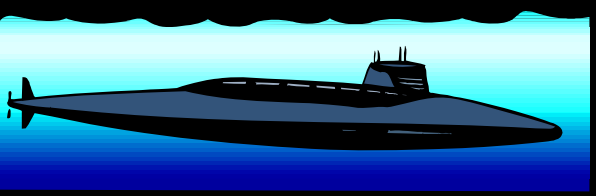
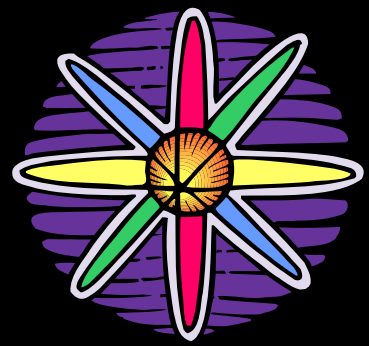
AWR Report Detailed Analysis

Mike Ault
Oracle Guru
Texas Memory Systems

A Texas Memory Systems Presentation



Michael R. Ault
Oracle Guru

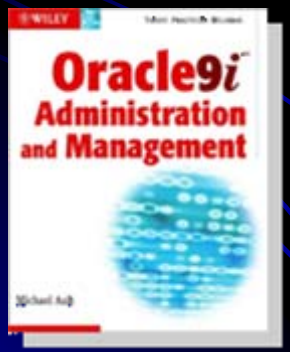


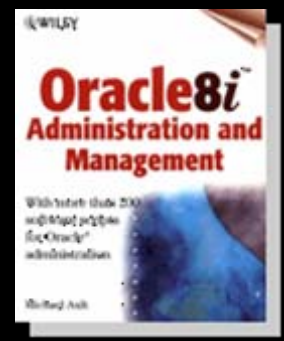
- Nuclear Navy 6 years
- Nuclear Chemist/Programmer 10 years
- Kennedy Western University Graduate
- Bachelors Degree Computer Science
- Certified in all Oracle Versions Since 6
- Oracle DBA, author, since 1990

ORACLE | CERTIFIED PROFESSIONAL

TMS
TEXAS MEMORY SYSTEMS

Books by Michael R. Ault





A Texas Memory Systems Presentation

TMS
TEXAS MEMORY SYSTEMS

Statspackanalyzer.com

Free Statspack/AWR Analysis

Sponsored by Texas Memory Systems

-Looks for IO bottlenecks and other configuration issues.

-Straightforward tuning advice

A Texas Memory Systems Presentation

Introduction

- Statspack was introduced in 8.1.7
- AWR came out in Oracle10g
- Both are very similar
- Both provide a top-down look at performance statistics

What Is AWR

- A background process
- A set of tables
- A set of reports
- Takes snapshots of statistics every hour
- Takes snapshot of high-cost SQL every hour

Preparation for Analysis

- Know your systems normal performance fingerprint
- Be familiar with Concepts and Tuning Guides
- Have “normal” AWR/Statspacks for comparison

Top-Down Approach

- Report starts with settings overview
- Next provides Top-5 waits
- Use the Waits to guide further investigation

AWR Report Header

```
WORKLOAD REPOSITORY report for
DB Name         DB Id    Instance     Inst Num Startup Time    Release     RAC
------------ ----------- ------------ -------- --------------- ----------- ---
AULTDB        4030696936 aultdb1             1 04-Aug-08 10:16 11.1.0.6.0  YES
Host Name        Platform                         CPUs Cores Sockets Memory(GB)
---------------- -------------------------------- ---- ----- ------- ----------
aultlinux3       Linux IA (32-bit)                   2     1       1       2.97
              Snap Id      Snap Time      Sessions Curs/Sess
            --------- ------------------- -------- ---------
Begin Snap:        91 04-Aug-08 12:00:15        41       1.2
  End Snap:        92 04-Aug-08 13:00:28        47       1.1
   Elapsed:               60.22 (mins)
   DB Time:              139.52 (mins)
Cache Sizes                     Begin        End
~~~~~~~~~~~                ---------- ----------
             Buffer Cache:     1,312M     1,312M  Std Block Size:         8K
         Shared Pool Size:       224M       224M      Log Buffer:    10,604K
```

Know Your Load Type!

- Online Transaction Processing
  - Few reads
  - Many writes
  - Many small transactions
  - Look for redo/undo and sequential read issues
- Decision Support/Data Warehouse
  - Many reads
  - Few writes (other then possible temp)
  - Few transactions
  - Look for sort/workarea and scattered read issues
- Mixed or Hybrid

Load Profile Section

Load Profile	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	2.3	7.1	0.63	1.05
DB CPU(s):	0.3	0.9	0.07	0.13
Redo size:	800.5	2,461.8		
Logical reads:	6,307.6	19,396.7		
Block changes:	3.6	10.9		
Physical reads:	2,704.9	8,317.8		
Physical writes:	86.9	267.3		
User calls:	2.2	6.8		
Parses:	2.0	6.1		
Hard parses:	0.0	0.1		
W/A MB processed:	932,965.4	2,868,990.9		
Logons:	0.1	0.2		
Executes:	3.7	11.3		
Rollbacks:	0.1	0.3		
Transactions:	0.3			

What Are Your Efficiencies

- Should be close to 100%
- Parse issues usually are a result of:
  - Bad bind variable usage
  - Insufficient memory
  - Will also be co-indicated by low percentage of memory for multiple SQL execution

Load Profile Section

```
Instance Efficiency Percentages (Target 100%)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
            Buffer Nowait %:  100.00       Redo NoWait %:   99.97
            Buffer  Hit   %:   96.09    In-memory Sort %:  100.00
            Library Hit   %:   98.17        Soft Parse %:   97.88
         Execute to Parse %:   45.80         Latch Hit %:   99.95
Parse CPU to Parse Elapsd %:    0.00     % Non-Parse CPU:   99.77
 Shared Pool Statistics        Begin    End
                              ------  ------
             Memory Usage %:   81.53   85.39
    % SQL with executions>1:   79.29   79.48
  % Memory for SQL w/exec>1:   76.73   78.19
```

Top 5 Waits Section

- Critical to look closely at this section
- Use highest wait times to guide investigation
  - DB FILE type waits – physical IO
  - BUFFER type waits – Logical IO
  - LOG type waits – Redo related
  - PX – Parallel Query
  - GC – Global Cache (RAC related)

Top 5 Waits Section

Top 5 Timed Foreground Events
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
db file sequential read	465,020	3,969	9	47.4	User I/O
DB CPU		995		11.9	
db file parallel read	2,251	322	143	3.8	User I/O
db file scattered read	15,268	153	10	1.8	User I/O
gc current block 2-way	108,739	116	1	1.4	Cluster
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

CPU and Memory

- Watch for number of CPUs
- Pay attention to changes in Memory size
- An idle CPU can be a bad thing
- Always look at IO Wait verses CPU usage
- If the system is IO bound CPU will be idle!

CPU and Memory Sections

```
Host CPU (CPUs:    2 Cores:    1 Sockets:    1)
~~~~~~~~          Load Average
                Begin       End     %User   %System      %WIO     %Idle
            --------- --------- --------- --------- --------- ---------
                 0.37      3.05      10.6       6.7      45.3      82.6
Instance CPU
~~~~~~~~~~~~
              % of total CPU for Instance:  14.8
              % of busy  CPU for Instance:  85.0
  %DB time waiting for CPU - Resource Mgr:   0.0
Memory Statistics
~~~~~~~~~~~~~~~~~                           Begin          End
                    Host Mem (MB):      3,041.4      3,041.4
                     SGA use (MB):      1,584.0      1,584.0
                     PGA use (MB):        169.0        301.7
    % Host Mem used for SGA+PGA:        57.64        57.64
```

RAC Specific Sections

- If you are on a Real Application Cluster these show up
- If not on RAC they don't
- Show health of Global Cache (GC)
- Show health of Global Enqueue (GES)
- Show health of interconnect (Latency, send, receive times)
- 1gb interconnect = 100 MB/sec (approx)

RAC Load Profiles

```
RAC Statistics  DB/Inst: AULTDB/aultdb1  Snaps: 91-92
                              Begin   End
                              ----- -----
          Number of Instances:     2     2
Global Cache Load Profile
~~~~~~~~~~~~~~~~~~~~~~~~~                  Per Second       Per Transaction
                                        ---------------  ---------------
  Global Cache blocks received:                 26.51            81.54
    Global Cache blocks served:                 26.02            80.01
     GCS/GES messages received:                156.31           480.68
         GCS/GES messages sent:                157.74           485.06
            DBWR Fusion writes:                  0.01             0.04
 Estd Interconnect traffic (KB)                481.59
Global Cache Efficiency Percentages (Target local+remote 100%)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Buffer access -  local cache %:   95.44
Buffer access - remote cache %:    0.65
Buffer access -         disk %:    3.91
```

Global Cache and Enqueue Workload

- Watch for timings
- If interconnect latency>IO subsystem latency, RAC is a bottleneck
- Components of time can show where issue is located

Global Cache and Enqueue

```
Global Cache and Enqueue Services - Workload Characteristics
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                    Avg global enqueue get time (ms):      0.2
          Avg global cache cr block receive time (ms):      1.8
      Avg global cache current block receive time (ms):      1.8
             Avg global cache cr block build time (ms):      0.0
              Avg global cache cr block send time (ms):      0.1
       Global cache log flushes for cr blocks served %:      0.8
             Avg global cache cr block flush time (ms):     17.5
          Avg global cache current block pin time (ms):      0.0
         Avg global cache current block send time (ms):      0.1
  Global cache log flushes for current blocks served %:      0.0
        Avg global cache current block flush time (ms):     20.0
Global Cache and Enqueue Services - Messaging Statistics
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                    Avg message sent queue time (ms):      1.1
              Avg message sent queue time on ksxp (ms):      1.3
                 Avg message received queue time (ms):      0.1
                Avg GCS message process time (ms):      0.0
                Avg GES message process time (ms):      0.0
                       % of direct sent messages:    35.13
                     % of indirect sent messages:    64.34
                    % of flow controlled messages:     0.54
          -------------------------------------------------------
```

Global Cache and Enqueue

The most important statistics in this entire section are:

```
     Avg global cache cr block receive time (ms):      1.8
Avg global cache current block receive time (ms):      1.8
```

These should be compared to an AWR report run on the other instance:

```
     Avg global cache cr block receive time (ms):      2.1
Avg global cache current block receive time (ms):      1.7
```

If the numbers on both or all RCA instances aren't similar then this could indicate a problem with the interconnect either at the OS buffer level or the NIC or interface cards themselves.

Time Model Statistics

- In early versions had to do this manually
- Some Statspack won't have this
- Shows where system is spending its time
- Generally you want SQL processing time high, parsing and other stuff low
- If SQL time>>DB CPU time then probably have IO issues

Time Model Statistics

Time Model Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Total time in database user-calls (DB Time): 8371.3s
-> Statistics including the word "background" measure background process time, and so do not contribute to the DB time statistic
-> Ordered by % or DB time desc, Statistic name

Statistic Name	Time (s)	% of DB Time
sql execute elapsed time	8,145.5	97.3
DB CPU	995.1	11.9
parse time elapsed	7.4	.1
hard parse elapsed time	5.2	.1
PL/SQL execution elapsed time	4.8	.1
Java execution elapsed time	0.7	.0
hard parse (sharing criteria) elapsed time	0.2	.0
sequence load elapsed time	0.1	.0
repeated bind elapsed time	0.1	.0
PL/SQL compilation elapsed time	0.0	.0
failed parse elapsed time	0.0	.0
hard parse (bind mismatch) elapsed time	0.0	.0
DB time	8,371.3	
background elapsed time	214.7	
background cpu time	75.8	

Operating System Statistics

- What you get depends on OS
- Some may not give IO timings

Operating System Statistics

Operating System Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> *TIME statistic values are diffed.
All others display actual values. End Value is displayed if different

-> ordered by statistic type (CPU Use, Virtual Memory, Hardware Config), Name

Statistic	Value	End Value
BUSY_TIME	126,029	
IDLE_TIME	597,505	
IOWAIT_TIME	327,861	
NICE_TIME	766	
SYS_TIME	48,452	
USER_TIME	76,784	
LOAD	0	3
PHYSICAL_MEMORY_BYTES	3,189,190,656	
NUM_CPUS	2	
NUM_CPU_CORES	1	
NUM_CPU_SOCKETS	1	
GLOBAL_RECEIVE_SIZE_MAX	4,194,304	
GLOBAL_SEND_SIZE_MAX	262,144	
TCP_RECEIVE_SIZE_DEFAULT	87,380	
TCP_RECEIVE_SIZE_MAX	1,048,576	
TCP_RECEIVE_SIZE_MIN	4,096	
TCP_SEND_SIZE_DEFAULT	65,536	
TCP_SEND_SIZE_MAX	1,048,576	
TCP_SEND_SIZE_MIN	4,096	

Operating System Statistics

Operating System Statistics - Detail DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Snap Time	Load	%busy	%user	%sys	%idle	%iowait
04-Aug 12:00:15	0.4	N/A	N/A	N/A	N/A	N/A
04-Aug 13:00:28	3.0	17.4	10.6	6.7	45.3	82.6

Foreground Wait events

- Foreground=user processes
- Usually most important
- Usual source for top 5 wait events
- 2 sections – classes and events
- Classes are the rolled up sums for waits

Foreground Wait Classes

Foreground Wait Class DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> s - second, ms - millisecond - 1000th of a second
-> ordered by wait time desc, waits desc
-> %Timeouts: value of 0 indicates value was < .5%. Value of null is truly 0
-> Captured Time accounts for 68.9% of Total DB time 8,371.33 (s)
-> Total FG Wait Time: 4,770.85 (s) DB CPU time: 995.13 (s)

Wait Class	Waits	%Time -outs	Total Wait Time (s)	Avg wait (ms)	%DB time
User I/O	518,267	0	4,449	9	53.1
DB CPU			995		11.9
Cluster	188,753	9	173	1	2.1
Other	3,806,446	100	146	0	1.7
Concurrency	1,854	2	2	1	0.0
Commit	15	0	1	39	0.0
Application	740	0	0	0	0.0
System I/O	40	0	0	3	0.0
Network	6,970	0	0	0	0.0
Configuration	0		0		0.0

Foreground Wait Events

Foreground Wait Events DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> s - second, ms - millisecond - 1000th of a second
-> Only events with Total Wait Time (s) >= .001 are shown
-> ordered by wait time desc, waits desc (idle events last)
-> %Timeouts: value of 0 indicates value was < .5%. Value of null is truly 0

Event	Waits	%Time -outs	Total Wait Time (s)	Avg wait (ms)	Waits /txn	% DB time
db file sequential read	465,020	0	3,969	9	395.8	47.4
db file parallel read	2,251	0	322	143	1.9	3.8
db file scattered read	15,268	0	153	10	13.0	1.8
gc current block 2-way	108,739	11	116	1	92.5	1.4
PX Deq: reap credit	3,247,703	100	107	0	2,764.0	1.3
gc cr grant 2-way	57,265	7	28	0	48.7	.3
gc cr multi block request	22,451	6	23	1	19.1	.3
enq: BF - allocation conte	14	93	14	983	0.0	.2
PX qref latch	555,843	100	9	0	473.1	.1
IPC send completion sync	1,070	52	8	8	0.9	.1
gc remaster	22	0	5	221	0.0	.1

Background Wait Events

- SMON, PMON, DBWR, LMON, LMS, etc process waits
- Usually not a big contributor
- Types of waits should correspond to foreground waits

Background Wait Events

Background Wait Events DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by wait time desc, waits desc (idle events last)
-> Only events with Total Wait Time (s) >= .001 are shown
-> %Timeouts: value of 0 indicates value was < .5%. Value of null is truly 0

Event	Waits	%Time -outs	Total Wait Time (s)	Avg wait (ms)	Waits /txn	% bg time
control file sequential re	8,336	0	72	9	7.1	33.5
control file parallel writ	1,287	0	31	24	1.1	14.5
db file parallel write	792	0	11	14	0.7	5.3
log file parallel write	701	0	11	15	0.6	4.9
events in waitclass Other	44,191	98	5	0	37.6	2.5
library cache pin	449	0	2	4	0.4	.8
db file sequential read	221	0	2	7	0.2	.8
gc cr multi block request	1,915	0	2	1	1.6	.7
os thread startup	19	0	1	56	0.0	.5
gc cr block 2-way	246	0	0	1	0.2	.2
db file scattered read	18	0	0	12	0.0	.1
db file parallel read	3	0	0	59	0.0	.1
gc current grant 2-way	98	0	0	1	0.1	.1

Wait event Histograms

- Allows you to see wait time distributions
- Not available in earlier versions of Statspack

Wait Event Histograms

Wait Event Histogram DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Units for Total Waits column: K is 1000, M is 1000000, G is 1000000000
-> % of Waits: value of .0 indicates value was <.05%. Value of null is truly 0
-> % of Waits: column heading of <=1s is truly <1024ms, >1s is truly >=1024ms
-> Ordered by Event (idle events last)

		% of Waits							
Event	Total Waits	<1ms	<2ms	<4ms	<8ms	<16ms	<32ms	<=1s	>1s
control file parallel writ	**1287**					**59.0**	**24.1**	**16.9**	
control file sequential re	**9147**		**23.4**	**21.3**	**23.3**	**22.3**	**6.8**	**2.9**	**.0**
db file parallel read	**2256**			**.3**	**1.0**	**7.4**	**32.6**	**56.8**	**1.9**
db file parallel write	792	.5	.8	4.2	28.7	50.0	8.8	7.1	
db file scattered read	**15K**		**.4**	**2.7**	**31.5**	**59.2**	**5.8**	**.5**	
db file sequential read	**465K**	**.0**	**.6**	**2.2**	**49.5**	**45.0**	**2.3**	**.4**	
gc cr grant 2-way	50K	87.2	11.1	1.3	.3	.2		.0	
gc cr multi block request	24K	59.0	36.8	3.0	.5	.6	.0		
gc current block 2-way	84K	6.5	87.7	5.2	.3	.2	.0		
library cache lock	488	82.8	10.9	4.9	1.0	.2	.2		
library cache pin	4371	77.6	11.1	7.4	3.1	.6	.0		
gcs remote message	274K	28.5	15.4	9.9	11.6	7.5	5.8	21.4	
ges remote message	53K	11.4	3.3	2.7	1.9	1.8	2.1	76.8	

Service Related Statistics

- New in 10g and above
- A service is a grouping of processes
- Users may be grouped in SYS$USER
- Application logins (single user) may be grouped with that user name

Service Statistics

Service Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by DB Time

Service Name	DB Time (s)	DB CPU (s)	Physical Reads (K)	Logical Reads (K)
aultdb	8,344	981	9,769	22,715
SYS$USERS	23	12	1	56
SYS$BACKGROUND	1	0	1	17
aultdbXDB	0	0	0	0

Service Wait Class Statistics

```
Service Wait Class Stats                    DB/Inst: AULTDB/aultdb1  Snaps: 91-92
-> Wait Class info for services in the Service Statistics section.
-> Total Waits and Time Waited displayed for the following wait
   classes:  User I/O, Concurrency, Administrative, Network
-> Time Waited (Wt Time) in seconds
Service Name
------------------------------------------------------------
 User I/O  User I/O  Concurcy  Concurcy     Admin     Admin   Network   Network
Total Wts   Wt Time Total Wts   Wt Time Total Wts   Wt Time Total Wts   Wt Time
--------- --------- --------- --------- --------- --------- --------- ---------
aultdb
   517710      4446       234         1         0         0      5828         0
SYS$USERS
      555         3      1615         1         0         0      1140         0
SYS$BACKGROUND
      350         3      3486         4         0         0         0         0
          ----------------------------------------------------------
```

The SQL Sections

- Total Elapsed Time – IO waits
- Total CPU Time – Sorting, hashing
- Total Buffer Gets – High logical IO
- Total Disk Reads – High physical IO
- Total Executions – May indicate loop issues
- Total Parse Calls – Memory issues
- Total Sharable Memory - Informational
- Total Version Count – May indicate unsafe bind variables
- Total Cluster Wait Time – Indicates physical issues (RPB, block size)

Instance Activity Statistics

- Many statistics here
- Some are useful
- Use to calculate ratios with other stats

Instance Activity Statistics

Instance Activity Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
CPU used by this session	77,997	21.6	66.4
CPU used when call started	288,270	79.8	245.3
DB time	2,547,336	705.0	2,168.0
Effective IO time	909,644	251.8	774.2
Number of read IOs issued	27,685	7.7	23.6
SQL*Net roundtrips to/from clien	6,970	1.9	5.9
bytes received via SQL*Net from	2,385,638	660.2	2,030.3
bytes sent via SQL*Net to client	2,595,626	718.4	2,209.0
consistent gets	22,777,682	6,303.9	19,385.3
consistent gets - examination	6,073,207	1,680.8	5,168.7
consistent gets direct	3,277,142	907.0	2,789.1
consistent gets from cache	14,648,585	4,054.1	12,466.9
consistent gets from cache (fast	193,221	53.5	164.4
db block changes	12,812	3.6	10.9
db block gets	13,389	3.7	11.4
db block gets from cache	13,364	3.7	11.4
db block gets from cache (fastpa	3,512	1.0	3.0
dirty buffers inspected	825	0.2	0.7

Instance Activity Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
enqueue timeouts	40	0.0	0.0
enqueue waits	499	0.1	0.4
execute count	13,287	3.7	11.3
free buffer inspected	556,747	154.1	473.8
free buffer requested	731,667	202.5	622.7
gc CPU used by this session	11,859	3.3	10.1
gc blocks lost	0	0.0	0.0
gc cr block build time	1	0.0	0.0
gc cr block flush time	7	0.0	0.0
gc cr block receive time	66	0.0	0.1
gc cr block send time	3	0.0	0.0
gc cr blocks received	361	0.1	0.3
gc cr blocks served	522	0.1	0.4
gc current block flush time	2	0.0	0.0
gc current block pin time	205	0.1	0.2
gc current block receive time	16,726	4.6	14.2
gc current block send time	577	0.2	0.5
gc current blocks received	95,445	26.4	81.2
gc current blocks served	93,484	25.9	79.6

Instance Activity Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
index fast full scans (direct re	90	0.0	0.1
index fast full scans (full)	4	0.0	0.0
index fast full scans (rowid ran	90	0.0	0.1
index fetch by key	3,086,965	854.3	2,627.2
index scans kdiixs1	29,551	8.2	25.2
leaf node 90-10 splits	19	0.0	0.0
leaf node splits	26	0.0	0.0
opened cursors cumulative	13,077	3.6	11.1
parse count (failures)	2	0.0	0.0
parse count (hard)	153	0.0	0.1
parse count (total)	7,202	2.0	6.1
parse time cpu	227	0.1	0.2
parse time elapsed	399	0.1	0.3
physical read IO requests	550,974	152.5	468.9
physical read bytes	32,562,569,216	9,011,916.7	27,712,824.9
physical read total IO requests	605,019	167.4	514.9
physical read total bytes	32,711,421,952	9,053,112.7	27,839,508.0
physical read total multi block	30,330	8.4	25.8
physical reads	9,773,380	2,704.9	8,317.8
physical reads cache	572,745	158.5	487.4
physical reads cache prefetch	153,965	42.6	131.0
physical reads direct	3,402,178	941.6	2,895.5
physical reads direct temporary	124,434	34.4	105.9
physical reads prefetch warmup	58,580	16.2	49.9

Instance Activity Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
physical write IO requests	4,983	1.4	4.2
physical write bytes	1,037,123,584	287,031.1	882,658.4
physical write total IO requests	15,031	4.2	12.8
physical write total bytes	1,085,801,472	300,503.1	924,086.4
physical write total multi block	4,062	1.1	3.5
physical writes	314,090	86.9	267.3
physical writes direct	124,459	34.4	105.9
physical writes direct (lob)	0	0.0	0.0
physical writes direct temporary	124,434	34.4	105.9
physical writes from cache	2,143	0.6	1.8
physical writes non checkpoint	124,952	34.6	106.3
recursive calls	78,415	21.7	66.7
recursive cpu usage	77,189	21.4	65.7
redo entries	7,832	2.2	6.7
redo log space requests	2	0.0	0.0
redo log space wait time	28	0.0	0.0
redo size	2,892,568	800.5	2,461.8
redo synch time	66	0.0	0.1
redo synch writes	72	0.0	0.1
redo wastage	196,192	54.3	167.0
redo write time	1,110	0.3	0.9
redo writes	701	0.2	0.6

Instance Activity Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
rollback changes - undo records	0	0.0	0.0
session cursor cache hits	12,415	3.4	10.6
session logical reads	22,791,070	6,307.6	19,396.7
sorts (memory)	3,875	1.1	3.3
sorts (rows)	1,460,468	404.2	1,243.0
summed dirty queue length	3,284	0.9	2.8
table fetch by rowid	1,322,667	366.1	1,125.7
table fetch continued row	13	0.0	0.0
table scan blocks gotten	2,780,775	769.6	2,366.6
table scan rows gotten	158,164,979	43,773.3	134,608.5
table scans (direct read)	776	0.2	0.7
table scans (long tables)	776	0.2	0.7
table scans (rowid ranges)	776	0.2	0.7
table scans (short tables)	2,255	0.6	1.9
transaction rollbacks	0	0.0	0.0
undo change vector size	1,870,904	517.8	1,592.3
user I/O wait time	445,246	123.2	378.9
user calls	7,943	2.2	6.8
user commits	794	0.2	0.7
user rollbacks	381	0.1	0.3
workarea executions - onepass	6	0.0	0.0
workarea executions - optimal	2,323	0.6	2.0

Instance Activity Statistics

```
Instance Activity Stats - Absolute Values
DB/Inst: AULTDB/aultdb1  Snaps: 91-9
-> Statistics with absolute values (should not be diffed)
Statistic                            Begin Value       End Value
-------------------------------- --------------- ---------------
session pga memory max               544,192,924   4,940,081,136
session cursor cache count                 2,266           8,279
session uga memory                73,033,165,084   3.393545E+11
opened cursors current                        48              54
workarea memory allocated                      0          16,041
logons current                                41              47
session uga memory max             4,427,536,236   5,963,059,828
session pga memory                   390,773,148     826,689,340
          ------------------------------------------------------------
Instance Activity Stats - Thread Activity
DB/Inst: AULTDB/aultdb1  Snaps: 91-92
-> Statistics identified by '(derived)' come from sources other than SYSSTAT
Statistic                                 Total  per Hour
-------------------------------- ----------------- ---------
log switches (derived)                        1      1.00
          ------------------------------------------------------------
```

Tablespace IO Statistics

- Show reads, writes and read times
- Do not show write times
- Use to determine where IO load is from
- Use file level stats to determine hot files (which partition is most active for example)

Tablespace IO Statistics

Tablespace IO Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by IOs (Reads + Writes) desc

Tablespace	Reads	Av Reads/s	Av Rd(ms)	Av Blks/Rd	Writes	Av Writes/s	Buffer Waits	Av Buf Wt(ms)
DATA	512,639	142	11.8	6.4	0	0	6	151.7
INDEXES	32,625	9	11.3	16.7	0	0	37	83.5
TEMP	4,024	1	17.6	30.9	4,014	1	0	0.0
SYSAUX	571	0	29.3	1.4	698	0	0	0.0
SYSTEM	471	0	5.3	1.8	56	0	0	0.0
UNDOTBS1	9	0	10.0	1.0	215	0	1	10.0
USERS	1	0	10.0	1.0	0	0	0	0.0

Data File IO Statistics

File IO Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by Tablespace, File

Tablespace	Filename	Reads	Av Reads/s	Av Rd(ms)	Av Blks/Rd	Writes	Av Writes/s	Buffer Waits	Av Buf Wt(ms)
DATA	+DATA/aultdb/datafile/data.257.660765277	501,566	139	10.8	5.2	0	0	6	151.7
DATA	+DATA/aultdb/datafile/data.259.660843403	11,073	3	56.9	64.5	0	0	0	0.0
INDEXES	+DATA/aultdb/datafile/indexes.258.660765437	32,625	9	11.3	16.7	0	0	37	83.5
SYSAUX	+DATA2/aultdb/datafile/sysaux.257.660755929	571	0	29.3	1.4	698	0	0	0.0
SYSTEM	+DATA2/aultdb/datafile/system.256.660755927	471	0	5.3	1.8	56	0	0	0.0
TEMP	+DATA2/aultdb/tempfile/temp.266.660756117	4,020	0	17.5	31.0	4,014	0	0	N/A
UNDOTBS1	+DATA2/aultdb/datafile/undotbs1.258.660755929	9	0	10.0	1.0	215	0	1	10.0
USERS	+DATA2/aultdb/datafile/users.259.660755929	1	0	10.0	1.0	0	0	0	0.0

Buffer Pool Statistics

- Buffer pools (default, keep, recycle, 2-32K)
- Shows how efficiently they are used
- Can help with sizing

Buffer Pool Statistics

Buffer Pool Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Standard block size Pools D: default, K: keep, R: recycle
-> Default Pools for other block sizes: 2k, 4k, 8k, 16k, 32k

P	Number of Buffers	Pool Hit%	Buffer Gets	Physical Reads	Physical Writes	Free Buff Wait	Writ Comp Wait	Buffer Busy Waits
D	159,244	91	6,287,434	572,581	2,143	0	0	44

Instance Recovery Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> B: Begin snapshot, E: End snapshot

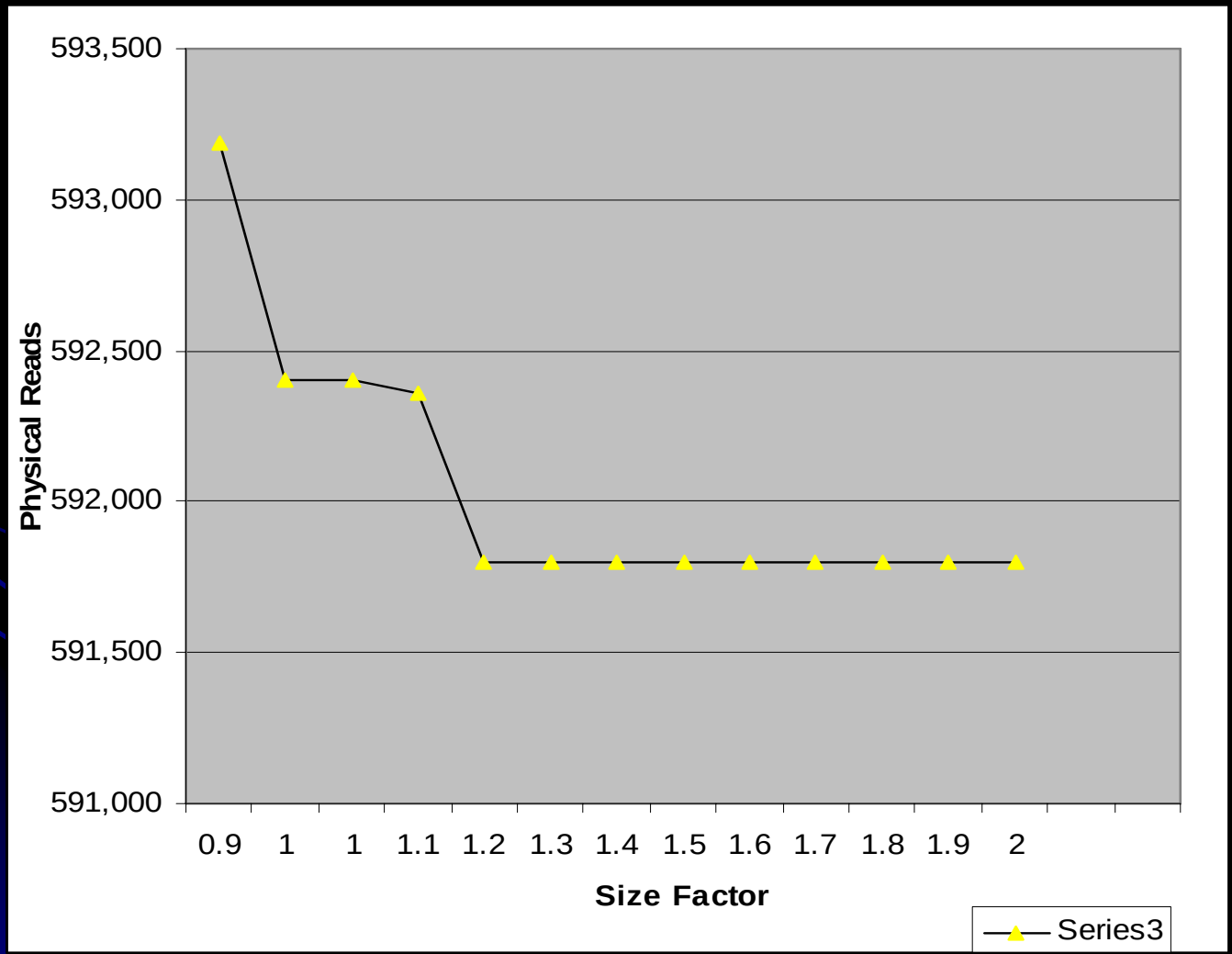
	Targt MTTR (s)	Estd MTTR (s)	Recovery Estd IOs	Actual Redo Blks	Target Redo Blks	Log File Size Redo Blks	Log Ckpt Timeout Redo Blks	Log Ckpt Interval Redo Blks
B	0	0	250	974	1015	92160	1015	N/A
E	0	0	292	1192	2751	92160	2751	N/A

Buffer Pool Advisory

Buffer Pool Advisory DB/Inst: AULTDB/aultdb1 Snap: 92
-> Only rows with estimated physical reads >0 are displayed
-> ordered by Block Size, Buffers For Estimate

P	Size for Est (M)	Size Factor	Buffers for Estimate	Est Phys Read Factor	Estimated Physical Reads
…					
D	896	.7	108,752	1.0	620,121
D	1,024	.8	124,288	1.0	599,692
D	1,152	.9	139,824	1.0	593,191
D	1,280	1.0	155,360	1.0	592,402
D	1,312	1.0	159,244	1.0	592,402
D	1,408	1.1	170,896	1.0	592,356
D	1,536	1.2	186,432	1.0	591,798
D	1,664	1.3	201,968	1.0	591,798
D	1,792	1.4	217,504	1.0	591,798
D	1,920	1.5	233,040	1.0	591,798
D	2,048	1.6	248,576	1.0	591,798
D	2,176	1.7	264,112	1.0	591,798
D	2,304	1.8	279,648	1.0	591,798
D	2,432	1.9	295,184	1.0	591,798
D	2,560	2.0	310,720	1.0	591,798

Buffer Pool Advisory



PGA Statistics

- Process Global Area – User processes
- Sort, Hash, Bitmap, Global Temporary operations
- Only Sorts really tracked
- Investigate if temporary IO high, but no disk sorts indicated
- V$SORT_USAGE good source
- V$SQL_WORKAREA_ACTIVE good for sorts and hashes
- Rule of thumb for size based on histogram:
  - High Optimal*20=PGA_AGGREGATE_TARGET

PGA Aggregate Summary

- PGA Cache Hit Percent is the total number of bytes processed in the PGA versus the total number of bytes processed plus extra bytes read/written in extra passes.
- Low values mean we need a higher PGA_AGGREGATE_TARGET setting

PGA Statistics

PGA Aggr Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> PGA cache hit % - percentage of W/A (WorkArea) data processed only in-memory

PGA Cache Hit %	W/A MB Processed	Extra W/A MB Read/Written
54.8	2,843	2,345

PGA Aggr Target Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

PGA Aggr Target Histogram DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> Optimal Executions are purely in-memory operations

Low Optimal	High Optimal	Total Execs	Optimal Execs	1-Pass Execs	M-Pass Execs
2K	4K	1,833	1,833	0	0
64K	128K	5	5	0	0
128K	256K	1	1	0	0
256K	512K	6	6	0	0
512K	1024K	439	439	0	0
1M	2M	6	6	0	0
2M	4M	6	6	0	0
4M	8M	14	14	0	0
8M	16M	6	6	0	0
16M	32M	4	4	0	0
64M	128M	3	3	0	0
256M	512M	6	0	6	0

PGA Statistics

PGA Memory Advisory DB/Inst: AULTDB/aultdb1 Snap: 92

-> When using Auto Memory Mgmt, minimally choose a pga_aggregate_target value where Estd PGA Overalloc Count is 0

PGA Target Est (MB)	Size Factr	W/A MB Processed	Estd Extra W/A MB Read/ Written to Disk	Estd P Cache Hit %	Estd PGA Overallo Count	Estd Time
64	0.1	3,388.1	6,390.6	35.0	22	1.6E+05
128	0.3	3,388.1	5,795.7	37.0	2	1.5E+05
256	0.5	3,388.1	4,885.0	41.0	1	1.3E+05
384	0.8	3,388.1	1,172.5	74.0	0	74,015
512	1.0	3,388.1	1,172.5	74.0	0	74,015
614	1.2	3,388.1	1,172.5	74.0	0	74,015
717	1.4	3,388.1	1,172.5	74.0	0	74,015
819	1.6	3,388.1	1,172.5	74.0	0	74,015
922	1.8	3,388.1	1,172.5	74.0	0	74,015
1,024	2.0	3,388.1	1,172.5	74.0	0	74,015
1,536	3.0	3,388.1	1,172.5	74.0	0	74,015
2,048	4.0	3,388.1	1,172.5	74.0	0	74,015
3,072	6.0	3,388.1	1,172.5	74.0	0	74,015
4,096	8.0	3,388.1	1,172.5	74.0	0	74,015

PGA Aggregate Summary

The example report didn't include a section on this so a section from another report has been used.

	PGA Aggr Target(M)	Auto PGA Target(M)	PGA Mem Alloc(M)	W/A PGA Used(M)	PGA W/A % Mem	%Auto W/A Mem	%Man W/A Mem	Global Mem Bound(K)
B	1,628	1,434	425.37	284.23	66.82	100.00	0.00	166,700
E	1,628	1,424	315.79	177.43	56.19	100.00	0.00	166,700

Shared Pool Advisory

- Some times not very useful
- Depend more on the shrink and grow sections or V$SGA_RESIZE_OPS

Shared Pool Advisory

Shared Pool Advisory DB/Inst: AULTDB/aultdb1 Snap: 92

-> SP: Shared Pool Est LC: Estimated Library Cache Factr: Factor

-> Note there is often a 1:Many correlation between a single logical object in the Library Cache, and the physical number of memory objects associated with it. Therefore comparing the number of Lib Cache objects (e.g. in v$librarycache), with the number of Lib Cache Memory Objects is invalid.

Shared Pool Size(M)	SP Size Factr	Est LC Size (M)	Est LC Mem Obj	Est LC Time Saved (s)	Est LC Time Saved Factr	Est LC Load Time (s)	Est LC Load Time Factr	Est LC Mem Obj Hits (K)
192	.9	3	495	4,555	1.0	41	1.0	12
224	1.0	33	4,350	4,555	1.0	41	1.0	96
256	1.1	45	6,645	4,557	1.0	39	1.0	96
288	1.3	45	6,645	4,558	1.0	38	.9	96
320	1.4	45	6,645	4,558	1.0	38	.9	96
352	1.6	45	6,645	4,558	1.0	38	.9	96
384	1.7	45	6,645	4,558	1.0	38	.9	96
416	1.9	45	6,645	4,558	1.0	38	.9	96
448	2.0	45	6,645	4,558	1.0	38	.9	96

Other Advisories

- SGA Target
  - Helps for SGA_TARGET settings
- Streams Pool
  - Only if streams are used, if you are getting spills, indicates pool is too small
- Java Pool
  - Only if you are using internal Java, similar to the PL/SQL area in the library caches

SGA Target Advisory

SGA Target Advisory DB/Inst: AULTDB/aultdb1
Snap: 92

SGA Target Size (M)	SGA Size Factor	Est DB Time (s)	Est Physical Reads
396	0.3	8,538	592,206
792	0.5	8,536	592,206
1,188	0.8	8,536	592,206
1,584	1.0	8,536	592,206
1,980	1.3	8,536	592,206
2,376	1.5	8,542	592,206
2,772	1.8	8,542	592,206
3,168	2.0	8,542	592,206

Streams and Java Pool Advisor

Streams Pool Advisory DB/Inst: AULTDB/aultdb1 Snap: 92

Size for Est (MB)	Size Factor	Est Spill Count	Est Spill Time (s)	Est Unspill Count	Est Unspill Time (s)
32	0.13	0	0	0	0
64	0.25	0	0	0	0
…					
608	2.38	0	0	0	0
640	2.50	0	0	0	0

Java Pool Advisory DB/Inst: AULTDB/aultdb1 Snap: 92

Java Pool Size(M)	JP Size Factr	Est LC Size (M)	Est LC Mem Obj	Est LC Time Saved (s)	Est LC Time Saved Factr	Est LC Load Time (s)	Est LC Load Time Factr	Est LC Mem Obj Hits
16	1.0	2	79	7	1.0	41	1.0	88
32	2.0	4	163	7	1.0	41	1.0	182

Buffer Waits Statistics

- In the old days we just got “buffer waits”
- Now they break it down for you
  - Data block – block sharing (block too big)
  - Undo header – Insufficient number of undo segments
  - File header block – freelist, freelist group issues
  - 1st level bmb – ASSM bitmap issues
  - Segment header – freelist, freelist group issues.
  - 2nd level bmb. – ASSM bitmap issues
- Look at the ITL waits section for more guidance

Buffer Waits Statistics

Buffer Wait Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by wait time desc, waits desc

Class	Waits	Total Wait Time (s)	Avg Time (ms)
data block	43	4	93
undo header	1	0	10

Enqueue Statistics

- Enqueues are serialization tools (247 of them)
- They “line up” access to memory or other resources
- In earlier versions were rather cryptic
- Now naming has improved

Enqueue Statistics

Enqueue Activity DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> only enqueues with waits are shown
-> Enqueue stats gathered prior to 10g should not be compared with 10g data
-> ordered by Wait Time desc, Waits desc

Enqueue Type (Request Reason)

Requests	Succ Gets	Failed Gets	Waits	Wt Time (s)	Av Wt Time(ms)
BF-BLOOM FILTER (allocation contention)					
2,618	2,611	7	14	14	982.14
TD-KTF map table enqueue (KTF dump entries)					
9	9	0	9	0	37.78
PS-PX Process Reservation					
648	616	32	208	0	.96
CF-Controlfile Transaction					
7,661	7,660	1	118	0	1.27
TM-DML					
5,559	5,559	0	16	0	3.75
XL-ASM Extent Fault Lock (fault extent map)					
14	14	0	1	0	60.00

Enqueue Statistics

- V$SESSION_WAIT and V$LOCK give more data about enqueues
- The P1, P2 and P3 values tell what the enqueue may have been waiting on
- For BF we get node#, parallelizer#, and bloom#

```
column parameter1 format a15
column parameter2 format a15
column parameter3 format a15
column lock format a8
Select
       substr(name,1,7) as
"lock",parameter1,parameter2,parameter3 from v$event_name
where name like 'enq%'
```

Undo Statistics

- Used to be rollback
- Parameters and some statistics still use rollback in their name
- May show issues with automatic undo tuning
- Transactions_per_rollback_segment, undo_retention can help control number and size of undo segments as well as undo tablespace size
- 10 created to begin with, then ratio of processes to TPRBS determine how many
- Only used for DML not selects

Undo Statistics

Undo Segment Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Min/Max TR (mins) - Min and Max Tuned Retention (minutes)
-> STO - Snapshot Too Old count, OOS - Out of Space count
-> Undo segment block stats:
-> uS - unexpired Stolen, uR - unexpired Released, uU - unexpired reUsed
-> eS - expired Stolen, eR - expired Released, eU - expired reUsed

Undo TS#	Num Undo Blocks (K)	Number of Transactions	Max Qry Len (s)	Max Tx Concurcy	Min/Max TR (mins)	STO/ OOS	uS/uR/uU/ eS/eR/eU
2	.1	1,090	1,725	3	18.8/42.8	0/0	0/0/0/0/0/0

Undo Segment Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Most recent 35 Undostat rows, ordered by Time desc

End Time	Num Undo Blocks	Number of Transactions	Max Qry Len (s)	Max Tx Concy	Tun Ret (mins)	STO/ OOS	uS/uR/uU/ eS/eR/eU
04-Aug 12:56	14	167	890	3	29	0/0	0/0/0/0/0/0
04-Aug 12:46	10	141	289	3	19	0/0	0/0/0/0/0/0
04-Aug 12:36	10	163	1,725	2	43	0/0	0/0/0/0/0/0
04-Aug 12:26	18	240	1,124	3	33	0/0	0/0/0/0/0/0
04-Aug 12:16	9	133	901	2	29	0/0	0/0/0/0/0/0
04-Aug 12:06	88	246	300	3	19	0/0	0/0/0/0/0/0

Latch Statistics

- There are a plethora of latch statistics
- Misses, unless they cause significant amount of sleeps aren't of concern
- Sleeps can be a problem
- May need to look at spin count if you have excessive sleeps
- Spin count (undocumented (_SPIN_COUNT) was based on CPU speed and 2000 setting was several years ago
- If *latch waits* or other latch related events aren't showing up, then latches probably aren't an issue
- Usually cache buffer and shared pool related latches are the major latches.

Latch Statistics

Latch Activity DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> "Get Requests", "Pct Get Miss" and "Avg Slps/Miss" are statistics for
willing-to-wait latch get requests
-> "NoWait Requests", "Pct NoWait Miss" are for no-wait latch get requests
-> "Pct Misses" for both should be very close to 0.0

Latch Name	Get Requests	Pct Get Miss	Avg Slps /Miss	Wait Time (s)	NoWait Requests	Pct NoWait Miss
KJC message pool free li	14,582	0.3	0.0	0	15,463	0.1
gc element	2,414,376	0.0	0.3	2	7,880	0.0
gcs resource hash	1,861,484	0.0	0.5	1	6	0.0
virtual circuit queues	1	0.0		0	0	N/A

Latch Sleep Breakdown DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by misses desc

Latch Name	Get Requests	Misses	Sleeps	Spin Gets
cache buffers chains	8,492,860	21,037	3	21,034
simulator lru latch	1,823,879	12,065	311	11,774
cache buffers lru chain	1,190,948	6,096	352	5,799
gc element	2,414,376	767	213	582
KJCT flow control latch	443,643	735	11	725

Latch Statistics

Latch Miss Sources DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> only latches with sleeps are shown
-> ordered by name, sleeps desc

Latch Name	Where	NoWait Misses	Sleeps	Waiter Sleeps
cache buffers lru chain	kcbzgws_1	0	248	272
gc element	kclnfndnewm	0	112	18
gcs resource hash	kjbassume	0	88	0

Mutex Sleep Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92
No data exists for this section of the report.

Parent Latch Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
No data exists for this section of the report.

Child Latch Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
No data exists for this section of the report.

Segment Access Areas

- Logical Reads – If excessive logical reads, this identifies the segments
- Physical Reads – If excessive physical reads this identifies the segments
- Lock Waits – If excessive enqueues (TX usually) look here
- ITL waits – header or segment buffer busy waits look here
- Buffer Busy waits – general buffer busy waits, look here
- Lots of blocks being transferred, look at the next three to determine which ones.
  - GC buffer busy
  - CR blocks received
  - Current blocks received

Segment Access Areas

Segments by Logical Reads DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Total Logical Reads: 22,791,070
-> Captured Segments account for 41.8% of Total

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	Logical Reads	%Total
TPCH	INDEXES	H_ORDERS_IDX1		INDEX	4,294,720	18.84
TPCH	DATA	H_LINEITEM		TABLE	2,117,568	9.29
TPCH	DATA	H_ORDER		TABLE	1,017,136	4.46
TPCH	INDEXES	SUPPLIER_IDX1		INDEX	626,848	2.75
TPCH	DATA	H_SUPPLIER		TABLE	620,432	2.72

Segments by Physical Reads DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Total Physical Reads: 9,773,380
-> Captured Segments account for 39.3% of Total

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	Physical Reads	%Total
TPCH	DATA	H_LINEITEM		TABLE	2,107,980	21.57
TPCH	DATA	H_ORDER		TABLE	894,131	9.15
** UNAVAIL	** UNAVAIL	** UNAVAILABLE **	AILABLE **	UNDEF	511,994	5.24
TPCH	DATA	H_PART		TABLE	123,676	1.27
TPCH	DATA	H_PARTSUPP		TABLE	117,400	1.20

Segment Access Areas

Segments by Row Lock Waits DB/Inst: AULTDB/aultdb1 Snaps: 91-92

No data exists for this section of the report.

Segments by ITL Waits DB/Inst: AULTDB/aultdb1 Snaps: 91-92

No data exists for this section of the report.

Segments by Buffer Busy Waits DB/Inst: AULTDB/aultdb1 Snaps: 91-92

No data exists for this section of the report.

Segments by Global Cache Buffer Busy DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> % of Capture shows % of GC Buffer Busy for each top segment compared
-> with GC Buffer Busy for all segments captured by the Snapshot

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	GC Buffer Busy	% of Capture
** UNAVAIL	** UNAVAIL	** UNAVAILABLE **	AILABLE **	UNDEF	9	81.82
TPCH	INDEXES	H_ORDERS_IDX1		INDEX	1	9.09
TPCH	INDEXES	PARTSUPP_IDX1		INDEX	1	9.09

Segment Access Areas

Segments by CR Blocks Received DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Total CR Blocks Received: 361
-> Captured Segments account for 35.5% of Total

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	CR Blocks Received	%Total
SYS	SYSTEM	JOB$		TABLE	22	6.09
SYS	SYSAUX	SMON_SCN_TIME		TABLE	21	5.82
SYSMAN	SYSAUX	MGMT_SYSTEM_PERFORMA		TABLE	12	3.32
SYSMAN	SYSAUX	MGMT_SYSTEM_PERF_LOG		INDEX	12	3.32
SYSMAN	SYSAUX	MGMT_TASK_QTABLE		TABLE	12	3.32

Segments by Current Blocks Received DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Total Current Blocks Received: 95,445
-> Captured Segments account for 99.9% of Total

Owner	Tablespace Name	Object Name	Subobject Name	Obj. Type	Current Blocks Received	%Total
TPCH	INDEXES	H_ORDERS_IDX1		INDEX	65,524	68.65
TPCH	DATA	H_ORDER		TABLE	24,149	25.30
TPCH	DATA	H_SUPPLIER		TABLE	2,232	2.34
SYS	SYSTEM	TAB$		TABLE	996	1.04
** UNAVAIL	** UNAVAIL	** UNAVAILABLE **	AILABLE **	UNDEF	776	.81

Library Cache Activity Sections

- Library caches are usually auto-tuned
- Not since version 6 have we had control
- Look to big changes in usage to show problems
- Sequences are a good example
- Segment extents are another

Library Cache Activity Sections

Dictionary Cache Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> "Pct Misses" should be very low (< 2% in most cases)
-> "Final Usage" is the number of cache entries being used

Cache	Get Requests	Pct Miss	Scan Reqs	Pct Miss	Mod Reqs	Final Usage
dc_awr_control	63	3.2	0	N/A	0	1
dc_database_links	2	0.0	0	N/A	0	1
dc_files	8	0.0	0	N/A	0	8
dc_global_oids	2,826	0.2	0	N/A	0	133
dc_histogram_data	1,151	11.6	0	N/A	0	750
dc_histogram_defs	3,213	5.6	0	N/A	0	3,460
dc_object_grants	484	0.0	0	N/A	0	17
dc_objects	7,172	1.0	0	N/A	17	2,203
dc_profiles	64	0.0	0	N/A	0	1
dc_rollback_segments	850	0.0	0	N/A	0	22
dc_segments	1,020	5.9	0	N/A	4	728
dc_sequences	13	30.8	0	N/A	13	0
dc_tablespaces	9,757	0.0	0	N/A	0	9
dc_users	13,294	0.0	0	N/A	0	142
global database name	4,485	0.0	0	N/A	0	1
outstanding_alerts	52	69.2	0	N/A	2	1

Library Cache Activity Sections

Dictionary Cache Stats (RAC) DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Cache	GES Requests	GES Conflicts	GES Releases
dc_awr_control	2	2	0
dc_global_oids	5	0	0
dc_histogram_defs	181	0	0
dc_objects	71	0	0
dc_segments	68	5	0
dc_sequences	26	5	0
dc_tablespaces	1	0	0
dc_users	5	0	0
outstanding_alerts	100	36	0

Library Cache Activity Sections

Library Cache Activity DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> "Pct Misses" should be very low

Namespace	Get Requests	Pct Miss	Pin Requests	Pct Miss	Reloads	Invali-dations
BODY	1,514	0.0	1,858	0.2	4	0
CLUSTER	44	0.0	16	0.0	0	0
INDEX	2	0.0	2	0.0	0	0
JAVA DATA	2	0.0	0	N/A	0	0
SQL AREA	2,246	1.5	17,091	2.5	121	6
TABLE/PROCEDURE	12,745	0.1	16,155	1.4	166	0
TRIGGER	376	0.0	423	0.0	0	0

Library Cache Activity (RAC) DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Namespace	GES Lock Requests	GES Pin Requests	GES Pin Releases	GES Inval Requests	GES Invali-dations
CLUSTER	16	16	0	0	0
INDEX	2	2	0	0	0
TABLE/PROCEDURE	4,553	15,492	0	0	0

Memory Dynamic Components Sections

- In 10g and Above memory is dynamic
- SGA_MAX_SIZE, SGA_TARGET in 10g control memory allocation for pools
- PGA_AGGREGATE_TARGET in 10g controls PGA allocations
- MEMORY_MAX_SIZE and MEMORY_TARGET controls the whole shooting match in 11g

Memory Dynamic Components Sections

Memory Dynamic Components DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Min/Max sizes since instance startup
-> Oper Types/Modes: INItializing,GROw,SHRink,STAtic/IMMediate,DEFerred
-> ordered by Component

Component	Begin Snap Size (Mb)	Current Size (Mb)	Min Size (Mb)	Max Size (Mb)	Oper Count	Last Op Typ/Mod
ASM Buffer Cach	.00	.00	.00	.00	0	STA/
DEFAULT 16K buf	.00	.00	.00	.00	0	STA/
DEFAULT 2K buff	.00	.00	.00	.00	0	STA/
DEFAULT 32K buf	.00	.00	.00	.00	0	STA/
DEFAULT 4K buff	.00	.00	.00	.00	0	STA/
DEFAULT 8K buff	.00	.00	.00	.00	0	STA/
DEFAULT buffer	1,312.00	1,312.00	1,296.00	1,328.00	2	GRO/DEF
KEEP buffer cac	.00	.00	.00	.00	0	STA/
PGA Target	512.00	512.00	512.00	512.00	0	STA/
RECYCLE buffer	.00	.00	.00	.00	0	STA/
SGA Target	1,584.00	1,584.00	1,584.00	1,584.00	0	STA/
Shared IO Pool	.00	.00	.00	.00	0	STA/
java pool	16.00	16.00	16.00	16.00	0	STA/
large pool	16.00	16.00	16.00	16.00	0	STA/
shared pool	224.00	224.00	208.00	240.00	2	SHR/DEF
streams pool	.00	.00	.00	.00	0	STA/

Memory Resize Operations Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> Resizes, Grows, Shrinks - Operations captured by AWR
   if there are operations on the same component for the same
   operation_type, target_size, and with the same start_time
   only one operation is captured
-> ordered by Component

Component	Min Size (Mb)	Max Size (Mb)	Avg Size (Mb)	Re-Sizes	Grows	Shrink
DEFAULT buffer	1,296.00	1,312.00	1,304.00	2	1	1
shared pool	224.00	240.00	232.00	2	1	1

Memory Resize Ops DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> Oper Types/Modes: INItializing,GROw,SHRink,STAtic/IMMediate,DEFerred
   Delta : change in size of the component
   Target Delta: displayed only if final size <> target_size
-> Status: COMplete/CANcelled/INActive/PENding/ERRor
-> ordered by start_time desc,component

Start	Ela (s)	Component	Oper Typ/Mod	Init Size (M	Delta	Target Delta	Final (M)	Sta
08/04 12:39:06	0	bufcache	GRO/DEF	1,296	16	N/A	1,312	COM
08/04 12:39:06	0	shared	SHR/DEF	240	-16	N/A	224	COM
08/04 12:02:59	2	bufcache	SHR/DEF	1,312	-16	N/A	1,296	COM
08/04 12:02:59	2	shared	GRO/DEF	224	16	N/A	240	COM

Process Memory Sections

- Controlled by PGA_AGGREGATE_TARGET in 10g
- MEMORY_MAX_SIZE and MEMORY_TARGET or PGA_AGGREGATE_TARGET in 11g

Process/SGA Memory Sections

Process Memory Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> B: Begin snap E: End snap
-> All rows below contain absolute values (i.e. not diffed over the interval)
-> Max Alloc is Maximum PGA Allocation size at snapshot time
-> Hist Max Alloc is the Historical Max Allocation for still-connected processes
-> ordered by Begin/End snapshot, Alloc (MB) desc

	Category	Alloc (MB)	Used (MB)	Avg Alloc (MB)	Std Dev Alloc (MB)	Max Alloc (MB)	Hist Max Alloc (MB)	Num Proc	Num Alloc
B	Other	153.1	N/A	3.5	4.9	24	24	44	44
	Freeable	13.3	.0	.9	.9	4	N/A	14	14
	JAVA	1.6	1.6	.8	1.2	2	2	2	2
	SQL	1.2	.5	.1	.1	0	3	20	13
	PL/SQL	.2	.1	.0	.0	0	0	42	42
E	Other	175.3	N/A	3.5	4.6	24	24	50	50
	Freeable	101.5	.0	5.3	10.1	28	N/A	19	19
	SQL	23.2	22.5	.9	1.7	5	166	26	19
	JAVA	1.6	1.6	.8	1.2	2	2	2	2
	PL/SQL	.2	.1	.0	.0	0	0	48	48

SGA Sections

- Look for sections whose “free” areas don’t change (may be too large)
- Look for areas with rapid unwarranted growth (memory leaks)

SGA Memory Sections

SGA Memory Summary DB/Inst: AULTDB/aultdb1 Snaps: 91-92

SGA regions	Begin Size (Bytes)	End Size (Bytes) (if different)
Database Buffers	1,375,731,712	
Fixed Size	1,300,968	
Redo Buffers	10,858,496	
Variable Size	671,090,200	
sum	2,058,981,376	

SGA breakdown difference DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> ordered by Pool, Name
-> N/A value for Begin MB or End MB indicates the size of that Pool/Name was insignificant, or zero in that snapshot

Pool	Name	Begin MB	End MB	% Diff
java	free memory	7.7	7.7	0.00
java	joxlod exec hp	8.1	8.1	0.00
java	joxs heap	.3	.3	0.00
large	ASM map operations hashta	.2	.2	0.00
large	PX msg pool	.3	.4	41.67
large	free memory	15.5	15.4	-0.79
shared	CCursor	3.4	4.0	15.27
shared	Heap0: KGL	4.0	4.1	1.74
shared	KGL handle	6.6	6.6	0.26
shared	KGLS heap	3.8	4.6	20.93
shared	KQR M PO	2.8	3.0	5.87
shared	PCursor	2.6	2.6	1.61
shared	PL/SQL MPCODE	12.8	13.0	1.53
shared	free memory	41.4	32.7	-20.92
shared	sql area	14.1	16.6	17.03
shared	type object de	2.7	2.7	0.02
	buffer_cache	1,312.0	1,312.0	0.00
	fixed_sga	1.2	1.2	0.00
	log_buffer	10.4	10.4	0.00

Streams Component Sections

- Only populated if you use streams
- These are taken from a different instance
- Look for excessive timings
- Look for spills to disk

Streams Component Sections

```
Streams CPU/IO Usage                    DB/Inst: AULTDB/aultdb1  Snaps: 91-92
-> Streams processes ordered by CPU usage -> CPU and I/O Time in micro
seconds   Session Type                  CPU Time  User I/O Time   Sys I/O
Time ----------------------- ------------ ------------ -------------
STREAMS Capture                2,128,000            0             0 STREAMS
Apply Reader                     609,945        2,050         1,341 Logminer
Builder                          185,835            0             0 Logminer
Preparer                         175,559            0             0 QMON
Slaves                           132,888            0             0 Logminer
Reader                            97,009            0       887,687 STREAMS Apply
Server                35,580                 0                0 STREAMS Apply
Coordinator              747                 0                0 QMON
Coordinator                          454                 0             0 Propagation
Sender                         0                 0                0          ----------
-------------------------------------------------   Streams
Capture                              DB/Inst: AULTDB/aultdb1  Snaps: 91-92 -> Lag
Change should be small or negative (in seconds)                  Captured
Enqueued                  Pct       Pct       Pct       Pct                    Per
  Per       Lag RuleEval  Enqueue RedoWait    Pause Capture
Name   Second   Second   Change     Time     Time     Time     Time -----------
- ------- -------- -------- -------- -------- -------- --------
STREAM_CAP          65       39       93        0       23        0       71
       ----------------------------------------------------------   Streams
```

Streams Component Sections

Apply /Inst: AULTDB/aultdb1 Snaps: 91-92

-> Pct DB is the percentage of all DB transactions that this apply handled

-> WDEP is the wait for dependency -> WCMT is the wait for commit

-> RBK is rollbacks -> MPS is messages per second

-> TPM is time per message in milli-seconds

-> Lag Change should be small or negative (in seconds)

Apply Name	Applied TPS	Pct DB	Pct WDEP	Pct WCMT	Pct RBK	Applied MPS	Dequeue TPM	Apply TPM	Lag Change
STREAM_APP	0	0	0	0	0	0	0	0	0

Buffered Queues /Inst: AULTDB/aultdb1 Snaps: 91-92 -> The Spill Rate should be very close to zero -> The Diff in Percentage Spilled should be very close to zero or negative

Queue Schema and Name	Incoming per second	Outgoing per second	Spilled per second	Diff Pct Spilled
STRMADMIN.REP_CAPTURE	39	39	0	0
STRMADMIN.REP_DEST_QU	0	0	0	0

Streams Component Sections

Buffered Subscribers /Inst: AULTDB/aultdb1 Snaps: 91-92 -> All
Subscribers should have a zero spill rate

Subscriber Name	Incoming per second	Outgoing per second	Spilled per second
STREAM_APP	793	793	6
STREAM_APP	0	0	0
PROXY: "STRMADMIN"."	391	391	0
PROXY: CATLIBRAR.ASU	0	0	0
PROXY: CATLIBRAR.ASU	-793	-793	-6

Rule Set /Inst: AULTDB/aultdb1 Snaps: 91-92 ->
Rule Sets ordered by Evaluations

Ruleset Name	Evals	Fast Evals	SQL Execs	CPU Time	Elapsed Time
STRMADMIN.RULESET$_10	3,174	0	3,174	1,565	2,784
SYS.ALERT_QUE_R	2	0	0	0	0
STRMADMIN.RULESET$_6	0	0	0	0	0
STRMADMIN.RULESET$_3	0	0	0	0	0
STRMADMIN.RULESET$_8	0	0	0	0	0

Resource Limit Statistics

Resource Limit Stats Inst: AULTDB/aultdb1 Snaps: 91-92 -
> only rows with Current or Maximum Utilization > 80% of Limit are shown ->
ordered by resource

name	Current	Maximum	Initial	Resource
Name	Utilization	Utilization	Allocation	Limit
sessions	51	65	150	150

Initialization Parameter Changes

- Generally speaking initialization parameters should be static
- Usually the DBA will make changes
- This section shows parameters with non-default values
- If things changed and you didn’t do it, investigate!

init.ora Parameters DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> if IP/Public/Source at End snap is different a '*' is displayed

Parameter Name	Begin value	End value (if different)
audit_file_dest	/home/oracle/app/product/oracle/a	
audit_trail	DB	
cluster_database	TRUE	
cluster_database_instances	2	
compatible	11.1.0.0.0	
control_files	+DATA2/aultdb/controlfile/current	
db_block_size	8192	
db_create_file_dest	+DATA2	
db_domain		
db_name	aultdb	
db_recovery_file_dest	+DATA2	
db_recovery_file_dest_size	2147483648	
diagnostic_dest	/home/oracle/app/product/oracle	
dispatchers	(PROTOCOL=TCP) (SERVICE=aultdbXDB	
instance_number	1	
memory_target	2197815296	
open_cursors	300	
processes	150	
remote_listener	LISTENERS_AULTDB	
remote_login_passwordfile	EXCLUSIVE	
spfile	+DATA/aultdb/spfileaultdb.ora	
star_transformation_enabled	TRUE	
thread	1	
undo_tablespace	UNDOTBS1	

Global Enqueue and Other RAC Sections

- RAC adds many more statistics
- Additional sections were added at the end of the report if you have RAC
- Much of these are helpful if you are ding RAC tuning
- This is a bit beyond this papers scope
- We will take a quick look

Global Enqueues

Global Enqueue Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total	per Second	per Trans
gcs immediate cr (null) converts	213,759	59.2	181.9
gcs msgs received	549,546	152.1	467.7
gcs side channel msgs logical	146,320	40.5	124.5
messages queue sent logical	142,750	39.5	121.5
messages received actual	236,086	65.3	200.9
messages received logical	564,794	156.3	480.7
messages sent directly	134,160	37.1	114.2
messages sent indirectly	245,700	68.0	209.1
messages sent pbatched	306,413	84.8	260.8
msgs received queued	564,808	156.3	480.7
msgs sent queue time (ms)	146,886	40.7	125.0
msgs sent queue time on ksxp (ms)	322,477	89.2	274.4
msgs sent queued	139,264	38.5	118.5
msgs sent queued on ksxp	243,401	67.4	207.1

Global CR Served

Global CR Served Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

Statistic	Total
CR Block Requests	486
CURRENT Block Requests	36
Data Block Requests	486
Undo Block Requests	0
TX Block Requests	7
Current Results	522
Fairness Down Converts	78
Fairness Clears	10
Flushes	4

Global CURRENT Served Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92

-> Pins = CURRENT Block Pin Operations
-> Flushes = Redo Flush before CURRENT Block Served Operations
-> Writes = CURRENT Block Fusion Write Operations

Statistic	Total	% <1ms	% <10ms	% <100ms	% <1s	% <10s
Pins	93,484	99.95	0.00	0.05	0.00	0.00
Flushes	1	0.00	0.00	100.00	0.00	0.00
Writes	43	0.00	4.65	74.42	18.60	2.33

Global Cache Transfers

Global Cache Transfer Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Immediate (Immed) - Block Transfer NOT impacted by Remote Processing Delays
-> Busy (Busy) - Block Transfer impacted by Remote Contention
-> Congested (Congst) - Block Transfer impacted by Remote System Load
-> ordered by CR + Current Blocks Received desc

Inst No	Block Class	CR Blocks Received	CR % Immed	CR % Busy	CR % Congst	Current Blocks Received	Current % Immed	Current % Busy	Current % Congst
2	data block	150	98.0	2.0	.0	95,402	99.8	.0	.2
2	undo header	200	100.0	.0	.0	3	100.0	.0	.0
2	Others	10	100.0	.0	.0	24	100.0	.0	.0

Global Cache Transfers

Global Cache Transfer Times (ms) DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Avg Time - average time of all blocks (Immed,Busy,Congst) in ms
-> Immed, Busy, Congst - Average times in ms
-> ordered by CR + Current Blocks Received desc

Inst No	Block Class	CR Avg Time (ms) All	CR Immed	CR Busy	CR Congst	Current Avg Time (ms) All	Current Immed	Current Busy	Current Congst
2	data blo	2.5	1.9	27.6	N/A	1.8	1.7	N/A	9.1
2	undo hea	1.4	1.4	N/A	N/A	1.6	1.6	N/A	N/A
2	others	1.4	1.4	N/A	N/A	1.4	1.4	N/A	N/A
2	undo blo	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Global Cache Transfers

Global Cache Transfer (Immediate) DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Immediate (Immed) - Block Transfer NOT impacted by Remote Processing Delays
-> % of Blocks Received requiring 2 or 3 hops
-> ordered by CR + Current Blocks Received desc

			CR			Current		
Src Inst	Block Class	Blocks Lost	Immed Blks Received	% 2hop	% 3hop	Immed Blks Received	% 2hop	% 3hop
2	data blo	0	147	100.0	0.0	95,204	100.0	0.0
2	undo hea	0	200	100.0	0.0	3	100.0	0.0
2	others	0	10	100.0	0.0	24	100.0	0.0
2	undo blo	0	0	N/A	N/A	0	N/A	N/A

Global Cache Times Immediate

Global Cache Times (Immediate) DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Blocks Lost, 2-hop and 3-hop Average times in (ms)
-> ordered by CR + Current Blocks Received desc

Src Inst	Block Class	Lost Time	CR Avg Time (ms) Immed	CR Avg Time (ms) 2hop	CR Avg Time (ms) 3hop	Current Avg Time (ms) Immed	Current Avg Time (ms) 2hop	Current Avg Time (ms) 3hop
2	data blo		1.9	1.9	N/A	1.7	1.7	N/A
2	undo hea		1.4	1.4	N/A	1.6	1.6	N/A
2	others		1.4	1.4	N/A	1.4	1.4	N/A
2	undo blo		N/A	N/A	N/A	N/A	N/A	N/A

Interconnect Ping Latency Stats DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Ping latency of the roundtrip of a message from this instance to -> target in
-> The target instance is identified by an instance number.
-> Average and standard deviation of ping latency is given in miliseconds
-> for message sizes of 500 bytes and 8K.
-> Note that latency of a message from the instance to itself is used as
-> control, since message latency can include wait for CPU

Target Instance	500B Ping Count	Avg Latency 500B msg	Stddev 500B msg	8K Ping Count	Avg Latency 8K msg	Stddev 8K msg
1	360	.64	1.04	360	.63	1.04
2	360	1.39	2.43	360	2.16	1.87

Interconnect Throughput

Interconnect Throughput by Client DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Throughput of interconnect usage by major consumers.
-> All throughput numbers are megabytes per second

Used By	Send Mbytes/sec	Receive Mbytes/sec
Global Cache	.20	.21
Parallel Query	.35	.39
DB Locks	.03	.03
DB Streams	.00	.00
Other	.00	.00

Interconnect Device Statistics

Interconnect Device Statistics DB/Inst: AULTDB/aultdb1 Snaps: 91-92
-> Throughput and errors of interconnect devices (at OS level).
-> All throughput numbers are megabytes per second

Device Name	IP Address	Public	Source

	Send Mbytes/sec	Send Errors	Send Dropped	Send Buffer Overrun	Send Carrier Lost
	Receive Mbytes/sec	Receive Errors	Receive Dropped	Receive Buffer Overrun	Receive Frame Errors
eth0	11.1.1.1		NO	Oracle Cluster Repository	
.77	0	0	0	0	
.82	1	0	0	1	

In Conclusion….

- AWR reports contain massive amounts of data
- Analysis of AWR can be complex
- Take a top down approach and review the areas that are pertinent
- Use review tools such as StatspackAnalyzer.com

Questions/Comments?

Thank You!

Mike Ault
Mike.ault@texmemsys.com