42

DBMS_AW_STATS

DBMS_AW_STATS contains subprograms for managing optimizer statistics for cubes and dimensions. Generating the statistics does not have a significant performance cost.

> **See Also:**
>
> *Oracle OLAP User's Guide* regarding use of the OLAP option to support business intelligence and analytical applications

This chapter contains the following topics:

- Using DBMS_AW_STATS
- Summary of DBMS_AW_STATS Subprograms

Using DBMS_AW_STATS

Cubes and dimensions are first class data objects that support multidimensional analytics. They are stored in a container called an analytic workspace. Multidimensional objects and analytics are available with the OLAP option to Oracle Database.

Optimizer statistics are used to create execution plans for queries that join two cube views or join a cube view to a table or a view of a table. They are also used for query rewrite to cube materialized views. You need to generate the statistics only for these types of queries.

Queries against a single cube do not use optimizer statistics. These queries are automatically optimized within the analytic workspace.

Summary of DBMS_AW_STATS Subprograms

DBMS_AW_STATS uses the ANALYZE and CLEAR procedures.

Table 42-1 DBMS_AW_STATS Package Subprograms

Subprogram	Description
ANALYZE Procedure	Generates optimizer statistics on cubes and cube dimensions.
CLEAR Procedure	Clears optimizer statistics from cubes and cube dimensions.

ANALYZE Procedure

This procedure generates optimizer statistics on a cube or a cube dimension.

These statistics are used to generate some execution plans, as described in "Using DBMS_AW_STATS".

For a cube, the statistics are for all of the measures and calculated measures associated with the cube. These statistics include:

- The average length of data values
- The length of the largest data value
- The minimum value
- The number of distinct values
- The number of null values

For a dimension, the statistics are for the dimension and its attributes, levels, and hierarchies. These statistics include:

- The average length of a value
- The length of the largest value
- The minimum value
- The maximum value

Syntax

```
DBMS_AW_STATS.ANALYZE
     (inname       IN VARCHAR2);
```

Parameters

Table 42-2 ANALYZE Procedure Parameters

Parameter	Description
`inname`	The qualified name of a cube or a dimension. For a cube, the format of a qualified name is *`owner.cube_name`*. For a dimension, the format is *`owner.dimension_name`*.

Usage Notes

Always analyze the dimensions first, then the cube.

After analyzing a dimension, analyze all cubes that use that dimension.

Example

This sample script generates optimizer statistics on `UNITS_CUBE` and its dimensions.

```
BEGIN
     DBMS_AW_STATS.ANALYZE('time');
     DBMS_AW_STATS.ANALYZE('customer');
     DBMS_AW_STATS.ANALYZE('product');
     DBMS_AW_STATS.ANALYZE('channel');
     DBMS_AW_STATS.ANALYZE('units_cube');
END;
/
```

The following statements create and display an execution plan for a`SELECT` statement that joins columns from `UNITS_CUBE_VIEW`, `CUSTOMER_PRIMARY_VIEW`, and the `ACCOUNTS` table:

```
EXPLAIN PLAN FOR SELECT
     cu.long_description customer,
                      a.city city,
                      a.zip_pc zip,
                      cu.level_name "LEVEL",
                      round(f.sales) sales
                 /* From dimension views and cube view */
                 FROM time_calendar_view t,
                      product_primary_view p,
                      customer_view cu,
                      channel_view ch,
                      units_cube_view f,
                      account a
                 /* Create level filters instead of GROUP BY */
                 WHERE t.long_description = '2004'
                     AND p.level_name ='TOTAL'
                     AND cu.customer_account_id like 'COMP%'
                     AND ch.level_name = 'TOTAL'
                 /* Join dimension views to cube view */
                     AND t.dim_key = f.TIME
                     AND p.dim_key = f.product
                     AND cu.dim_key = f.customer
                     AND ch.dim_key = f.channel
                     AND a.account_id = cu.customer_account_id
                 ORDER BY zip;

SQL>  SELECT plan_table_output FROM TABLE(dbms_xplan.display());

PLAN_TABLE_OUTPUT
-------------------------------------------------------------------------------------------------
Plan hash value: 3890178023

-----------------------------------------------------------------------------------------------
| Id  | Operation                        | Name       | Rows  | Bytes | Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT                 |            |     1 |    89 |     6  (34)| 00:00:01 |
|   1 |  SORT ORDER BY                   |            |     1 |    89 |     6  (34)| 00:00:01 |
|*  2 |   HASH JOIN                      |            |     1 |    89 |     5  (20)| 00:00:01 |
|   3 |    JOINED CUBE SCAN PARTIAL OUTER|            |       |       |            |          |
|   4 |     CUBE ACCESS                  | UNITS_CUBE |       |       |            |          |
|   5 |     CUBE ACCESS                  | CHANNEL    |       |       |            |          |
|   6 |     CUBE ACCESS                  | CUSTOMER   |       |       |            |          |
|   7 |     CUBE ACCESS                  | PRODUCT    |       |       |            |          |
|*  8 |     CUBE ACCESS                  | TIME       |     1 |    55 |     2   (0)| 00:00:01 |
|*  9 |    TABLE ACCESS FULL             | ACCOUNT    |     3 |   102 |     2   (0)| 00:00:01 |
-----------------------------------------------------------------------------------------------

Predicate Information (identified by operation id):
---------------------------------------------------

   2 - access("A"."ACCOUNT_ID"=SYS_OP_ATG(VALUE(KOKBF$),39,40,2))
   8 - filter(SYS_OP_ATG(VALUE(KOKBF$),16,17,2)='2004' AND
              SYS_OP_ATG(VALUE(KOKBF$),39,40,2) LIKE 'COMP%' AND
              SYS_OP_ATG(VALUE(KOKBF$),47,48,2)='TOTAL' AND
              SYS_OP_ATG(VALUE(KOKBF$),25,26,2)='TOTAL')
   9 - filter("A"."ACCOUNT_ID" LIKE 'COMP%')

Note
-----
   - dynamic statistics used for this statement

30 rows selected.
```

CLEAR Procedure

This procedure clears the statistics generated by the ANALYZE procedure.

Syntax

```
DBMS_AW_STATS.CLEAR (
          inname        IN  VARCHAR2;
```

Parameters

Table 42-3   CLEAR Procedure Parameters

Parameter	Description
`inname`	The qualified name of a cube or a dimension. For a cube, the format of a qualified name is *`owner.cube_name`*. For a dimension, the format is *`owner.dimension_name`*.

Examples

The following scripts clears the statistics from `UNITS_CUBE` and its dimensions.

```
BEGIN
     DBMS_AW_STATS.clear('units_cube');
     DBMS_AW_STATS.clear('time');
     DBMS_AW_STATS.clear('customer');
     DBMS_AW_STATS.clear('product');
     DBMS_AW_STATS.clear('channel');
END;
/
```