9

Comparing Database Performance Over Time

This chapter describes how to compare database performance over time using Automatic Workload Repository (AWR) Compare Periods reports and contains the following topics:

About Automatic Workload Repository Compare Periods Reports

Performance degradation of the database occurs when your database was performing optimally in the past, but has over time gradually degraded to a point where it becomes noticeable to the users. AWR Compare Periods report enables you to compare database performance over time.

An AWR report shows AWR data during a period in time between two snapshots (or two points in time). The AWR Compare Periods report, on the other hand, shows the difference between two periods in time (or two AWR reports, which equates to four snapshots). Using AWR Compare Periods reports helps you to identify detailed performance attributes and configuration settings that differ between two time periods.

For example, assume that a batch workload runs daily during a maintenance window between 10:00 p.m. and midnight is showing poor performance and is now completing at 2 a.m instead. You can generate an AWR Compare Periods report for the time period from 10:00 p.m. to midnight on a day when performance was good, and another report for the time period from 10:00 a.m. to 2 a.m. on a day when performance was poor. You can then compare these reports to identify configuration settings, workload profile, and statistics that differ between these two time periods. Based on those differences, you can more easily diagnose the cause of the performance degradation.

The two time periods selected in an AWR Compare Periods report can be of different durations because the report normalizes the statistics by the amount of time spent on the database for each time period, and presents statistical data ordered by the largest difference between the time periods.

SQL monitor data from SGA buffers are considered part of the AWR repository. This can be exported as part of the AWR report, and imported.

> **Note:**
>
> Each PDB manages its own SGA memory resources for SQL monitoring. The database administrator should monitor and tune per-PDB SGA limits. The database administrator should also monitor and tune retention policies within the container database (CDB) and pluggable databases (PDBs).

> **See Also:**
>
> - "*Oracle Multitenant Administrator's Guide*" for information about how manageability features, including AWR features, work in a multitenant container database (CDB).
> - "Automatic Workload Repository" for information about the AWR
> - "Generating Automatic Workload Repository Reports" for information about AWR reports

Generating Automatic Workload Repository Compare Periods Reports

If the performance of your database degrades over time, AWR Compare Periods reports enable you to compare two periods in time to identify key differences that can help you diagnose the cause of the performance degradation.

AWR Compare Periods reports are divided into multiple sections. The HTML report includes links that can be used to navigate quickly between sections. The content of the report contains the workload profile of the system for the selected range of snapshots.

- User Interfaces for Generating AWR Compare Periods Reports
- Generating an AWR Compare Periods Report Using the Command-Line Interface

User Interfaces for Generating AWR Compare Periods Reports

The primary interface for generating AWR Compare Periods reports is Oracle Enterprise Manager Cloud Control (Cloud Control). Whenever possible, generate AWR Compare Periods reports using Cloud Control.

If Cloud Control is unavailable, then generate AWR Compare Periods reports by running SQL scripts. The DBA role is required to run these scripts.

> **See Also:**
>
> *Oracle Database 2 Day + Performance Tuning Guide* for information about generating AWR Compare Periods reports using Cloud Control

Generating an AWR Compare Periods Report Using the Command-Line Interface

This topic describes how to generate AWR Compare Periods reports by running SQL scripts in the command-line interface.

- Generating an AWR Compare Periods Report for the Local Database
- Generating an AWR Compare Periods Report for a Specific Database
- Generating an Oracle RAC AWR Compare Periods Report for the Local Database

- Generating an Oracle RAC AWR Compare Periods Report for a Specific Database

Generating an AWR Compare Periods Report for the Local Database

The `awrddrpt.sql` SQL script generates an HTML or text report that compares detailed performance attributes and configuration settings between two selected time periods on the local database instance.

To generate an AWR Compare Periods report on the local database instance using the command-line interface:

1. At the SQL prompt, enter:

   ```
   @$ORACLE_HOME/rdbms/admin/awrddrpt.sql
   ```

2. Specify whether you want an HTML or a text report:

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

   In this example, an HTML report is chosen.

3. Specify the number of days for which you want to list snapshot IDs in the first time period.

   ```
   Enter value for num_days: 2
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the last 2 days are displayed.

4. Specify a beginning and ending snapshot ID for the first time period:

   ```
   Enter value for begin_snap: 102
   Enter value for end_snap: 103
   ```

   In this example, the snapshot with a snapshot ID of 102 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 103 is selected as the ending snapshot for the first time period.

5. Specify the number of days for which you want to list snapshot IDs in the second time period.

   ```
   Enter value for num_days2: 1
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the previous day are displayed.

6. Specify a beginning and ending snapshot ID for the second time period:

   ```
   Enter value for begin_snap2: 126
   Enter value for end_snap2: 127
   ```

   In this example, the snapshot with a snapshot ID of 126 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 127 is selected as the ending snapshot for the second time period.

7. Enter a report name, or accept the default report name:

   ```
   Enter value for report_name:
   Using the report name awrdiff_1_102_1_126.txt
   ```

   In this example, the default name is accepted and an AWR report named `awrdiff_1_102_126` is generated.

Generating an AWR Compare Periods Report for a Specific Database

The `awrddrpi.sql` SQL script generates an HTML or text report that compares detailed performance attributes and configuration settings between two selected time periods on a specific database and instance. This script enables you to specify a database identifier and instance for which AWR Compare Periods report will be generated.

To generate an AWR Compare Periods report on a specific database instance using the command-line interface:

1. At the SQL prompt, enter:

   ```
   @$ORACLE_HOME/rdbms/admin/awrddrpi.sql
   ```

2. Specify whether you want an HTML or a text report:

   ```
   Enter value for report_type: text
   ```

   In this example, a text report is chosen.

3. A list of available database identifiers and instance numbers are displayed:

   ```
   Instances in this Workload Repository schema
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
      DB Id     Inst Num DB Name      Instance     Host
   ----------- -------- ------------ ------------ ------------
    3309173529        1 MAIN         main         examp1690
    3309173529        1 TINT251      tint251      samp251
   ```

   Enter the values for the database identifier (`dbid`) and instance number (`inst_num`) for the first time period:

   ```
   Enter value for dbid: 3309173529
   Using 3309173529 for Database Id for the first pair of snapshots
   Enter value for inst_num: 1
   Using 1 for Instance Number for the first pair of snapshots
   ```

4. Specify the number of days for which you want to list snapshot IDs in the first time period.

   ```
   Enter value for num_days: 2
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the last 2 days are displayed.

5. Specify a beginning and ending snapshot ID for the first time period:

   ```
   Enter value for begin_snap: 102
   Enter value for end_snap: 103
   ```

   In this example, the snapshot with a snapshot ID of 102 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 103 is selected as the ending snapshot for the first time period.

6. Enter the values for the database identifier (`dbid`) and instance number (`inst_num`) for the second time period:

   ```
   Enter value for dbid2: 3309173529
   Using 3309173529 for Database Id for the second pair of snapshots
   Enter value for inst_num2: 1
   Using 1 for Instance Number for the second pair of snapshots
   ```

7. Specify the number of days for which you want to list snapshot IDs in the second time period.

```
Enter value for num_days2: 1
```

A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the previous day are displayed.

8. Specify a beginning and ending snapshot ID for the second time period:

   ```
   Enter value for begin_snap2: 126
   Enter value for end_snap2: 127
   ```

   In this example, the snapshot with a snapshot ID of 126 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 127 is selected as the ending snapshot for the second time period.

9. Enter a report name, or accept the default report name:

   ```
   Enter value for report_name:
   Using the report name awrdiff_1_102_1_126.txt
   ```

   In this example, the default name is accepted and an AWR report named `awrdiff_1_102_126` is generated on the database instance with a database ID value of `3309173529`.

Generating an Oracle RAC AWR Compare Periods Report for the Local Database

The `awrgdrpt.sql` SQL script generates an HTML or text report that compares detailed performance attributes and configuration settings between two selected time periods using the current database identifier and all available database instances in an Oracle Real Application Clusters (Oracle RAC) environment.

> **Note:**
>
> In an Oracle RAC environment, generate an HTML report (instead of a text report) because it is much easier to read.

To generate an AWR Compare Periods report for Oracle RAC on the local database instance using the command-line interface:

1. At the SQL prompt, enter:

   ```
   @$ORACLE_HOME/rdbms/admin/awrgdrpt.sql
   ```

2. Specify whether you want an HTML or a text report:

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

   In this example, an HTML report is chosen.

3. Specify the number of days for which you want to list snapshot IDs in the first time period.

   ```
   Enter value for num_days: 2
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the last 2 days are displayed.

4. Specify a beginning and ending snapshot ID for the first time period:

   ```
   Enter value for begin_snap: 102
   Enter value for end_snap: 103
   ```

In this example, the snapshot with a snapshot ID of 102 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 103 is selected as the ending snapshot for the first time period.

5. Specify the number of days for which you want to list snapshot IDs in the second time period.

   ```
   Enter value for num_days2: 1
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the previous day are displayed.

6. Specify a beginning and ending snapshot ID for the second time period:

   ```
   Enter value for begin_snap2: 126
   Enter value for end_snap2: 127
   ```

   In this example, the snapshot with a snapshot ID of 126 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 127 is selected as the ending snapshot for the second time period.

7. Enter a report name, or accept the default report name:

   ```
   Enter value for report_name:
   Using the report name awrracdiff_1st_1_2nd_1.html
   ```

   In this example, the default name is accepted and an AWR report named `awrrac_1st_1_2nd_1.html` is generated.

Generating an Oracle RAC AWR Compare Periods Report for a Specific Database

The `awrgdrpi.sql` SQL script generates an HTML or text report that compares detailed performance attributes and configuration settings between two selected time periods using specific databases and instances in an Oracle RAC environment. This script enables you to specify database identifiers and a comma-delimited list of database instances for which AWR Compare Periods report will be generated.

> **Note:**
>
> In an Oracle RAC environment, you should always generate an HTML report (instead of a text report) because they are much easier to read.

To generate an AWR Compare Periods report for Oracle RAC on a specific database using the command-line interface:

1. At the SQL prompt, enter:

   ```
   @$ORACLE_HOME/rdbms/admin/awrgdrpi.sql
   ```

2. Specify whether you want an HTML or a text report:

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

   In this example, an HTML report is chosen.

3. A list of available database identifiers and instance numbers are displayed:

   ```
   Instances in this Workload Repository schema
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
      DB Id     Inst Num DB Name      Instance     Host
   ```

```
----------- -------- ------------ ------------ ------------
 3309173529        1 MAIN         main         examp1690
 3309173529        1 TINT251      tint251      samp251
 3309173529        2 TINT251      tint252      samp252
 3309173529        3 TINT251      tint253      samp253
 3309173529        4 TINT251      tint254      samp254
```

Enter the values for the database identifier (`dbid)` and instance number (`instance_numbers_or_all)` for the first time period:

```
Enter value for dbid: 3309173529
Using 3309173529 for Database Id for the first pair of snapshots
Enter value for inst_num: 1,2
Using instances 1 for the first pair of snapshots
```

4. Specify the number of days for which you want to list snapshot IDs in the first time period.

   ```
   Enter value for num_days: 2
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the last 2 days are displayed.

5. Specify a beginning and ending snapshot ID for the first time period:

   ```
   Enter value for begin_snap: 102
   Enter value for end_snap: 103
   ```

   In this example, the snapshot with a snapshot ID of 102 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 103 is selected as the ending snapshot for the first time period.

6. A list of available database identifiers and instance numbers are displayed:

   ```
   Instances in this Workload Repository schema
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
      DB Id    Inst Num DB Name      Instance     Host
   ----------- -------- ------------ ------------ ------------
    3309173529        1 MAIN         main         examp1690
    3309173529        1 TINT251      tint251      samp251
    3309173529        2 TINT251      tint252      samp252
    3309173529        3 TINT251      tint253      samp253
    3309173529        4 TINT251      tint254      samp254
   INSTNUM1
   -----------------------------------------------------
   1,2
   ```

   Enter the values for the database identifier (`dbid2)` and instance numbers (`instance_numbers_or_all2`) for the second time period:

   ```
   Enter value for dbid2: 3309173529
   Using 3309173529 for Database Id for the second pair of snapshots
   Enter value for instance_numbers_or_all2: 3,4
   ```

7. Specify the number of days for which you want to list snapshot IDs in the second time period.

   ```
   Enter value for num_days2: 1
   ```

   A list of existing snapshots for the specified time range is displayed. In this example, snapshots captured in the previous day are displayed.

8. Specify a beginning and ending snapshot ID for the second time period:

   ```
   Enter value for begin_snap2: 126
   Enter value for end_snap2: 127
   ```

In this example, the snapshot with a snapshot ID of 126 is selected as the beginning snapshot, and the snapshot with a snapshot ID of 127 is selected as the ending snapshot for the second time period.

9. Enter a report name, or accept the default report name:

   ```
   Enter value for report_name:
   Using the report name awrracdiff_1st_1_2nd_1.html
   ```

   In this example, the default name is accepted and an AWR report named `awrrac_1st_1_2nd_1.html` is generated.

Interpreting Automatic Workload Repository Compare Periods Reports

After generating an AWR Compare Periods report for the time periods you want to compare, review its contents to identify possible causes of performance degradation over time.

The content of the AWR Compare Periods report is divided into the following sections:

- Summary of the AWR Compare Periods Report
- Details of the AWR Compare Periods Report
- Supplemental Information in the AWR Compare Periods Report

Summary of the AWR Compare Periods Report

The report summary is at the beginning of the AWR Compare Periods report, and summarizes information about the snapshot sets and workloads used in the report.

The report summary contains the following sections:

- Snapshot Sets
- Host Configuration Comparison
- System Configuration Comparison
- Load Profile
- Top 5 Timed Events

Snapshot Sets

The Snapshot Sets section displays information about the snapshot sets used for this report, such as instance, host, and snapshot information.

Host Configuration Comparison

The Host Configuration Comparison section compares the host configurations used in the two snapshot sets. For example, the report compares physical memory and number of CPUs. Any differences in the configurations are quantified as percentages differed in the %Diff column.

System Configuration Comparison

The System Configuration Comparison section compares the database configurations used in the two snapshot sets. For example, the report compares the System Global Area (SGA) and log buffer sizes. Any differences in the configurations are quantified as percentages differed in the %Diff column.

Load Profile

The Load Profile section compares the workloads used in the two snapshot sets. Any differences in the workloads are quantified as percentages differed in the %Diff column.

Top 5 Timed Events

The Top 5 Timed Events section displays the five timed events or operations that consumed the highest percentage of total database time (DB time) in each of the snapshot sets.

Details of the AWR Compare Periods Report

The details section follows the report summary of the AWR Compare Periods report, and provides extensive information about the snapshot sets and workloads used in the report.

The report details contains the following sections:

- Time Model Statistics
- Operating System Statistics
- Wait Events
- Service Statistics
- SQL Statistics
- Instance Activity Statistics
- I/O Statistics
- Advisory Statistics
- Wait Statistics
- Undo Segment Summary
- Latch Statistics
- Segment Statistics
- In-Memory Segment Statistics
- Dictionary Cache Statistics
- Library Cache Statistics
- Memory Statistics
- Streams Statistics

Time Model Statistics

The Time Model Statistics section compares time model statistics in the two snapshot sets. The time model statistics are ordered based on the difference in total DB time spent on a particular type of operation between the two snapshot sets, and are listed in descending order. Time model statistics at the top of this section have the greatest differential between the two snapshot sets, and the related operations may be possible causes for performance degradation over time.

See Also:

"Time Model Statistics" for information about time model statistics

Operating System Statistics

The Operating System Statistics section compares operating system statistics in the two snapshot sets. This section provides an overall state of the operating system during each of the two periods being compared.

Wait Events

The Wait Events section compares the wait events in the two snapshot sets.

The first section lists the classes of wait events, including user I/O and system I/O. The classes are listed in descending order by absolute value of the % of DB time column.

The second section lists the wait events. The wait events are ordered based on the difference in total DB time spent on the wait event between the two snapshot sets, and are listed in descending order. Wait events at the top of this section have the greatest differential between the two snapshot sets, and may be possible causes for performance degradation over time.

See Also:

"Wait Events Statistics" for information about wait events and wait classes

Service Statistics

The Service Statistics section compares services in the two snapshot sets. The services are ordered based on the difference in total DB time spent on a particular service between the two snapshot sets, and are listed in descending order.

SQL Statistics

The SQL Statistics section compares the top SQL statements in the two snapshot sets. The SQL statements are ordered based on different comparison methods, but in all cases, the top ten SQL statements with the greatest differential between the two snapshot sets are shown.

The SQL statements shown in this section may be possible causes for performance degradation over time, and are ordered based on the following categories:

- Top 10 SQL Comparison by Execution Time
- Top 10 SQL Comparison by CPU Time
- Top 10 SQL Comparison by Buffer Gets
- Top 10 SQL Comparison by Physical Reads
- Top 10 SQL Comparison by Executions
- Top 10 SQL Comparison by Parse Calls
- Complete List of SQL Text

Top 10 SQL Comparison by Execution Time

SQL statements in this subsection are ordered based on the difference in total DB time spent processing the SQL statement between the two snapshot sets and are listed in descending order.

SQL statements shown in this subsection that consumed a high percentage of DB time in the one time period, but not in the other, are likely the high-load SQL statements that caused the performance degradation and should be investigated. Review the SQL statements in the Complete List of SQL Text subsection of the report and tune them, if necessary.

> **See Also:**
>
> *Oracle Database SQL Tuning Guide* for information about tuning SQL statements

Top 10 SQL Comparison by CPU Time

SQL statements in this subsection are ordered based on the difference in CPU time spent processing the SQL statement between the two snapshot sets, and are listed in descending order.

Top 10 SQL Comparison by Buffer Gets

SQL statements in this subsection are ordered based on the difference in the number of total buffer cache reads or buffer gets made when processing the SQL statement between the two snapshot sets, and are listed in descending order.

Top 10 SQL Comparison by Physical Reads

SQL statements in this subsection are ordered based on the difference in the number of physical reads made when processing the SQL statement between the two snapshot sets, and are listed in descending order.

Top 10 SQL Comparison by Executions

SQL statements in this subsection are ordered based on the difference in the number of executions per second (based on DB time) when processing the SQL statement between the two snapshot sets, and are listed in descending order.

Top 10 SQL Comparison by Parse Calls

SQL statements in this subsection are ordered based on the difference in the number of total parses made when processing the SQL statement between the two snapshot sets, and are listed in descending order. Parsing is one stage in the processing of a SQL statement.

When an application issues a SQL statement, the application makes a parse call to Oracle Database. Making parse calls can greatly affect the performance of a database and should be minimized as much as possible.

> **See Also:**
>
> *Oracle Database Concepts* for information about parsing

Complete List of SQL Text

This subsection displays the SQL text of all SQL statements listed in the SQL Statistics section.

Instance Activity Statistics

The Instance Activity Statistics section compares the statistic values of instance activity between the two snapshot sets. For each statistic, the value of the statistic is shown along with the differentials measured by DB time, elapsed time, and per transaction.

The instance activity statistics are categorized into the following subsections:

- Key Instance Activity Statistics
- Other Instance Activity Statistics

Key Instance Activity Statistics

This subsection displays the difference in key instance activity statistic values between the two snapshot sets.

Other Instance Activity Statistics

This subsection displays the difference in instance activity for all other statistics between the two snapshot sets.

I/O Statistics

The I/O Statistics section compares the I/O operations performed on tablespaces and database files between the two snapshot sets. A drastic increase in I/O operations between the two snapshots may be the cause of performance degradation over time.

For each tablespace or database file, the difference in the number of reads, writes, and buffer cache waits (or buffer gets) are quantified as a percentage. The database files are ordered based on different comparison methods, but in all cases, the top 10 database files with the greatest differential between the two snapshot sets are shown.

The I/O statistics are divided into the following categories:

- Tablespace I/O Statistics
- Top 10 File Comparison by I/O
- Top 10 File Comparison by Read Time
- Top 10 File Comparison by Buffer Waits

Tablespace I/O Statistics

Tablespaces shown in this subsection are ordered by the difference in the number of normalized I/Os performed on the tablespace between the two snapshot sets, and are listed in descending order. Normalized I/Os are the sum of average reads and writes per second.

Top 10 File Comparison by I/O

Database files shown in this subsection are ordered by the difference in the number of normalized I/Os performed on the database file between the two snapshot sets, and are listed in descending order. Normalized I/Os are the sum of average reads and writes per second.

Top 10 File Comparison by Read Time

Database files shown in this subsection are ordered by the difference in the percentage of DB time spent reading data from the database file between the two snapshot sets, and are listed in descending order.

Top 10 File Comparison by Buffer Waits

Database files shown in this subsection are ordered by the difference in the number of buffer waits (waits caused during a free buffer lookup in the buffer cache) performed on the database file between the two snapshot sets, and are listed in descending order.

Advisory Statistics

The Advisory Statistics section compares program global area (PGA) memory statistics between the two snapshot sets, and is divided into the following categories:

- PGA Aggregate Summary
- PGA Aggregate Target Statistics

PGA Aggregate Summary

This subsection compares the PGA cache hit ratio between the two snapshot sets.

PGA Aggregate Target Statistics

This subsection compares the key statistics related to the automatic PGA memory management between the two snapshot sets.

Wait Statistics

The Wait Statistics section compares statistics for buffer waits and enqueues between the two snapshot sets.

The wait statistics are divided into the following categories:

- Buffer Wait Statistics
- Enqueue Activity

Buffer Wait Statistics

This subsection compares buffer waits between the two snapshot sets. Buffer waits happen during a free buffer lookup in the buffer cache.

Enqueue Activity

This subsection compares enqueue activities between the two snapshot sets. Enqueues are shared memory structures (or locks) that serialize access to database resources and can be associated with a session or transaction.

> **See Also:**
>
> *Oracle Database Reference* for information about enqueues

Undo Segment Summary

The Undo Segment Summary section compares the use of undo segments in the two periods. The chart compares the number of undo blocks in the two periods, the number of transactions that use those blocks, and the maximum length of queries. The STO/OOS column indicates the number of snapshot too old and out of space counts.

Latch Statistics

The Latch Statistics section compares the number of total sleeps for latches between the two snapshot sets in descending order.

Latches are simple, low-level serialization mechanisms to protect shared data structures in the SGA. For example, latches protect the list of users currently accessing the database and the data structures describing the blocks in the buffer cache. A server or background process acquires a latch for a very short time while manipulating or looking up one of these structures. The implementation of latches is operating system dependent, particularly in regard to whether and how long a process will wait for a latch.

Segment Statistics

The Segment Statistics section compares segments, or database objects (such as tables and indexes), between the two snapshot sets. The segments are ordered based on different comparison methods, but in all cases the top five segments with the greatest differential between the two snapshot sets are shown.

The segments shown in this may be the causes of performance degradation over time, and are ordered based on the following categories:

- Top 5 Segments Comparison by Logical Reads
- Top 5 Segments Comparison by Physical Reads
- Top 5 Segments Comparison by Row Lock Waits
- Top 5 Segments Comparison by ITL Waits
- Top 5 Segments Comparison by Buffer Busy Waits

Top 5 Segments Comparison by Logical Reads

Segments shown in this subsection are ordered based on the difference in the number of logical reads (total number of reads from disk or memory) performed on the segment between the two snapshot sets, and are listed in descending order.

If an extremely high percentage of logical reads are made on a database object, then the associated SQL statements should be investigated to determine if data access to the database object need to be tuned using an index or a materialized view.

> **See Also:**
>
> *Oracle Database SQL Tuning Guide* for information about optimizing data access paths

Top 5 Segments Comparison by Physical Reads

Segments shown in this subsection are ordered based on the difference in the number of physical reads (such as disk reads) performed on the segment between the two snapshot sets, and are listed in descending order.

Top 5 Segments Comparison by Row Lock Waits

Segments shown in this subsection are ordered based on the difference in the number of waits on row locks for the segment between the two snapshot sets, and are listed in descending order.

Row-level locks are primarily used to prevent two transactions from modifying the same row. When a transaction needs to modify a row, a row lock is acquired.

> **See Also:**
>
> *Oracle Database Concepts* for information about row locks

Top 5 Segments Comparison by ITL Waits
Segments shown in this subsection are ordered based on the difference in the number of interested transaction list (ITL) waits for the segment between the two snapshot sets, and are listed in descending order.

Top 5 Segments Comparison by Buffer Busy Waits
Segments shown in this subsection are ordered based on the difference in the number of buffer busy waits for the segment between the two snapshot sets, and are listed in descending order.

In-Memory Segment Statistics

The In-Memory Segment Statistics section compares in-memory segment statistics between the two snapshot sets and lists the top in-memory segments based on number of scans, database block changes, populate CU activities, and repopulate CU activities. These statistics provide an insight into how in-memory segments are utilized by user workload. The In-Memory Segment Statistics section is displayed in AWR Compare Periods report only if Oracle Database has in-memory activity.

Dictionary Cache Statistics

The Dictionary Cache Statistics section compares the number of get requests performed on the dictionary cache between the two snapshot sets in descending order. The difference is measured by the number of get requests per second of both total DB time and elapsed time.

The dictionary cache is a part of the SGA that stores information about the database, its structures, and its users. The dictionary cache also stores descriptive information (or metadata) about schema objects, which is accessed by Oracle Database during the parsing of SQL statements.

> **See Also:**
>
> "Data Dictionary Cache Concepts" for information about the dictionary cache

Library Cache Statistics

The Library Cache Statistics section compares the number of get requests performed on the library cache between the two snapshot sets in descending order. The difference is measured by the number of get requests per second of both total DB time and elapsed time.

The library cache is a part of the SGA that stores table information, object definitions, SQL statements, and PL/SQL programs.

> **See Also:**
>
> "Library Cache Concepts" for information about the library cache

Memory Statistics

The Memory Statistics section compares process and SGA memory statistics between the two snapshot sets, and is divided into the following categories:

- Process Memory Summary
- SGA Memory Summary
- SGA Breakdown Difference

Process Memory Summary

This subsection summarizes the memory use of processes in the two time periods. The process categories include SQL, PL/SQL, and other.

SGA Memory Summary

This subsection summarizes the SGA memory configurations for the two snapshot sets.

SGA Breakdown Difference

This subsection compares SGA memory usage for each of its subcomponents between the two snapshot sets. The difference is measured based on the percentage changed in the beginning and ending values of memory usage between the two snapshot sets.

Advanced Queuing Statistics

The Advanced Queuing Statistics section compares CPU time, I/O time, and other statistics.

Supplemental Information in the AWR Compare Periods Report

The supplemental information is at the end of the AWR Compare Periods report, and provides information that is useful but not essential about the snapshot sets and workloads used in the report.

The supplemental information contains the following sections:

- init.ora Parameters
- Complete List of SQL Text

init.ora Parameters

The init.ora Parameters section lists all the initialization parameter values for the first snapshot set. Any changes in the values of the initialization parameters between the two snapshot sets are listed for the second snapshot set with the changed value shown.

Complete List of SQL Text

The Complete List of SQL Text section lists each statement contained in the workloads by SQL ID and shows the text of the SQL statement.