Tuning with AWR

Adi Kalfon
System Support Consultant
Oracle Advanced Customer Services

Agenda

- What is AWR ?
- How to go through all the information?
- How can we maximize the use of it?
- Useful tools to complete the whole picture

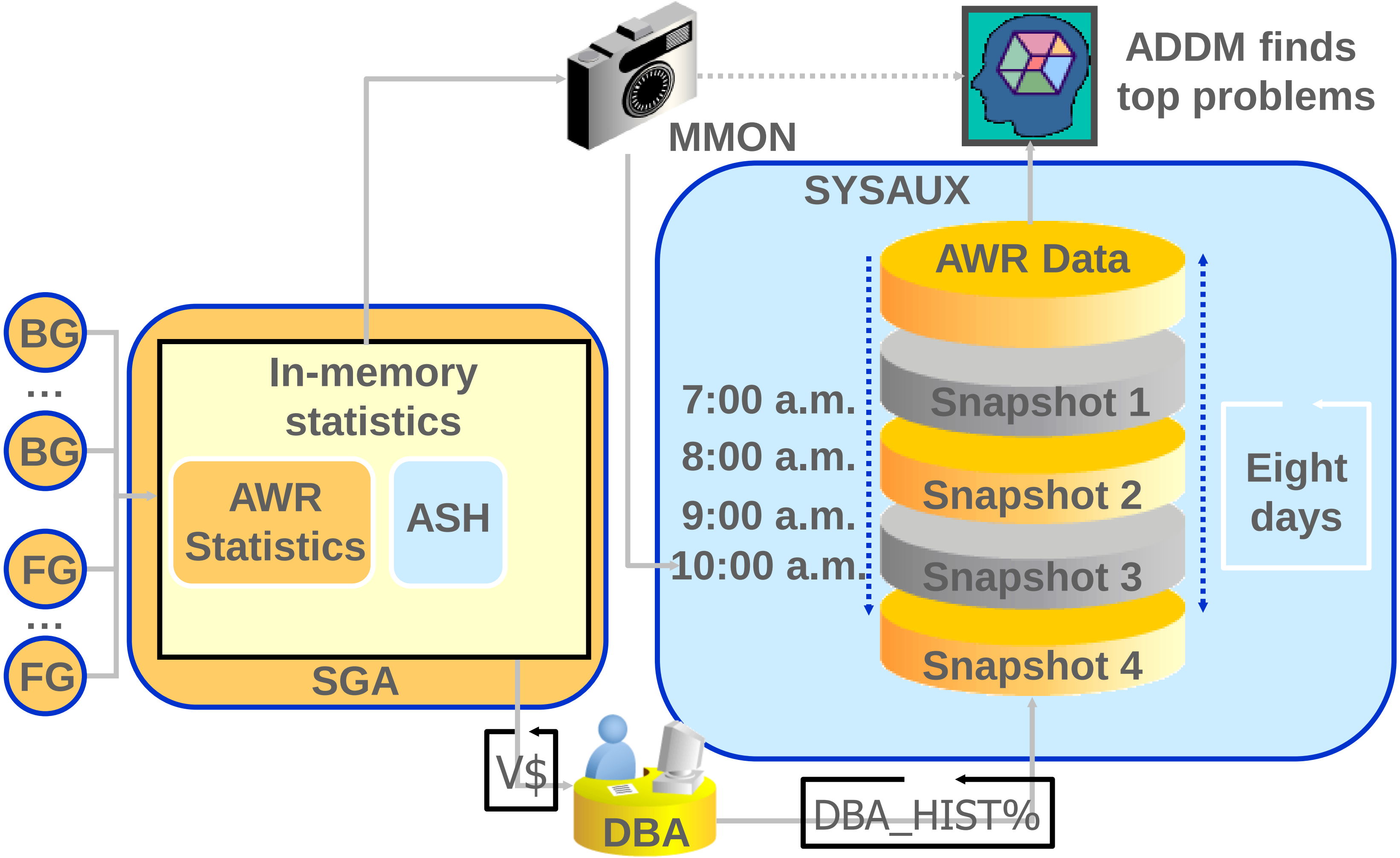
What is AWR?

Automatic Workload Repository

- The most well known performance report.
- Available since Oracle 10g.
- Next generation of statspack report.
- Based on automatic snapshots taken every 1 hour and saved in SYSAUX tablespace for 8 days by default.
- Based on v$ and dba_hist views.

What is AWR?

Automatic Workload Repository



Before you start

- **Ask the right questions:**
  - What does the application do?
  - What is the problem?
  - What is the system workload?
  - What changed?
  - What is the goal?
- **Has something changed?**
  - Obtain information pre- and post- change
- **Cross information with OS statistics**
  - Finding a host bottleneck doesn’t fix the problem
  - It does give you some clues

Before you start

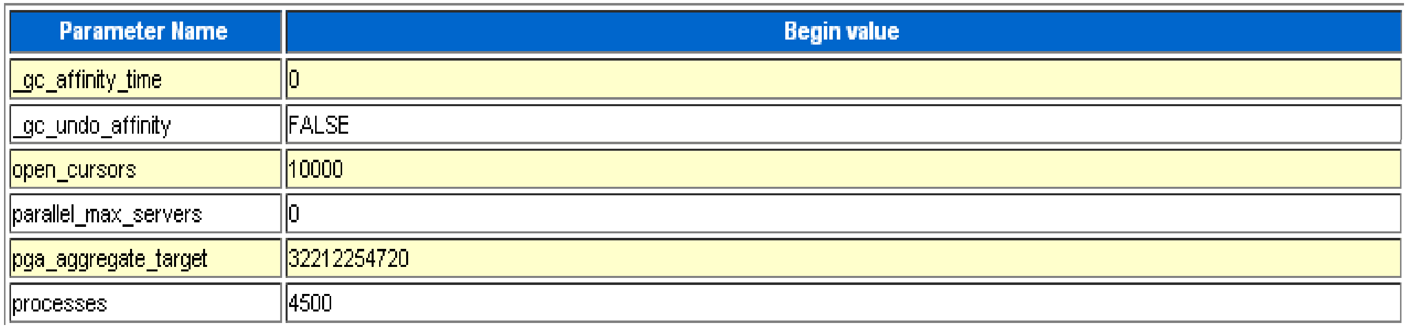
Change	Likely Impact
Oracle Parameters	✓
System Environment	✓
Object Configuration	✓
Application	✓

Interpreting an AWR Report

A Little Background Reading

- Is the software current?
- Are any unusual parameters in use?
- Is RAC in use?
  - Many problems are the same as non-RAC
  - But you may need to know more about the workload

DB Name	DB Id	Instance	Inst num	Release	RAC	Host
ACME	1214027630	acme	1	**10.2.0.2.0**	NO	y507

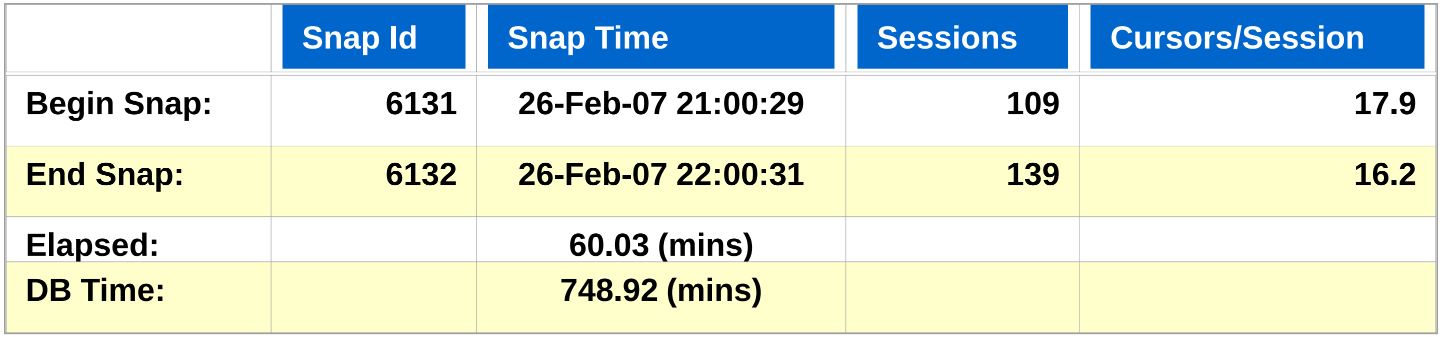


Parameter Name	Begin value
_gc_affinity_time	0
_gc_undo_affinity	FALSE
open_cursors	10000
parallel_max_servers	0
pga_aggregate_target	32212254720
processes	4500

Interpreting an AWR Report

Is the database the problem?

- Is the database busy doing something?



	Snap Id	Snap Time	Sessions	Cursors/Session
Begin Snap:	6131	26-Feb-07 21:00:29	109	17.9
End Snap:	6132	26-Feb-07 22:00:31	139	16.2
Elapsed:		60.03 (mins)		
DB Time:		748.92 (mins)		

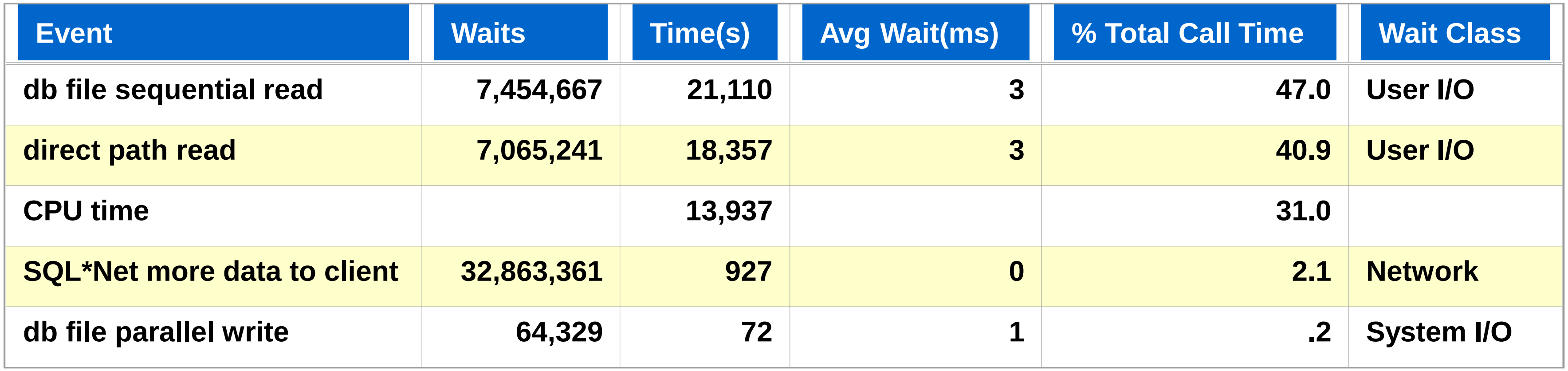
- **DB Time**
  - Total time in database calls by foreground sessions
  - Includes CPU time, IO time and non-idle wait time
  - Total DB time = sum of DB time for all active sessions

➢ **The Goal: To Reduce Total DB time**

Interpreting an AWR Report

What are we waiting for?

Top 5 Timed Events



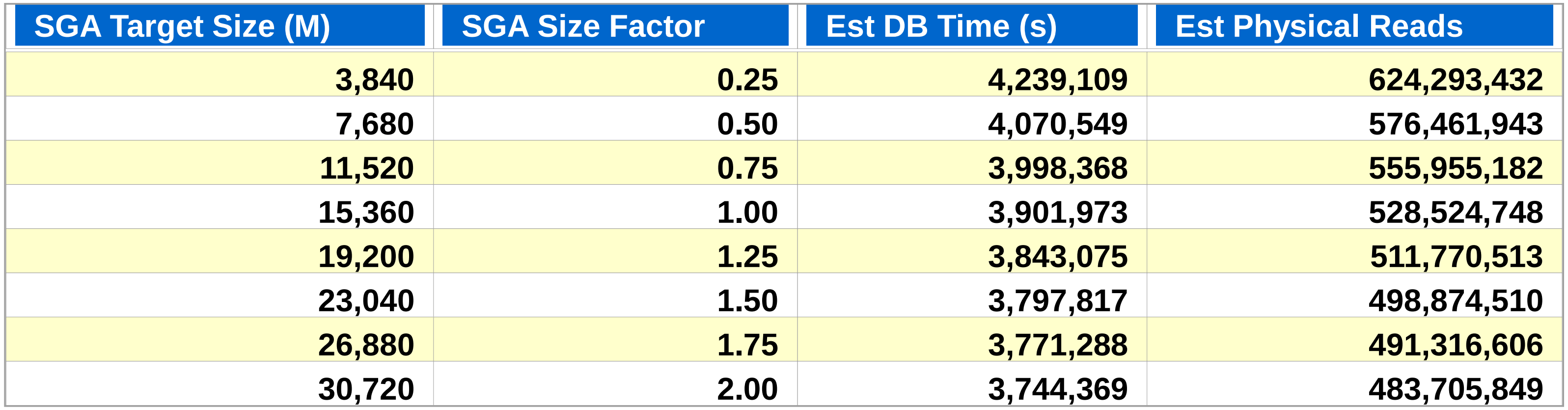
Event	Waits	Time(s)	Avg Wait(ms)	% Total Call Time	Wait Class
db file sequential read	7,454,667	21,110	3	47.0	User I/O
direct path read	7,065,241	18,357	3	40.9	User I/O
CPU time		13,937		31.0	
SQL*Net more data to client	32,863,361	927	0	2.1	Network
db file parallel write	64,329	72	1	.2	System I/O

Interpreting an AWR Report

So what is the problem?

- Actually there isn't one!
  - The customer is happy with performance
- Could the application run faster? Maybe.
  - Increase the buffer cache? Need to check
  - Optimize the storage? Need to check
  - Tune the application? Most Probable

SGA Target Advisory



SGA Target Size (M)	SGA Size Factor	Est DB Time (s)	Est Physical Reads
3,840	0.25	4,239,109	624,293,432
7,680	0.50	4,070,549	576,461,943
11,520	0.75	3,998,368	555,955,182
15,360	1.00	3,901,973	528,524,748
19,200	1.25	3,843,075	511,770,513
23,040	1.50	3,797,817	498,874,510
26,880	1.75	3,771,288	491,316,606
30,720	2.00	3,744,369	483,705,849

IO Issues

- Application usage: Table scans / index usage, parallel executions
- Background processes workload: DBWR, LGWR, ARCH, CKPT
- Files workload → oracle level (extents, hot spots, fragmentation)
  → OS level (read/write per second, striping etc)

Relevant issues:

db file scattered read

db file sequential read

direct path read/write

parallel statistics

Cache sizes

log file/archive statistics

other files statistics: control/system/data/temp/undo

Application Issues

- Commit and log file sync
- Locks and enqueues
- Parses
- Logons
- Hot spots

Relevant issues:

Library cache*

cursor : mutex x

Latches: buffer cache/shared pool/session allocation

Enqueues (TX,TM,UL etc)

read by other session

Log file sync

Memory Issues

- Configuration issues: cache size, shared pool etc.
- Internal management: latches
- Fragmentation

Relevant issues:

Buffer busy waits

Free buffer waits

Log file sync

Shared pool*

RAC Issues

- Cache Fusion
- Interconnect and IO
- Configuration: services, parallel

Relevant issues:

GC cr *

GC current *

Interconnect traffic

IO access

Memory latches

Application enqueues

Methodology

➔How to check yourself?

- Use Data Dictionary views
- Use ADDM (Automatic Database Diagnostics Monitor) to:
  - Quantify problems
  - Quantify recommendations
  - Identify the root cause
- Use ASH (Active Session History)
- Use O/S statistics (iostat, oswatcher)
- AWR related scripts

Use Data Dictionary views

You suspect the problem is in the SQL level?

➔Look for SQL that causes table scan with high bytes

```
SELECT distinct sql_id, object_name,
bytes,partition_start,
partition_stop,cpu_cost,io_cost
FROM v$sql_plan
WHERE options='FULL'AND operation='TABLE ACCESS‘
AND bytes> 10000000;
```

You suspect the problem is related to buffer cache hot spots?

➔ look for problematic sql’s

```
SELECT sql_id,user_id,sum(time_waited),count(*)
FROM wrh$_active_session_history a,$event_name b
WHERE a.event_id=b.event_id AND name IN ('buffer
busy waits','gc buffer busy')
GROUP BY sql_id,user_id;
```

Use Data Dictionary views

➔You suspect the problem relates to shared pool issues:

Check your shared pool library status:

```
SELECT count(*),sql_id,substr(sql_text,0,20)
FROM v$sqlstats
WHERE executions=1 and parse_calls=1
GROUP BY sql_id,substr(sql_text,0,20)
HAVING count(*)>20;
```

Check shared pool size:

```
SELECT component, min_size, current_size,
last_oper_type,last_oper_time
FROM v$sga_dynamic_components;
```

Check memory resize operations (11g):

```
SELECT component,oper_type,initial_size,
final_size,status,start_time,end_time
FROM v$memory_resize_ops
```

Use Data Dictionary views

➔ You suspect it’s a PGA issue:

```
SELECT sid,sql_id,work_area_size,expected_size,
actual_mem_used,max_mem_used,
number_passes,tempseg_size
FROM v$sql_workarea_active ORDER BY sid;
```

➔You suspect IO issues:

```
SELECT snap_id,file#,phyrds, phywrts,
singleblkrds,readtim, writetim,
singleblkrdtim, wait_count
FROM wrh$_filestatxs where file#=11
ORDER BY snap_id;
```

➔You suspect parallel execution issues:

```
SELECT distinct qcsid,degree,req_degree
FROM v$px_session;
```

AWR Related Scripts

- **awrrpt.sql** Displays statistics for a range of snapshot Ids.
- **awrrpti.sql** Displays statistics for a range of snapshot Ids on a specified database and instance.
- **awrsqrpt.sql** Displays statistics of a particular SQL statement for a range of snapshot Ids. Run this report to inspect or debug the performance of a SQL statement.
- **awrsqrpi.sql** Displays statistics of a particular SQL statement for a range of snapshot Ids on a specified database and instance. Run this report to inspect or debug the performance of a SQL statement on a specific database and instance.
- **awrddrpt.sql** Compares detailed performance attributes and configuration settings between two selected time periods.
- **awrddrpi.sql** Report that compares detailed performance attributes and configuration settings between two selected time periods on a specific database and instance.

No AWR? No ADDM? No ASH?

Use Statspack Instead

- A Statspack report does
  - Contain most of the information you might need
  - Gives an overall view
  - Good for system-wide problems
- A Statspack report doesn't
  - Differentiate problem and symptom
  - Identify the root cause
  - Not so useful for specific sessions
- Use additional tools
  - Use SQL_TRACE and tkprof for specific sessions
  - 3rd party tools

Interpreting an AWR Report

RAC

➔Global cache load profile-
lists the number of blocks and messages that were sent and received and the number of Fusion writes.

```
Global Cache Load Profile
~~~~~~~~~~~~~~~~~~~~~~~~~            Per Second       Per Transaction
                                   ---------------   -----------------
 Global Cache blocks received:       13.93                 9.64
   Global Cache blocks served:       16.03                11.09
    GCS/GES messages received:       63.75                44.10
        GCS/GES messages sent:       79.59                55.06
           DBWR Fusion writes:       0.25                 0.17
Estd Interconnect traffic (KB)       267.65
```

The estimated interconnect traffic (per second) = ((blocks sent + blocks received)*block size + (messages sent + messages received)*message size

Interpreting an AWR Report

RAC

➔Service Statistics- Shows the resources used by all the service instance supports

```
Service Statistics
-> ordered by DB Time
                                                      Physical    Logical
Service Name          DB Time (s)   DB CPU (s)          Reads       Reads
--------------------- ------------ ------------ ---------- ----------
SYS$USERS                2,472.1       244.7       114,990    7,509,339
eeatlp1                  13.6          13.4        0          56
SYS$BACKGROUND           0.0           0.0         256        104,847
eeatlp1XDB               0.0           0.0         0          0
----------------------------------------------------------------------
```

Interpreting an AWR Report

11gR2

➔Memory Resize OPS - Shows information about memory resize operations.

```
Memory Resize Operations Summary

                        Min          Max          Avg     Re-
  Component          Size (Mb)    Size (Mb)    Size (Mb)  Sizes  Grows  Shrink
 --------------- ----------- ----------- ----------- ------ ------ -------
 DEFAULT buffer      1,488.00     1,520.00     1,504.00      3      1       2
 shared pool           608.00       640.00       624.00      3      2       1
 -------------------------------------------------------------------------
```

```
Memory Resize
                    Ela               Oper        Init         Target   Final
   Start            (s)  Component   Typ/Mod Size (M  Delta  Delta      (M) Sta
 -------------- ----- ----------- ------- ------- ------ ------ ------ ----
 09/02 15:46:08      0 bufcache    SHR/IMM   1,520    -16    N/A  1,504 COM
 09/02 15:46:08      0 bufcache    SHR/IMM   1,504    -16    N/A  1,488 COM
 09/02 15:46:08      0 bufcache    GRO/DEF   1,488     32    N/A  1,520 COM
 09/02 15:46:08      0 shared      SHR/DEF     640    -32    N/A    608 COM
 09/02 15:46:08      0 shared      GRO/IMM     624     16    N/A    640 COM
 09/02 15:46:08      0 shared      GRO/IMM     608     16    N/A    624 COM
 ---------------------------------------------------------------------------
```

OS Stats section on AWR report

```
Statistic      Total      Comment
-------------- ---------- ----------------------------------------
AVG_BUSY_TIME      77,518  /* BUSY_TIME / NUM_CPUS */
AVG_IDLE_TIME     281,226  /* IDLE_TIME / NUM_CPUS */
AVG_IOWAIT_TIME    24,128  /* IOWAIT_TIME / NUM_CPUS */
AVG_SYS_TIME        5,664  /* SYS_TIME / NUM_CPUS */
AVG_USER_TIME      71,747  /* USER_TIME / NUM_CPUS */
BUSY_TIME         621,022  /* time eq of %usr+%sys in sar output */
IDLE_TIME       2,250,637  /* time equiv of %idle in sar */
IOWAIT_TIME       193,913  /* time equiv of %wio in sar */
SYS_TIME           46,166  /* time equiv of %sys in sar */
USER_TIME         574,856  /* time equiv of %usr in sar */
LOAD                    0  /* ??? */
OS_CPU_WAIT_TIME 677,100  /* time waiting on run queues */
RSRC_MGR_CPU_WAIT_TIME 0  /* time waited coz of resource manager */
PHYSICAL_MEMORY_BYTES  16,508,780,544  /* total memory in use */
NUM_CPUS                8  /* number of CPUs reported by OS */
NUM_CPU_CORES          4  /* number of CPU sockets on motherboard */
```

to convert the times (expressed in seconds) back into percentages, then total elapsed time is :

- Total elapsed time = BUSY_TIME + IDLE_TIME + IOWAIT TIME

OR

- Total elapsed time = SYS_TIME + USER_TIME + IDLE_TIME + IOWAIT_TIME

Modifying Snapshot Settings

Basic Settings:

INTERVAL affects how often in minutes that snapshots are automatically generated.

RETENTION affects how long in minutes that snapshots are stored in the workload repository.

TOPNSQL affects the number of Top SQL to flush for each SQL criteria (Elapsed Time, CPU Time, Parse Calls, Shareable Memory, and Version Count).

The value for this setting will not be affected by the statistics/flush level and will override the system default behaviour for the AWR SQL collection.

View the current AWR retention settings:

```
SELECT * FROM dba_hist_wr_control;
```

Modifying Snapshot Settings

To adjust the settings, use the MODIFY_SNAPSHOT_SETTINGS procedure.

For example:

```
BEGIN
DBMS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS
(retention => 43200,
interval => 10,
topnsql => 50);
END;
/
```

In this example, the retention period is specified as 43200 minutes (30 days), the interval between each snapshot is specified as 10 minutes, and the number of Top SQL to flush for each SQL criteria as 50.

Useful Commands

➔ Create baseline, save the data for future analysis

```
exec dbms_workload_repository.create_baseline
(start_snap_id => 1003, end_snap_id => 1013,
baseline_name => 'baseline_OCT10');
```

To see stored baselines use dba_hist_baseline view

➔ Export AWR data and Import to different database

```
exec DBMS_SWRF_INTERNAL.AWR_EXTRACT (dmpfile=>
  awr_data.dmp', mpdir => 'DIR_BDUMP',
  bid => 1003, eid => 1013);
```

```
exec DBMS_SWRF_INTERNAL.AWR_LOAD (SCHNAME =>
  'AWR_TEST', dmpfile => 'awr_data.dmp',
  dmpdir => 'DIR_BDUMP');
```

Summary

- Know your system: DB, Application & O/S.
- Keep a track on your base statistics (baseline).
- Look for correlations between AWR and other tuning tools.
- When comparing, use the same time characteristics window.