

Tips on Performance Diagnosis Using AWR SQL Report

July 2008

INTRODUCTION

Have you ever looked through an AWR report and needed to see more information about a particular SQL statement? Or perhaps wanted to get more detail on a SQL statement that may have been performing well one day, but had regressed the next? This information is available in AWR and can be viewed using the AWR SQL report that was introduced in Oracle Database 10gR2. The report provides additional information for a SQL statement, including execution plans and corresponding SQL statistics for the different execution plans.

GENERATING A RERORT

The report is generated using the `awrsqrpt.sql` script in `$ORACLE_HOME/rdbms/admin`.

Like the regular AWR report, this script will prompt you for a report format (text or html), the number of days and a pair of begin/end snapshots. It will also prompt for the `sql_id` on which to generate the report, along with a report name. The default report name is

`awrsqrpt_<instance#>_<begin_snap>_<end_snap>.[html|txt]`. Note that the `sql_id` is not part of the report name, so if you are generating reports for multiple `sql_ids` in the same snapshot range, name the reports appropriately.

Use
`$ORACLE_HOME/rdbms/admin/awrsqrpt`
to generate the AWR SQL report

```
SQL> @?/rdbms/admin/awrsqrpt
```

```
Current Instance
~~~~~
```

DB Id	DB Name	Inst Num	Instance
1185083199	ORCL	1	orcl

```
Specify the Report Type
~~~~~
```

```
Would you like an HTML report, or a plain text report?
Enter 'html' for an HTML report, or 'text' for plain text
Defaults to 'html'
```

```
Enter value for report_type: html
```

```
Type Specified: html
```

```
Instances in this Workload Repository schema
~~~~~
```

DB Id	Inst Num	DB Name	Instance	Host
-------	----------	---------	----------	------

```

* 1185083199      1 ORCL      orcl      stacr30

Using 1185083199 for database Id
Using          1 for instance number

Specify the number of days of snapshots to choose from
~~~~~
Entering the number of days (n) will result in the most recent
(n) days of snapshots being listed. Pressing <return> without
specifying a number lists all completed snapshots.

Enter value for num days: 3

Listing the last 3 days of Completed Snapshots

Instance      DB Name      Snap Id      Snap Started      Snap
-----
orcl          ORCL          561 07 Jul 2008 00:00      1
              562 07 Jul 2008 01:00      1
              563 07 Jul 2008 02:00      1
              564 07 Jul 2008 03:00      1
              565 07 Jul 2008 04:00      1
...
              623 09 Jul 2008 14:00      1
              624 09 Jul 2008 15:00      1

Specify the Begin and End Snapshot Ids
~~~~~
Enter value for begin_snap: 623
Begin Snapshot Id specified: 623

Enter value for end_snap: 624
End Snapshot Id specified: 624

Specify the SQL Id
~~~~~
Enter value for sql_id: f3cmbb55dvgr1
SQL ID specified: f3cmbb55dvgr1

Specify the Report Name
~~~~~
The default report file name is awrsqlrpt_1_623_624.html. To use this
name,
press <return> to continue, otherwise enter an alternative.

Enter value for report_name:

Using the report name awrsqlrpt_1_623_624.html

```

REPORT CONTENTS

The first part of the report is similar to the regular AWR report. It provides general information on the database name, instance, the host name. It also includes information about the begin/end snapshots, the elapsed time as well as the DB time accumulated between the two snapshots.

WORKLOAD REPOSITORY SQL Report

Snapshot Period Summary

DB Name	DB Id	Instance	Inst num	Startup Time	Release	RAC
ORCL	1185083199	orcl	1	02-Jul-08 10:07	11.1.0.6.0	NO

	Snap Id	Snap Time	Sessions	Cursors/Session
Begin Snap:	623	09-Jul-08 14:00:47	29	1.8
End Snap:	624	09-Jul-08 15:00:05	43	4.9
Elapsed:		59.31 (mins)		
DB Time:		277.45 (mins)		

SQL Summary

The SQL summary section shows the SQL Id, the total elapsed time accumulated and a fragment of the sql text, hyperlinked to the full SQL text at the bottom of the report. It also includes summary statistics for the execution plans captured during the interval for that one sql_id.

In the following screenshot, we see there are two execution plans captured for the SQL Id during the snapshot interval. From this summary, we see that #1 was executed more often. We also see that both plans were captured during the snapshot id 624.

SQL Summary

SQL Id	Elapsed Time (ms)	Module	Action	SQL Text
f3cmbb55dvgr1	7,033,792	SQL*Plus		SELECT PROD_NAME, PROD_DESC, SUM(QUANTITY_SOLD) FROM PRODUCTS P, SALES...

[Back to Top](#)

SQL ID: f3cmbb55dvgr1

- 1st Capture and Last Capture Snap IDs refer to Snapshot IDs within the snapshot range
- [SELECT PROD_NAME, PROD_DESC, SUM\(QUANTITY_SOLD\) FROM PRODUCTS P, SALES...](#)

#	Plan Hash Value	Total Elapsed Time(ms)	Executions	1st Capture Snap ID	Last Capture Snap ID
1	2855547969	3,565,682	11,932	624	624
2	629759164	3,468,110	9,443	624	624

[Back to Top](#)

Execution Plans and Statistics

The report then shows the details for each of the execution plans – including SQL statistics and along with the execution plan. The plan statistics can be used to compare the performance of the different execution plans that occurred during the interval. The execution plans can then be examined to determine the difference in performance between the plans.

From the following screen shots, we see that #1 was executed more often, and incurred less elapsed time and cpu time per execution than #2. The difference in per execution statistics is minimal which would indicate that these two plans are comparable in performance.

Plan 1(PHV: 2855547969)

- [Plan Statistics](#)
- [Execution Plan](#)

[Back to Top](#)

Plan Statistics

- % Total DB Time is the Elapsed Time of the SQL statement divided into the Total Database Time multiplied by 100

Stat Name	Statement Total	Per Execution	% Snap Total
Elapsed Time (ms)	3,565,682	298.83	21.42
CPU Time (ms)	1,181,925	99.06	22.02
Executions	11,932		
Buffer Gets	1,695,579	142.10	4.75
Disk Reads	2	0.00	0.03
Parse Calls	9	0.00	0.02
Rows	787,299	65.98	
User I/O Wait Time (ms)	18		
Cluster Wait Time (ms)	0		
Application Wait Time (ms)	0		
Concurrency Wait Time (ms)	1,088		
Invalidations	0		
Version Count	3		
Sharable Mem(KB)	17		

[Back to Plan 1\(PHV: 2855547969\)](#)

[Back to Top](#)

Execution Plan

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	Pstart	Pstop
0	SELECT STATEMENT				37 (100)			
1	HASH GROUP BY		56	4200	37 (19)	00:00:01		
2	HASH JOIN		56	4200	36 (17)	00:00:01		
3	VIEW	VW_GBC_5	56	952	33 (19)	00:00:01		
4	HASH GROUP BY		56	1064	33 (19)	00:00:01		
5	FILTER							
6	PARTITION RANGE ITERATOR		18845	349K	31 (13)	00:00:01	KEY	KEY
7	TABLE ACCESS FULL	SALES	18845	349K	31 (13)	00:00:01	KEY	KEY
8	TABLE ACCESS FULL	PRODUCTS	72	4176	3 (0)	00:00:01		

[Back to Plan 1\(PHV: 2855547969\)](#)

[Back to Top](#)

Plan 2(PHV: 629759164)

- [Plan Statistics](#)
- [Execution Plan](#)

[Back to Top](#)

Plan Statistics

- % Total DB Time is the Elapsed Time of the SQL statement divided into the Total Database Time multiplied by 100

Stat Name	Statement Total	Per Execution	% Snap Total
Elapsed Time (ms)	3,468,110	367.27	20.83
CPU Time (ms)	1,140,682	120.79	21.25
Executions	9,443		
Buffer Gets	1,337,227	141.61	3.75
Disk Reads	2	0.00	0.03
Parse Calls	8	0.00	0.01
Rows	621,427	65.81	
User I/O Wait Time (ms)	0		
Cluster Wait Time (ms)	0		
Application Wait Time (ms)	0		
Concurrency Wait Time (ms)	94		
Invalidations	0		
Version Count	3		
Sharable Mem(KB)	42		

[Back to Plan 2\(PHV: 629759164\)](#)

[Back to Top](#)

Execution Plan

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time	Pstart	Pstop
0	SELECT STATEMENT					180 (100)			
1	HASH GROUP BY		3565	254K	1648K	180 (4)	00:00:03		
2	FILTER								
3	HASH JOIN		16244	1158K		39 (13)	00:00:01		
4	TABLE ACCESS FULL	PRODUCTS	72	4176		3 (0)	00:00:01		
5	PARTITION RANGE ITERATOR		16244	237K		35 (12)	00:00:01	KEY	KEY
6	TABLE ACCESS FULL	SALES	16244	237K		35 (12)	00:00:01	KEY	KEY

[Back to Plan 2\(PHV: 629759164\)](#)

[Back to Top](#)

In order to analyze a SQL statement that has regressed from one day to another, simply generate an AWR SQL report for a time when the SQL statement was performing well, and a second AWR SQL report for a comparable time when the SQL statement was performing poorly. The execution plans and the statistics can then be compared between the two reports to determine why the SQL statement has regressed. Not all regressions are due to plan changes – it may be that the statement is executed more often, retrieves more data, or incurs more wait time. Comparing both the statistics and the execution plans can help determine the cause of the regression.

EM SCREENS

The information in the AWR SQL report is also available in the EM Performance screens.

There is a drop down menu to choose the plan hash value for which to display the SQL statistics.

The screenshot displays the Oracle Enterprise Manager Performance page for an AWR SQL report. The page is titled "Database Control" and shows the following information:

- Text:** A SQL query is shown: `SELECT PROD_NAME, PROD_DESC, SUM(QUANTITY_SOLD) FROM PRODUCTS P, SALES S WHERE P.PROD_ID = S.PROD_ID AND S.TIME_ID BETWEEN TO_DATE(:B1, 'dd-mm-yyyy') - 31 AND TO_DATE(:B1, 'dd-mm-yyyy') GROUP BY PROD_NAME, PROD_DESC`
- Details:** A dropdown menu for "Plan Hash Value" is open, showing options: "All", "629759164", and "2855547969". A message states: "There are multiple plans found for this SQL statement."
- Summary:** A line graph titled "Active Sessions" shows the number of active sessions over time from 1:59 to 2:55 on Jul 9, 2008. The y-axis ranges from 0.0 to 2.0.
- General:**
 - Module: SQL*Plus
 - Action: Parsing Schema SH
 - PL/SQL Source (Line Number): SH.DEMOLOAD (25)
 - SQL Profile: n/a
 - SQL Plan Baseline: n/a
- Activity By Waits:** A pie chart shows the distribution of activity by wait event:
 - Remaining Waits (66.9%) - 67%
 - CPU (33.1%) - 33%
- Shared Cursors Statistics:**
 - Total Parses: 9
 - Hard Parses: 1
 - Child Cursors: 1
 - Loaded Plans: 1
 - Invalidations: 0
 - Largest Cursor Size (KB): 17.47
 - All Cursor Size (KB): 17.47
 - First Load Time: Jul 9, 2008 1:52:36 PM (UTC-07:00)
 - Last Load Time: Jul 9, 2008 2:02:46 PM (UTC-07:00)
- Execution Statistics:**

	Total	Per Execution	Per Row
Executions	11,798	1	0.02
Elapsed Time (sec)	3,526.93	0.30	<0.01
CPU Time (sec)	1,168.91	0.10	<0.01
Buffer Gets	1,677,255	142.16	2.15
Disk Reads	2	<0.01	<0.01
Direct Writes	0	0.00	0.00
Rows	778,443	65.98	1
Fetches	790,240	66.98	1.02
- Other Statistics:**
 - Executions that Fetched all Rows (%): 99.97
 - Average Persistent Mem (KB): 14.78
 - Average Runtime Mem (KB): 14.09
 - Serializable Aborts: 0
 - Remote: No
 - Obsolete: No
 - Child Latch Number: 0

CONCLUSION

The AWR SQL report shows detailed SQL information that is available in the EM performance pages. This report can be used to compare the execution plans and the execution statistics of SQL statements, and provides additional detail for SQL that is not available in a regular AWR report.



Tips on Performance Diagnosis Using AWR SQL Report
July 2008

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2008, Oracle and/or its affiliates. All rights reserved.
This document is provided for information purposes only and the contents hereof are subject to change without notice.
This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.