

実行した SQL 文に対する実行統計と実行計画を調査する方法

- ・ライブラリ・キャッシュに残っている実行統計と実行計画
- ・AWRスナップショットに記録されている実行統計と実行計画

ライブラリ・キャッシュに残っている実行統計と実行計画の調査

【実行統計】

```
SET LONG 2000000000
SELECT SQL_ID , ELAPSED_TIME , CPU_TIME , EXECUTIONS ,
       BUFFER_GETS , DISK_READS , PLAN_HASH_VALUE ,
       SQL_FULLTEXT
FROM V$SQL
WHERE SQL_TEXT LIKE '%<検索 文字列>%' ;
```

列 名	説 明
ELAPSED_TIME	経過時間 (CPU 時間+待機時間) (マイクロ秒)
CPU_TIME	CPU 時間 (マイクロ秒)
EXECUTIONS	実行回数
BUFFER_GETS	バッファ取得数
DISK_READS	ディスク読取り数
PLAN_HASH_VALUE	SQL プランの数値表現値

※ 出力する値は、合計値です

SQL 1 実行あたりの平均値を求める場合には、実行回数で除算してください

【実行計画】

```
SET LONG 2000000000
SELECT *
FROM table( DBMS_XPLAN.DISPLAY_CURSOR
            ( sql_id => '<SQL_ID>', format => '<出力レベル>' ) ) ;
```

指 定 値	説 明
BASIC	操作 ID、操作名、操作オプションのみの表示
TYPICAL	操作 ID、操作名、操作オプション、行数、バイト数、オペティマイザ・コストが表示 プルーニング情報、パラレル情報、述語情報は、適用可能時のみ表示
SERIAL	TYPICAL からパラレル情報を除いた情報が表示される
ALL	TYPICAL で表示される項目に加えて、PROJECTION、ALIAS、REMOTE SQL の情報が表示
ADVANCED	TYPICAL で表示される項目に加えて、クエリブロック、アウトライン列情報の情報が表示

AWR スナップショットに記録されている実行統計と実行計画の調査

【実行統計】

```
col BEGIN_INTERVAL_TIME format a22
col CPU_TIME_DELTA format 999999999999999999
SET LONG 2000000000

SELECT ss.BEGIN_INTERVAL_TIME , st.SQL_ID ,
       st.ELAPSED_TIME_TOTAL , st.ELAPSED_TIME_DELTA ,
       st.CPU_TIME_TOTAL , st.CPU_TIME_DELTA ,
       st.EXECUTIONS_TOTAL , st.EXECUTIONS_DELTA ,
       st.BUFFER_GETS_TOTAL , st.BUFFER_GETS_DELTA ,
       st.DISK_READS_TOTAL , st.DISK_READS_DELTA ,
       st.PLAN_HASH_VALUE
FROM DBA_HIST_SNAPSHOT ss , DBA_HIST_SQLSTAT st
WHERE ss.DBID = st.DBID
      AND ss.INSTANCE_NUMBER = st.INSTANCE_NUMBER
      AND ss.SNAP_ID = st.SNAP_ID ;
```

列名	説明
ELAPSED_TIME_TOTAL	経過時間（CPU 時間＋待機時間）（マイクロ秒）の累積値
ELAPSED_TIME_DELTA	ライブラリ・キャッシュに入れられた後からの経過時間（CPU 時間＋待機時間）の増加分合計
CPU_TIME_TOTAL	CPU 時間（マイクロ秒）の累積値
CPU_TIME_DELTA	ライブラリ・キャッシュに入れられた後からの CPU 時間（マイクロ秒）の増加分合計
EXECUTION_TOTAL	実行回数の累積値
EXECUTION_DELTA	ライブラリ・キャッシュに入れられた後からの実行回数の増加分合計
BUFFER_GETS_TOTAL	バッファ取得数の累積値
BUFFER_GETS_DELTA	ライブラリ・キャッシュに入れられた後からのバッファ取得数の増加分合計
DISK_READS_TOTAL	ディスク読取り数の累積値
DISK_READS_DELTA	ライブラリ・キャッシュに入れられた後からのディスク読取り数の増加分合計
PLAN_HASH_VALUE	SQL プランの数値表現値

※ 出力する値は、合計値です

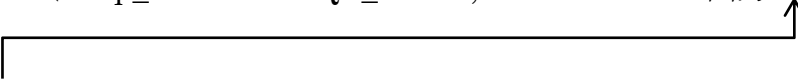
SQL 1 実行あたりの平均値を求める場合には、実行回数で除算してください

【実行計画】

```
SET LONG 2000000000
```

```
SELECT *
```

```
FROM table( DBMS_XPLAN.DISPLAY_AWR  
            ( sql_id => '<SQL_ID >', format => '<出力レベル>' ));
```



指定値	説明
BASIC	操作 ID、操作名、操作オプションのみの表示
TYPICAL	操作 ID、操作名、操作オプション、行数、バイト数、オペティマイザ・コストが表示 プルーニング情報、パラレル情報は、適用可能時のみ表示
SERIAL	TYPICAL からパラレル情報を除いた情報が表示される
ALL	TYPICAL で表示される項目に加えて、PROJECTION、ALIAS、REMOTE SQL の情報が表示
ADVANCED	TYPICAL で表示される項目に加えて、クエリブロック、アウトライン列情報の情報が表示

実行された SQL 文が過去と同じ実行計画が使われていたか確認する方法

V\$SQL ビュー、および DBA_HIST_SQLSTAT 表の中の PLAN_HASH_VALUE 列の値が、SQL の実行計画を表現した値である

この値が同一であれば、同じ実行計画が利用されたことになる

しかし、同一の SQL 文であっても異なる PLAN_HASH_VALUE 値である場合は、実行計画が異なっていたということになる

ライブラリ・キャッシュに残っている

1つの SQL_ID に、複数の PLAN_HASH_VALUE 値が存在する SQL 文

```
Col SQL_ID Format A15
break on SQL_ID on SQL_TEXT

Select  DISTINCT  OUT_FILE.SQL_ID ,
          OUT_FILE.PLAN_HASH_VALUE ,
          OUT_FILE.SQL_TEXT

From (

Select  DIS.SQL_ID ,
          Count(DIS.PLAN_HASH_VALUE)  DIS_COUNT
      From (
          Select distinct SQL_ID ,
                          PLAN_HASH_VALUE
          From  V$SQL
        ) DIS
      Group By  SQL_ID
      Having  Count(*) > 1
    ) IN_FILE
,  V$SQL  OUT_FILE
WHERE  IN_FILE.SQL_ID = OUT_FILE.SQL_ID
Order By  OUT_FILE.SQL_ID , OUT_FILE.PLAN_HASH_VALUE;
```

AWR スナップショットに残っている

1つの SQL_ID に、複数の PLAN_HASH_VALUE 値が存在する SQL 文

```
Col SQL_ID Format A15
```

```
break on SQL_ID on SQL_TEXT
```

```
Select  OUT_FILE.SQL_ID , OUT_FILE.SQL_PLAN_HASH_VALUE ,  
        SQL_TEXT
```

```
From (
```

```
Select  DIS.SQL_ID ,  
        Count(DIS.SQL_PLAN_HASH_VALUE) DIS_COUNT
```

```
From (
```

```
Select  distinct SQL_ID ,  
        SQL_PLAN_HASH_VALUE  
From    DBA_HIST_ACTIVE_SESS_HISTORY
```

```
) DIS
```

```
Group By SQL_ID
```

```
Having  Count(*) > 1
```

```
) IN_FILE
```

```
, DBA_HIST_ACTIVE_SESS_HISTORY OUT_FILE
```

```
, DBA_HIST_SQLTEXT
```

```
WHERE IN_FILE.SQL_ID = OUT_FILE.SQL_ID
```

```
And  OUT_FILE.SQL_ID = DBA_HIST_SQLTEXT.SQL_ID
```

```
Order By OUT_FILE.SQL_ID , OUT_FILE.SQL_PLAN_HASH_VALUE;
```

【実行統計】の表示実行例)

ライブラリ・キャッシュに残っている【実行統計】

```
SET LONG 2000000000
SELECT SQL_ID , ELAPSED_TIME , CPU_TIME , EXECUTIONS ,
       BUFFER_GETS , DISK_READS , PLAN_HASH_VALUE ,
       SQL_FULLTEXT
FROM V$SQL
WHERE SQL_TEXT LIKE '%<検索 文字列>%' ;
```

SQL_ID	ELAPSED_TIME	CPU_TIME	BUFFER_GET	PLAN_HASH_VALUE	EXECUTIONS	DISK_READS
bvhnf4gvv1m8t	5006	0	1	10	0	903671040
2uc9d1b6mf11s	55429	46875	1	24	0	647829036

【実行計画】の実行例

```
SET LONG 2000000000
SELECT *
FROM table(DBMS_XPLAN.DISPLAY_CURSOR
           ( sql_id => 'bvhnf4gvv1m8t', format => 'TYPICAL' ));
```

PLAN_TABLE_OUTPUT

SQL_ID bvhnf4gvv1m8t

```
SELECT SQL_ID , ELAPSED_TIME , CPU_TIME , EXECUTIONS ,
       BUFFER_GETS , DISK_READS , PLAN_HASH_VALUE ,
       SQL_FULLTEXT FROM V$SQL WHERE SQL_TEXT LIKE '%<検索
文字列>%'
```

Plan hash value: 903671040

Id	Operation	Name	Row	Byte
0	SELECT STATEMENT			
* 1	FIXED TABLE FULL	X\$KGLCUR_CH	1	2616

Note

1 - filter(("KGLNAOBJ" IS NOT NULL AND INTERNAL_FUNCTION
("CON_ID") AND "KGLNAOBJ" LIKE '%<検索 文字列>%'
AND "INST_ID"=USERENV('INSTANCE')))

【ライブラリ・キャッシュの中の1つのSQL_IDに、複数のPLAN_HASH_VALUE値が存在するSQL文の確認】 の実行例

```
Col SQL_ID Format A15  
break on SQL_ID on SQL_TEXT  
  
Select DISTINCT OUT_FILE.SQL_ID ,  
                OUT_FILE.PLAN_HASH_VALUE ,  
                OUT_FILE.SQL_TEXT  
  
From (   
  
    Select DIS.SQL_ID ,  
           Count(DIS.PLAN_HASH_VALUE) DIS_COUNT  
    From (   
  
        Select distinct SQL_ID ,  
                        PLAN_HASH_VALUE  
        From V$SQL  
  
    ) DIS  
    Group By SQL_ID  
    Having Count(*) > 1  
  
    ) IN_FILE  
    , V$SQL OUT_FILE  
    WHERE IN_FILE.SQL_ID = OUT_FILE.SQL_ID  
Order By OUT_FILE.SQL_ID , OUT_FILE.PLAN_HASH_VALUE;
```

SQL_ID	PLAN_HASH_VALUE	SQL_TEXT
0fr8zhn4ymu3v	1231101765 3815847153	select intcol#,type,flags,lobcol
0h2255k026qr3	894252314 1557246855	delete /* QOSH:PURGE_OLD
1j6tnz8fcm4c3	675971078 1200386806	deleteselect count(*) from sys.

```
Select DISTINCT SQL_ID, PLAN_HASH_VALUE , SQL_TEXT
From V$SQL
Where SQL_ID = '0fr8zhn4ymu3v' ;
```

SQL_ID	PLAN_HASH_VALUE	SQL_TEXT
0fr8zhn4ymu3v	1231101765 3815847153	select intcol#,type,flags,lobcol

AWR スナップショットに記録されている

【実行統計】 の実行例

```
col BEGIN_INTERVAL_TIME format a22
col CPU_TIME_DELTA format 999999999999999999
SET LONG 2000000000

SELECT ss.BEGIN_INTERVAL_TIME , st.SQL_ID ,
       st.ELAPSED_TIME_TOTAL , st.ELAPSED_TIME_DELTA ,
       st.CPU_TIME_TOTAL , st.CPU_TIME_DELTA ,
       st.EXECUTIONS_TOTAL , st.EXECUTIONS_DELTA ,
       st.BUFFER_GETS_TOTAL , st.BUFFER_GETS_DELTA ,
       st.DISK_READS_TOTAL , st.DISK_READS_DELTA ,
       st.PLAN_HASH_VALUE
FROM DBA_HIST_SNAPSHOT ss , DBA_HIST_SQLSTAT st
WHERE ss.DBID = st.DBID
      AND ss.INSTANCE_NUMBER = st.INSTANCE_NUMBER
      AND ss.SNAP_ID = st.SNAP_ID ;
```

BEGIN_INTERVAL _TIME	SQL_ID	ELAPSED_TIME _TOTAL	ELAPSED_TIME _DELTA
19-02-14 13:41:03.706	gjaap3w3qbf8c	14806	11126
19-02-14 13:41:03.706	9nyqtbp7n0hr7	736573	736573
19-02-14 13:41:03.706	01d5n1nm17r2h	196267	196267
19-02-14 13:41:03.706	aaz3bayav0jwj	344539	58888
19-02-14 13:41:03.706	121ffmrc95v7g	2009407	84957

【実行計画】 の実行例

```
SET LONG 2000000000
SELECT *
FROM table( DBMS_XPLAN.DISPLAY_AWR
            ( sql_id => '2sxqgx5hx76qr', format => 'TYPICAL' ) );
```

PLAN_TABLE_OUTPUT

SQL_ID 2sxqgx5hx76qr

```
select /*+ rule */ bucket, endpoint, col#, epvalue, epvalue_raw,
ep_repeat_count, endpoint_enc from histgrm$ where obj#=1 and
intcol#=2 and row#=3 order by bucket
```

Plan hash value: 3312420081

Id	Operation	Name	Cost
0	SELECT STATEMENT		
1	SORT ORDER BY		0
2	TABLE ACCESS CLUSTER	HISTGRM\$	
3	INDEX UNIQUE SCAN	I_OBJ#_INTCOL#	

Note

- cpu costing is off (consider enabling it)

【AWR スナップショットの中の1つの SQL_ID に、複数の PLAN_HASH_VALUE 値が存在する SQL 文の確認】 の実行例

```
Col SQL_ID Format A15
break on SQL_ID on SQL_TEXT

Select  OUT_FILE.SQL_ID , OUT_FILE.SQL_PLAN_HASH_VALUE ,
        SQL_TEXT
From (
    Select  DIS.SQL_ID ,
            Count(DIS.SQL_PLAN_HASH_VALUE) DIS_COUNT
    From (
        Select distinct SQL_ID ,
                        SQL_PLAN_HASH_VALUE
        From  DBA_HIST_ACTIVE_SESS_HISTORY
    ) DIS
    Group By SQL_ID
    Having  Count(*) > 1
) IN_FILE
,  DBA_HIST_ACTIVE_SESS_HISTORY OUT_FILE
,  DBA_HIST_SQLTEXT
WHERE IN_FILE.SQL_ID = OUT_FILE.SQL_ID
      And OUT_FILE.SQL_ID = DBA_HIST_SQLTEXT.SQL_ID
Order By OUT_FILE.SQL_ID , OUT_FILE.SQL_PLAN_HASH_VALUE;
```

SQL_ID	PLAN_HASH_VALUE	SQL_TEXT
0fr8zhn4ymu3v	1231101765 3815847153	select intcol#,type,flags,lobcol
0h2255k026qr3	894252314 1557246855	delete /* QOSH:PURGE_OLD
1j6tnz8fcm4c3	675971078 1200386806	deleteselect count(*) from sys.