

- 【 保存されている SQL 文と SQL 実行計画の識別 】
- 【 ライブラリ・キャッシュやAWRレポート、Statspack レポートからの SQL_ID、PLAN_HASH_VALUE、実行計画の調査方法 】
- 【 SPM が管理する「SQL 管理ベース」からの SQL_HANDLE、PLAN_NAME、実行計画の調査方法 】
- 【 SPM が管理する「SQL 管理ベース」の操作関数 】

項目名の出現元とその内容説明

- SQL_ID : ライブラリ・キャッシュやAWRレポート、Statspack レポートで SQL 文を識別する識別子の値
- PLAN_HASH_VALUE : 実行計画の識別子~~ハッシュ値~~
ライブラリ・キャッシュやAWRレポート、Statspack レポートでの SQL 文の実行計画を管理する識別キー
- SQL_HANDLE : SPM が管理する「SQL 管理ベース」の中で、SQL 文を識別するために付与された識別子
- PLAN_NAME : SPM が管理する「SQL 管理ベース」の中で、実行計画を識別するために付与された識別子
- SIGNATURE : SPM が管理する「SQL 管理ベース」の中で、対象の SQL 文から作成されたハッシュ値
- PLAN_ID : SPM が管理する「SQL 管理ベース」の中で、実行計画を基に作成したハッシュ値

項目名が示す内容

存在場所 / 識別対象	SQL 文	実行計画
ライブラリ・キャッシュ、AWR、Statspack レポート内での識別子	SQL_ID	PLAN_HASH_VALUE
SPM (SQL Plan Management) の「SQL 管理ベース」内の識別キー	SQL_HANDLE	PLAN_NAME
「SQL 管理ベース」内の値に対するハッシュ値	SIGNATURE	PLAN_ID
V\$SQL ビュー内で出現するときの「SQL 管理ベース」に関するハッシュ値	EXACT_MATCHING_SIGNATURE	SQL_PLAN_BASELINE

※ SIGNATURE と PLAN_ID の項目は、オプティマイザが実行計画を作成したときに SPM に登録してある SQL 文や実行計画と同一かを比較するために使われる
 これが、何かの動的ビューやディクショナリのキーとなっている項目ではない

保存されている SQL 文と SQL 実行計画の識別

【 ライブラリ・キャッシュやAWRレポート、Statspack レポートからの SQL_ID、PLAN_HASH_VALUE、実行計画の調査方法 】

SQL_ID : SQL 文のハッシュ値

PLAN_HASH_VALUE : 実行計画のハッシュ値

ライブラリ・キャッシュやAWRレポート、Statspack レポートでの SQL 文を管理する識別キー

操 作 関 数	利 用 目 的
A) → v\$sql ビュー	ライブラリ・キャッシュの中から、SQL テキスト文、SQL_ID、PLAN_HASH_VALUE、EXACT_MATCHING_SIGNATURE、SQL_PLAN_BASELINE を出力
B) → dba_hist_sqltext ディクショナリ	AWR 自動ワークリポジトリの中から、SQL テキスト文を検索条件として、SQL_ID 値の出力
C) → dba_hist_sqlstat ディクショナリ dba_hist_snapshot ディクショナリ	AWR 自動ワークリポジトリの中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値とスナップショットの ID を出力
D) → dba_sqlset_statements ディクショナリ	STS (SQL チューニングセット) の中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値と SQL テキスト文の出力
E) → dbms_xplan.display_cursor ファンクション	ライブラリ・キャッシュ or SPM の「SQL 計画ベースライン」の中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値と実行計画の内容 (SQL 計画ベースライン) の出力 ※ Note 部分に SQL plan baseline の記述があれば、SPM の SQL 計画ベースラインが使用されていることになる
F) → dbms_xplan.display_awr ファンクション	AWR 自動ワークリポジトリの中から、SQL_ID を検索条件として、実行計画の内容の出力
G) → dbms_xplan.display_sqlset ファンクション	STS (SQL チューニングセット) の中から、SQL Tuning Set Name と SQL_ID を検索条件として、実行計画の内容の出力

【 SPM が管理する「SQL 管理ベース」からの SQL_HANDLE、PLAN_NAME、実行計画の調査方法 】

SQL_HANDLE : SQL を識別するために付与された識別子

PLAN_NAME : 実行計画を識別するために付与された識別子

SPM が管理する SQL 管理ベースの識別キー

操 作 関 数	利 用 目 的
J) → dba_sql_plan_baselines ディクショナリ	「SQL 管理ベース」の SQL 計画ベースラインの中から、SQL テキスト文、SQL_HANDLE、PLAN_NAME (SQL 実行計画)、SIGNATURE を出力
K) → dbms_xplan.display_sql _plan_baseline ファンクション	「SQL 管理ベース」の SQL 計画ベースラインの中から、SQL_HANDLE を検索条件として、PLAN_NAME と SQL 実行計画内容の出力

SPM が管理する「SQL 管理ベース」の操作関数の一覧

SPM が管理する SQL 管理ベースの識別キー

操 作 関 数	操 作 内 容
N) -1→ dbms_spm. load_plans_from_cursor_ cache	SQL_ID と SQL_PLAN_BASELINE を用いた「SQL 管理ベース」への SQL 実行計画の新規登録
N) -2→	SQL_ID と PLAN_HASH_VALUE と SQL_HANDLE を用いた「SQL 管理ベース」への SQL 実行計画の追加登録
O) -1→ dbms_spm. drop_sql_plan_baseline	SQL_HANDLE と PLAN_NAME を用いた「SQL 管理ベース」への SQL 実行計画の削除
P) -1→ dbms_spm. evolve_sql_plan_baseline ファンクション	SQL 文に関連付けられている SQL 計画履歴を承認して、SQL 計画ベースラインへと変更する
P) -2→ dbms_spm. alter_sql_plan_baseline ファンクション	属性名/値の形式で SQL 文と関連付けられている 1 つまたはすべての計画の属性を変更する 登録されている実行計画の利用の可否を設定する

保存 SQL の属性 (DBA_SQL_PLAN_BASELINES デictionaryの列項目)

列名	説明
SQL_HANDLE	SQL を識別するために付与された識別子 (SPM 管理用) (同じ SQL の場合は、同一の SQL_HANDLE となる)
PLAN_NAME	実行計画を識別するために付与された識別子 (SPM 管理用)
CREATED	SQL 計画ベースラインへ実行計画が登録された日時
ACCEPTED	承認されているかを表すフラグ YES/NO
ENABLED	実行計画が利用可能かを表すフラグ YES/NO
SQL_TEXT	SQL 文のテキスト

(参考) Dictionaryや動的ビュー (V\$PLAN など) で SQL や実行計画の比較に使う列名は、**SQL_ID** (SQL の識別子) と **PLAN_HASH_VALUE** (実行計画のハッシュ値識別子) である

オプティマイザが内部で使う SQL 文の比較項目

(dba_sql_plan_baselines)

この2つの項目は、オプティマイザが実行計画を作成したときに SPM に登録してある SQL 文や実行計画と同一か比較するために使われる

※ これが、何かの動的ビューやDictionaryのキーとなっている項目ではない

SIGNATURE	対象の SQL 文から作成されたハッシュ値
PLAN_ID	対象の SQL 文から作成した実行計画を基に作成したハッシュ値

(v\$sql ビュー)

SQL_PLAN_BASELINE 列は、「SQL 管理ベース」に保存された SQL 実行計画の識別キー

v\$sql ビューで管理している項目 (列名)

SQL_ID	対象の SQL 文の識別子
PLAN_HASH_VALUE	対象の SQL 文から作成した実行計画の識別子

【 ライブラリ・キャッシュやAWRレポート、Statspack レポートからの SQL_ID、PLAN_HASH_VALUE、実行計画の調査方法 】

v\$sql ビュー

A)

ライブラリ・キャッシュの中から、SQL テキスト文、SQL_ID、PLAN_HASH_VALUE、EXACT_MATCHING_SIGNATURE、SQL_PLAN_BASELINE を出力

v\$sql ビューで出力する列名（条件指定できる列名）と出力項目内容

列名	内容説明
SQL_TEXT	SQL テキスト文
SQL_ID	ライブラリ・キャッシュの SQL 文識別子
PLAN_HASH_VALUE	ライブラリ・キャッシュの実行計画の識別子 ハッシュ値
EXACT_MATCHING_SIGNATURE	SPM の「SQL 管理ベース」での SQL 文のハッシュ値 (v\$sql に出現)
SQL_PLAN_BASELINE	SPM の「SQL 管理ベース」での実行計画の識別子 (v\$sql に出現)
CHILD_NUMBER	

使用例)

```
sql> select SQL_TEXT, SQL_ID, PLAN_HASH_VALUE,
           EXACT_MATCHING_SIGNATURE,
           SQL_PLAN_BASELINE, CHILD_NUMBER
       from v$sql
       where <列名> = <条件> ;
```

条件例) SQL_TEXT like 'select . . . from . . . %'

```
SQL_TEXT                SQL_ID                PLAN_HASH_VALUE
-----
select * from tab2  where . . . .    3m7d74pkw534z                2200541503

EXACT_MATCHING_SIGNATURE  SQL_PLAN_BASELINE  CHILD_NUMBER
-----
2667826512551042531      SQL_PLAN_2a1h62h9gkrg                0
```

dba_hist_sqltext デictionary

B)

AWR 自動ワークリポジトリの中から、SQL テキスト文を検索条件として、SQL_ID 値の出力

ビューで出力する列名（条件指定できる列名）と 出力項目内容

列名	内容説明
SQL_TEXT	SQL テキスト文
SQL_ID	ライブラリ・キャッシュの SQL 文識別子

使用例)

```
sql> select SQL_ID, SQL_TEXT from dba_hist_sqltext
       where <列名> = <条件> ;
```

条件例) SQL_TEXT like 'select . . . from . . . %'

```
SQL_ID          SQL_TEXT
-----
9zv8xn2891xf7  select * from tab2 where . . . .
```


dba_hist_sqlstat、dba_hist_snapshot ディクショナリ

C)

AWR 自動ワークリポジトリの中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値とスナップショットの ID の出力

ビューで出力する列名（条件指定できる列名）と出力項目内容

列名	内 容 説 明
SQL_ID	ライブラリ・キャッシュの SQL 文識別子
PLAN_HASH_VALUE	ライブラリ・キャッシュの実行計画の識別子
DBID	データベース ID
INSTANCE_NUMBER	D/B インスタンス番号
SNAP_ID	スナップショット ID
BEGIN_INTERVAL_TIME	スナップショットの開始日と時刻

```
sql> select  sql.SNAP_ID,
            to_char( snap.BEGIN_INTERVAL_TIME, 'yyyy/mm/dd hh24:mi:ss' ),
            sql.SQL_ID,  sql.PLAN_HASH_VALUE
from  dba_hist_sqlstat  sql ,  dba_hist_snapshot  snap
where  sql.dbid = snap.dbid
       and sql.instance_number = snap.instance_number
       and sql.snap_id = snap.snap_id
       and SQL_ID = '<SQL_ID 値>'
order by  sql.snap_id ;
```

SNAP_ID	to_char(snap.BEGIN_INT	SQL_ID	PLAN_HASH_VALUE
670	2014/08/02 15:09:58	9zv8xn2891xf7	615168685

dba_sqlset_statements ディクショナリ

D)

STS (SQL チューニングセット) の中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値と SQL テキスト文の出力

ビューで出力する列名 (条件指定できる列名) と 出力項目内容

列 名	内 容 説 明
SQL_ID	ライブラリ・キャッシュの SQL 文識別子
PLAN_HASH_VALUE	ライブラリ・キャッシュの実行計画の識別子
SQL_TEXT	SQL テキスト文
PARSING_SCHEMA_NAME	スキーマ名

```
sql> select sql_id, plan_hash_value, parsing_schema_name,
           substr( sql_text, 1, 100 ) sql_text
from dba_sqlset_statements
where sqlset_name = '<作成 SQL チューニングセット名>'
order by sql_id ;
```

```
SQL_ID          PLAN_HASH_VALUE  PARSING_SCHEMA_NAME
-----
8xc7nx19821fx6  515985324        KOZUE

SQL_TEXT
-----
Select e.emp , e.name , d.dname from emp e , dept d
Where . . . . .
```

dbms_xplan.display_cursor ファンクション

E)

ライブラリ・キャッシュ ~~or SPMの「SQL計画ベースライン」~~の中から、SQL_ID を検索条件として、PLAN_HASH_VALUE 値と実行計画の内容 (SQL 計画ベースライン) の出力

※ Note 部分に SQL plan baseline の記述があれば、SPM の「SQL 計画ベースライン」が使用されていることになる

```
select * from table(dbms_xplan.display_cursor('<SQL_ID 値>'));
```

PLAN_TABLE_OUT

SQL_ID : 3m7d74pkw543z, child number 0

SQL TEXT select * from tab2 where c2 > 999

Plan hash value : 1004337041

ID	OPERATION	Name	Rows
0	SELECT STATEMENT		
1	TABLE ACCESS BY INDEX ROWID	TAB2	10
2	INDEX RANGE SCAN	TAB2_ID_PK	10

Bytes	COST (%CPU)	TIME
	12 (100)	
2200	12 (0)	00:00:01
1	0 (0)	00:00:01

実行計画

Predicate Information (identified by operation id)

2 -access ("C2" = 999)

述語情報

NOTE

dynamic sampling used for this statement (level = 2)

SQL plan baseline SQL_PLAN_2a1h62h9gkrg371befc37 used for this statement

NOTE 部に

SQL plan baseline の記述があれば、SPM (SQL Plan Management) の SQL 計画ベースラインが使用されたこととなります

dbms_xplan.display_awr ファンクション

F)

AWR 自動ワークリポジトリの中から、SQL_ID を検索条件として、実行計画の内容の出力

```
select * from table( dbms_xplan.display_awr('<SQL_ID>') ) ;
```

ビューで出力する列名（条件指定できる列名）と出力項目内容

列名	内容説明
SQL_ID	ライブラリ・キャッシュの SQL 文識別子
SQL_TEXT	SQL テキスト文
PLAN_HASH_VALUE	ライブラリ・キャッシュの実行計画の識別子 ハッシュ値
	実行計画

PLAN_TABLE_OUT

SQL_ID : 3m7d74pkw543z, child number 0

SQL TEXT select * from tab2 where c2 > 999

Plan hash value : 1004337041

ID	OPERATION	Name	Rows
0	SELECT STATEMENT		
1	TABLE ACCESS BY INDEX ROWID	TAB2	10
2	INDEX RANGE SCAN	TAB2_ID_PK	10

Bytes	COST (%CPU)	TIME
	12 (100)	
2200	12 (0)	00:00:01
1	0 (0)	00:00:01

実行計画

Predicate Information (identified by operation id)

2 -access ("C2" = 999)

述語情報

NOTE

dynamic sampling used for this statement (level = 2)

SQL plan baseline SQL_PLAN_2a1h62h9gkrg371befc37 used for this statement

dbms_xplan.display_sqlset ファンクション

G)

STS (SQL チューニングセット) の中から、SQL Tuning Set Name と SQL_ID を検索条件として、実行計画の内容の出力

```
sql> select * from table( dbms_xplan.display_sqlset( '<SQL チューニング
セット名>', '<SQL_ID 値>' ) ) ;
```

PLAN_TABLE_OUTPUT

SQL Tuning Set Name : STS_SPM01
SQL Tuning Set Owner : SYS
SQL_ID : 3m7d74pkw543z, child number 0
SQL TEXT select * from tab2 where c2 > 999

Plan hash value : 1004337041

ID	OPERATION	Name	Rows
0	SELECT STATEMENT		
1	TABLE ACCESS BY INDEX ROWID	TAB2	10
2	INDEX RANGE SCAN	TAB2_ID_PK	10

Bytes	COST (%CPU)	TIME
	12 (100)	
2200	12 (0)	00:00:01
1	0 (0)	00:00:01

実行計画

Predicate Information (identified by operation id)

2-access ("C2" = 999)

述語情報

【 SPM が管理する「SQL 管理ベース」からの SQL_HANDLE、PLAN_NAME、
実行計画の調査方法 】

dba_sql_plan_baselines ディクショナリ

J)

「SQL 管理ベース」の SQL 計画ベースラインの中から、SQL テキスト文、
SQL_HANDLE、PLAN_NAME (SQL 実行計画)、SIGNATURE を出力

使用例)

```
SQL > select  SQL_HANDLE ,  PLAN_NAME ,  
              to_char( CREATED, 'YYYY/MM/DD HH24:MI:SS')  CREATED ,  
              ACCEPTED ,  ENABLED ,  SQL_TEXT,  SIGNATURE ,  
              REPRODUCED ,  to_char( LAST_EXECUTED ,  
              'YYYY/MM/DDHH 24:MI:SS')  LAST_EXECUTED  
from  dba_sql_plan_baselines  
where SQL_TEXT like 'select . . . from . . . %'  
order by  CREATED  ;
```

実行結果

SQL_HANDLE	PLAN_NAM
SQL_PLAN_2a1h62h9gkrg37befc37	SQL_2506061412f95de3

CREATED	ACCEPTED	ENABLED
2018/09/24 10:45:26	YES	YES

SQL_TEXT
select ename , empno , dectno from emp where

SIGNATURE	REPRODUCED	LAST_EXECUTED
2887569654589429895	YES	2018/09/24 10:45:26

出力項目内容

列名	内容説明
SQL_TEXT	SQL テキスト文
SQL_HANDLE	SPM の「SQL 管理ベース」での SQL 文の識別子
PLAN_NAME	SPM の「SQL 管理ベース」での実行計画の識別子
SIGNATURE	SPM の「SQL 管理ベース」での SQL 文のハッシュ値
PLAN_ID	SPM の「SQL 管理ベース」での実行計画のハッシュ値
CREATED	SQL 計画ベースラインへ実行計画が登録された日時
ACCEPTED	承認されているかを表すフラグ YES/NO
ENABLED	実行計画が利用可能かを表すフラグ YES/NO
REPRODUCED	「SQL 管理ベース」の SQL 実行計画が利用可能かを表すフラグ YES/NO
LAST_EXECUTED	SPM の「SQL 管理ベース」の SQL 実行計画が利用された日時（最終利用日ではない） 一度、日付が更新されると 6.5 日間は更新されない

【 dba_sql_plan_baselines と v\$sql の連結 SELECT 文】

```

select  vsql.SQL_TEXT ,  vsql.SQL_ID ,  vsql.PLAN_HASH_VALUE ,
        vsql.EXACT_MATCHING_SIGNATURE ,  spm.SIGNATURE ,
        vsql.SQL_PLAN_BASELINE ,  spm.PLAN_NAME ,
        vsql.CHILD_NUMBER ,  spm.SQL_HANDLE ,
        to_char( spm.CREATED , 'YYYY/MM/DD HH24:MI:SS')  CREATED ,
        spm.ACCEPTED ,  spm.ENABLED ,  spm.REPRODUCED ,
        to_char( spm.LAST_EXECUTED, 'YYYY/MM/DDHH24:MI:SS')
        LAST_EXECUTED
from    v$sql  vsql ,  dba_sql_plan_baselines  spm
where   vsql.SQL_TEXT like 'select .. from ...%'
        and vsql.EXACT_MATCHING_SIGNATURE = spm.SIGNATURE
        and vsql.SQL_PLAN_BASELINE = spm.PLAN_NAME
;

```

K) -1

「SQL 管理ベース」の SQL 計画ベースラインの中から、SQL_HANDLE を検索条件として、PLAN_NAME と SQL 実行計画内容の出力

SQL_HANDLE (前ページ参照) を使用して、SELECT を行う

```
sql> select * from table( dbms_xplan.display_sql_plan_baseline(
    'SQL_e18f9b7e02f01359'));
```

PLAN_TABLE_OUT

SQL handle : SQL_e18f9b7e02f01359

SQL TEXT select * from EMPLOYEES E, DEPARTMENT D
 where SARARY > 10000
 and E.DEPARTMENT_ID = D.DEPARTMENT_ID

Plan name : SQL_PLAN_f33wvgs1g04ut8847c07a

Enabled : YES Fixed: NO Accepted : YES Origin : MANUAL_LOAD

Plan hash value : 1004337041

ID	OPERATION	Name	Rows
0	SELECT STATEMENT		45
1	NESTED LOOPS		45
2	TABLE ACCESS FULL	EMPLOYEES	46
3	TABLE ACCESS BY INDEX ROWID	DEPARTMENT	1
4	INDEX UNIQUE SCAN	DEPT_ID_PK	1

Bytes	COST (%CPU)	TIME
1305	87 (0)	00:00:02
1305	87 (0)	00:00:02
322	41 (0)	00:00:01
22	1 (0)	00:00:01
1	0 (0)	00:00:01

実行計画

Predicate Information (identified by operation id)

2 – filter (“SARARY” > 10000)

4 – access (“E.”DEPARTMENT_ID” = “D.”DEPARTMENT_ID”)

述語情報

【 SPM が管理する SQL 管理ベースの操作関数 】

dbms_spm.load_plans_from_cursor_cache ファンクション

N) -1

- (1) SQL_ID と PLAN_HASH_VALUE を用いて、「SQL 管理ベース」への SQL 実行計画の**新規登録**

```
sql> set serveroutput on
```

```
sql> declare
```

```
    ret number;
```

```
begin
```

```
    ret := dbms_spm.load_plans_from_cursor_cache(
```

```
        sql_id => '<SQL_ID 値>',
```

```
        plan_hash_value => <PLAN_HASH_VALUE 値> );
```

```
    dbms_output.put_line('LOAD PLANS : ' || ret) ;
```

```
end;
```

```
/
```

「SQL 管理ベース」へ新規登録させる時のキー (sql_handle) 値は、ロジックにて算出される

実行結果

LOAD PLANS : 1 ※ 登録した SQL 計画ベースラインの数が戻り値として表示されます

N) -2

- (2) SQL_ID と PLAN_HASH_VALUE と SQL_HANDLE を用いた「SQL 管理ベース」への SQL 実行計画の**追加登録**

```
sql> set serveroutput on
```

```
sql> declare
```

```
    ret number;
```

```
begin
```

```
    ret := dbms_spm.load_plans_from_cursor_cache(
```

```
        sql_id => '<SQL_ID 値>',
```

```
        plan_hash_value => <PLAN_HASH_VALUE 値> ,
```

```
        sql_handle => '<SQL_HANDLE 値>');
```

```
    dbms_output.put_line('LOAD PLANS : ' || ret) ;
```

```
end;
```

```
/
```

登録させるライブラリ・キャッシュの実行計画を示すためのキー値

「SQL 管理ベース」へ追加登録させるためのキー値

実行結果

LOAD PLANS : 1 ※ 登録した SQL 計画ベースラインの数が戻り値として表示されます

dbms_spm.drop_sql_plan_baseline ファンクション

○) -1

SQL_HANDLE と PLAN_NAME を用いた「SQL 管理ベース」の SQL 実行計画の削除

```
sql> set serveroutput on
sql> declare
    ret number ;
begin
    ret := dbms_spm.drop_sql_plan_baseline(
        sql_handle => '<SQL_HANDLE 値>',
        plan_name => '<PLAN_NAME 値>');
    dbms_output.put_line(' DROP PLANS : ' || ret) ;
end;
/
```

実行結果

DROP PLANS : 1 ※ 削除した SQL 計画ベースラインの数が戻り値として表示されます

P) -1

SQL 文に関連付けられている SQL 計画履歴を承認して、SQL 計画ベースラインへと変更する

```
dbms_spm.evolve_sql_plan_baseline (
    sql_handle  IN VARCHAR2 := NULL,
    plan_name   IN VARCHAR2 := NULL,
    time_limit  IN INTEGER := DBMS_SPM.AUTO_LIMIT,
    verify      IN VARCHAR2 := 'YES',
    commit      IN VARCHAR2 := 'YES' )
return CLOB;
```

```
dbms_spm.evolve_sql_plan_baseline (
    plan_list   IN DBMS_SPM.NAME_LIST,
    time_limit  IN INTEGER := DBMS_SPM.AUTO_LIMIT,
    verify      IN VARCHAR2 := 'YES',
    commit      IN VARCHAR2 := 'YES' )
return CLOB;
```

戻り値

書式設定されたテキスト・レポートが含まれている CLOB

レポートには、ACCEPTED ステータスへの変更が可能な未承認の計画が順に示される

また、verify = 'YES' の場合は、それらの計画のパフォーマンスの検証結果も示される

パラメータ	説明
sql_handle	SQL 文の識別子 NULL にすると、plan_name を指定しないかぎり、SQL 計画ベースライン内の未承認の実行計画を含むすべての文が対象とされる
plan_name	実行計画の識別子 デフォルトの NULL にすると、識別された SQL 文が対象とされる sql_handle と sql_handle が共に NULL の場合は、すべての SQL 文の SQL 計画ベースラインで未承認の計画が対象とされる
plan_list	実行計画名のリスト リストのない各実行計画は、同じ SQL 文に属するものでも、別の SQL 文に属するものでもかまわない

time_limit	<p>時間制限(分)</p> <p>これは <code>verify = 'YES'</code> の場合にのみ適用される</p> <p>時間制限はグローバルで、次のように使用され、たとえば最初の未承認の実行計画を検証する場合の時間制限は入力値と同じ値に設定され、2番目の未承認の実行計画を検証する場合の制限時間は(入力値 - 最初の実行計画の検証に要した時間)という具合に設定される</p>
verify	<p>未承認の実行計画を承認済の実行計画に変更する前に、計画を実行し、パフォーマンスを比較するかどうかを指定するパラメータ</p> <p>パフォーマンスの検証では、未承認の計画および対応する SQL 計画ベースラインから選択された計画が実行され、これらの計画のパフォーマンス統計が比較される</p> <p>未承認の計画でパフォーマンスが向上した場合に限り、未承認の計画は承認済の計画に変更される</p> <ul style="list-style-type: none"> • 'YES'(デフォルト)にすると、未承認の計画を承認済の計画に変更する前に、未承認の計画でパフォーマンスが向上することを検証する • 'NO'にすると、検証は実行されず、未承認の実行計画が承認済の実行計画に強制的に変更される
commit	<p>未承認の計画の <code>ACCEPTED</code> ステータスを'NO'から'YES'に更新するかどうかを指定します。</p> <ul style="list-style-type: none"> • 'YES'(デフォルト): 条件を満たす未承認の計画を更新し、<code>verify = 'YES'</code> の場合はその更新およびパフォーマンスの検証結果を示すレポートをする • 'NO': 更新せずにレポートを生成する <p><code>verify = 'NO'</code> と <code>commit = 'NO'</code> を指定すると、操作できなくなる</p>

【注意事項】

ADMINISTER SQL MANAGEMENT OBJECT 権限が必要となる

P) -2

属性名/値の形式で SQL 文と関連付けられている 1 つまたはすべての計画の属性を変更する

登録されている実行計画の利用の可否を設定する

```
dbms_spm.alter_sql_plan_baseline (
    sql_handle IN VARCHAR2 := NULL,
    plan_name IN VARCHAR2 := NULL,
    attribute_name IN VARCHAR2,
    attribute_value IN VARCHAR2)
return PLS_INTEGER;
```

戻り値
変更した計画の数

パラメータ	説明
sql_handle	SQL 文のハンドル 属性変更のために SQL 文に関連付けられている実行計画を識別する NULL の場合は、plan_name を指定する必要がある
plan_name	実行計画名 特定の実行計画を識別する デフォルトの NULL にすると、sql_handle で識別される SQL 文に関連付けられているすべての実行計画の属性が設定される NULL の場合は、sql_handle を指定する必要があります。
attribute_name	設定する計画属性の名前
attribute_value	設定する計画属性の値

属性名	属性値	説明
attribute_name	attribute_value	
enabled	YES	実行計画をオプティマイザで使用できる
	NO	実行計画をオプティマイザで使用できない
fixed	YES	実行計画の自動登録を行わない状態
	NO	実行計画の自動登録を行わせる状態
autopurge	YES	特定の期間使用されない場合には実行計画は、ページされる
	NO	特定の期間使用されなくても実行計画は、ページされない
plan_name	計画の名前	最大 30 文字の文字列
description	計画の説明	最大 500 バイトの文字列