258360765	17	9.351	1959399513	81643	.015	*	-9.3363534	Regressed

Regressed SQL

```
SQL> @awr_plan_change_all 2
Enter value for sql_id: f3dmvyhb80a3c
```

SNAP_ID	NODE	BEGIN_INTERVAL_TIME	SQL_ID	PLAN_HASH_VALUE	EXECS	AVG_ETIME	AVG_CPU_TIME
37365	1	10-AUG-16 10.00.42.154 AM	f3dmvyhb80a3c	3596954981	558	6.360	5.989
37366	1	10-AUG-16 11.00.02.498 AM	f3dmvyhb80a3c		597	6.038	5.738
37367	1	10-AUG-16 12.00.15.310 PM	f3dmvyhb80a3c		601	5.962	5.713
37368	1	10-AUG-16 01.00.03.469 PM	f3dmvyhb80a3c		596	6.030	5.776
37369	1	10-AUG-16 02.00.09.605 PM	f3dmvyhb80a3c		577	6.240	5.954
37370	1	10-AUG-16 03.00.17.792 PM	f3dmvyhb80a3c		577	6.209	5.921
37371	1	10-AUG-16 04.00.08.722 PM	f3dmvyhb80a3c		578	6.238	5.958
37372	1	10-AUG-16 05.00.20.405 PM	f3dmvyhb80a3c		568	6.348	6.062
37373	1	10-AUG-16 06.00.32.517 PM	f3dmvyhb80a3c		543	6.629	6.331
37379	1	11-AUG-16 01.00.01.148 AM	f3dmvyhb80a3c	2260335173	1	2,215.397	2,131.329
37379	1	11-AUG-16 01.00.01.148 AM	f3dmvyhb80a3c	3249808128	1	1,387.308	1,338.448
37383	1	11-AUG-16 05.00.04.139 AM	f3dmvyhb80a3c	2260335173	1	3,620.944	3,510.651

Creating Baselines

- Find the plan in a STS
- Create a baseline
- Verify if it's being used



Find Statement in STS

```
SQL> @find_sts_sql
```

```
Enter SQL_ID: f3dmvyhb80a3c
```

SQLSET_NAME	SQLSET_OWNER	PLAN_HASH_VALUE
-----	-----	-----
STS_11G_AWR	STS_OWNER	3596954981
STS_11G_CURCACHE_1	STS_OWNER	3596954981
STS_11G_CURCACHE_2	STS_OWNER	3596954981



Create SQL Plan Baseline

```
SQL> @create_spb_from_sts.sql
```

```
Enter SQL_ID: f3dmvyhb80a3c
```

```
Enter SQL Set Name: STS_11G_CURCACHE_1
```

```
Enter SQL Set Owner: STS_OWNER
```

```
Enter optional Plan Hash Value: 3596954981
```

```
SQL> @baselines
```

SQL_HANDLE	SIGNATURE	PLAN_NAME	LAST_EXECUTED	ACC	FIX	ENA
...						
SQL_c8415cc67a8e1de7	14429916688601193959	SQL_PLAN_chhawstx8w7g7c24dff08	11-AUG-16 09.34.43.000000 AM	YES	YES	YES
...						

Flush, but verify

```
SQL> @flush_cursor
Enter value for sql_id: f3dmvyhb80a3c

'EXECDBMS_SHARED_POOL.PURGE(''||ADDRESS||','||HASH_VALUE||''','C');--RUNFROMINSTANCE'||INST_ID
-----
exec DBMS_SHARED_POOL.PURGE ('0000013F89F95BB8, 377497708', 'C'); -- run from instance1

SQL> @sq f3dmvyhb80a3c
```

INST_ID	CHILD_NUMBER	EXECUTIONS	PLAN_HASH_VALUE	SQL_PLAN_BASELINE	LAST_ACTIVE_TIME	AVG_ELAPSED
1	0	24	3596954981	SQL_PLAN_chhawstx8w7g7c24dff08	20160811 09:43:56	6.39465699

- Same rules apply as in previous releases. Statements have to be completed or killed before they can be flushed

accenture[>]operations

All good?

- 11g CBO bugs sometimes are fixed in 12c
- Sometimes the CBO just can't reproduce
- Check `dbms_xplan.display_sql_plan_baseline`
- SQLD360
- 10053!
 - Grep for SPM
- Watch out for SYS calls
 - SPM: disallowed: SQL with a bootstrap object run by SYS (Bug14029891)



High performance. Delivered.

Notes, Bugs, Patches...

- SPM Master Note: (**Doc ID 1359841.1**)
- Things to Consider When Upgrading From 12.1.0.1 to Avoid Problems with (SPM) (**Doc ID 2035897.1**)

[Document 1948958.1](#) Patches to Consider for 11.2.0.3 to Avoid Problems with (SPM)

[Document 2034706.1](#) Patches to Consider for 11.2.0.4 to Avoid Problems with (SPM)

[Document 2035898.1](#) Patches to Consider for 12.1.0.2 to Avoid Problems with (SPM)



accenture operations

Licensing...

- SPM is included in EE
- STS is extra \$\$\$
- What's what?
 - DBMS_SPM (good)
 - DBMS_SQLTUNE (\$\$\$)
- Alternatives?



accenture[>]operations

All good! But SPM/STS isn't just for upgrades...

- Upgrades
- Application Releases
- System changes
- Anything that could effect performance
- STS don't take up all that much space
 - What's a few GB among friends?
- STS can be backed up and dropped
- But,...



High performance. Delivered.

accenture[>]operations

Disclaimer!

- SQL Plan Baselines are a band-aid
 - (the way we are using them)
- They don't fix the real problem
 - Stats? Bugs? Most likely stats...
- Just the symptom
- Once performance is stable, revisit and properly fix



High performance. Delivered.



References

- Links, blogs, white papers

https://blogs.oracle.com/optimizer/entry/upgrade_to_oracle_database_12c1

http://kerryosborne.oracle-guy.com/papers/SPM_12c.pdf

<https://carlos-sierra.net/>

<https://mauro-pagano.com/>

<http://www.oracle.com/technetwork/database/bi-datawarehousing/twp-sql-plan-mgmt-12c-1963237.pdf>

<http://www.oracle.com/technetwork/database/bi-datawarehousing/twp-optimizer-with-oracledb-12c-1963236.pdf>

Questions?

- Andy Klock Contact Details
 - Email: andy.klock@accenture.com
 - Blog: checkyourlogsblog.com
 - Twitter: [@andyklock](https://twitter.com/andyklock)
- NYC Oracle Hackers Meetup
 - 9/20 (3rd Tuesdays of each month) at Manhattan d.b.a bar



High performance. Delivered.