

単体 SQL 文に対する処理効率向上のための改善策の見つけ方

実行速度を上げるための SQL 文の記述方法のアドバイス [チューニング・アドバイザー] (対象 SQL 文 : SQL*PLUS から、直前に実行した SQL 文)

コマンドモードにて、SQL チューニング・アドバイザーを実行して、SQL 文のチューニング方法の分析を行う方法

通常は、Enterprise Manager (OEM 画面) を使って、SQL チューニング・アドバイザー処理を行います。

しかし、OEM 画面が使えない場合には、SQL*Plus から DBMS_SQLTUNE パッケージを使用することで、SQL チューニング・アドバイザーからのアドバイスレポートの表示や、SQL プロファイル (調査結果に対する各適用項目のこと) の適用が行えます。

指摘 (アドバイス) 事項

チューニング・アドバイザーでは、索引を使用出来るような Where 句の記述変更や and 条件句の記述変更によるテーブルの結合方法の変更など、SQL 文の記述の変更でアクセス性能が向上するような内容を指摘します

インデックス作成など、オブジェクトを別途作成するようなアドバイス指摘は行わない (これは、SQL アクセス・アドバイザーの役割)

利用ユーザーに貸与する権限

- SELECT ANY DICTIONARY
- OEM_ADVISOR ロール
- EXECUTE ON SYS.DBMS_SQLTUNE

```
GRANT SELECT ANY DICTIONARY TO "KOZUE";  
GRANT "OEM_ADVISOR" TO "KOZUE";  
GRANT EXECUTE ON SYS.DBMS_SQLTUNE TO "KOZUE";
```

注意事項

推奨事項で、ファンクション索引の作成 (create a function-base index) が提示された場合には、バッファキャッシュの消費、ストレージ領域の消費、更新処理のオーバーヘッド増加となるので、実行時間短縮とトレード・オフして考える

調査手順（実行手順）

(1) チェック対象の SQL の実行

```
SQL> Select * from emp
      2 Where MONTHS_BETWEEN( . . . . .
```

(2) 調査スクリプトの実行

```
SQL> @d:\temp\sqltune.sql
```

↓
必要な調査スクリプト (sqltune.sql)

```
set long 10000
set longchunksize 10000
set linesize 150
set pagesize 1000

VARIABLE my_sql_id VARCHAR2(13);
VARIABLE my_task_name VARCHAR2(64);

-- このセッションが最後に実行した SQL の SQL_ID を取得
begin
SELECT prev_sql_id INTO :my_sql_id FROM v$session
  WHERE sid = sys_context('USERENV', 'SID');
end;
/
-- ★
-- SQL_ID を指定して、SQL チューニングアドバイザー・タスクを作成
exec :my_task_name := DBMS_SQLTUNE.CREATE_TUNING_TASK( sql_id
=> :my_sql_id)

-- SQL チューニングアドバイザー・タスクの実行
/* ※ SQL チューニングアドバイザー・タスクを動作させるだけであって、
パフォーマンス改善用の処理を実行しているのではない */
exec DBMS_SQLTUNE.EXECUTE_TUNING_TASK(:my_task_name)

-- レポートの出力
SELECT DBMS_SQLTUNE.REPORT_TUNING_TASK(:my_task_name)
  FROM dual;

-- SQL チューニングアドバイザー・タスクの削除
exec DBMS_SQLTUNE.DROP_TUNING_TASK(:my_task_name)
```

調査結果のチェックポイント（出力セクション）

GENERAL INFORMATION SECTION ← 一般通知セクション

Number of SQL Restructure Finding : 数字 ← 発見した個数

FINDINGS SECTION (数字 finding) ← 問題点発見セクション

(分析結果)
SQL 文について、発見した事項の説明

Recommendation

対処方法の推奨事項

(調査結果例)

- Consider accepting the recommended **SQL profile**.

```
execute dbms_sqltune.accept_sql_profile(task_name=> 'TASK_300',  
                                       replace=> TRUE);
```

※ SQL プロファイルの実行を行い、オプティマイザの補助統計の作成を推奨している

このステートメントを SQL/Plus にコピー・ペーストして実行することにより、パフォーマンス効率が上がる実行計画を、オプティマイザが作成できるようになる

Rationale

推奨する根拠

EXPLAIN PLANS SECTION ← 実行計画

対処方法の根拠を検討する場合に、参考となる元の SQL に対する実行計画

1 – Original (対処適用前)

.....

2 – Using SQL Profile (対処適用後)

.....

(調査例)

```
select * from emp ← 調査用テスト SQL 文
where MONTHS_BETWEEN( SYSDATE, birthday ) >= 60;

@d:¥temp¥sqltune.sql
```

————— 実行結果 —————

【異常処理状態】

```
-----
Schema Name      : SYSTEM
SQL ID           : 9babjv8yq8ru3
SQL Text        : BEGIN DBMS_OUTPUT.GET_LINES(:LINES, :NUMLINES);
-----
```

ADDITIONAL INFORMATION SECTION

- SQL 文の型がサポートされていません。

このエラーが発生した原因は、ホスト変数:my_sql_id に実行した SQL 文の SQL_ID が、スクリプトの中でセットされないため。
対応策は、スクリプトの中のコマンドを個々に実行し、その中でホスト変数:my_sql_id に SQL_ID を明示的にセットする

```
select sql_id , sql_text
from v$sqlarea
where sql_text like 'select * from emp%';
```

SQL_ID	SQL_TEXT
6wh4tsk1p2fbk	select * from emp where MONTHS_BETWEEN(SYSDATE, birthday) >= 60

```
execute :my_sql_id := '6wh4tsk1p2fbk'
```

この後は、前ページの★印のステップから続行する

【正常処理状態】

DBMS_SQLTUNE.REPORT_TUNING_TASK(:MY_TASK_NAME)

GENERAL INFORMATION SECTION

Tuning Task Name : タスク_381
Tuning Task Owner : KOZUE
Workload Type : Single SQL Statement
Scope : COMPREHENSIVE
Time Limit(seconds) : 1800
Completion Status : COMPLETED
Started at : 05/24/2018 13:52:12
Completed at : 05/24/2018 13:52:12

Schema Name : SYSTEM
SQL ID : fs8xnsz5bqsbv
SQL Text : select * from emp where MONTHS_BETWEEN
(SYSDATE, birthday) >= 60

There are no recommendations to improve the statement.

(オブジェクトの統計情報が作成されていない例)

```
select * from emp2 where empno=1;
```

```
select sql_id , sql_text from v$sqlarea where sql_text like 'select * from emp2%';
```

SQL_ID	SQL_TEXT
85m0sfskacbj	select * from emp2 where empno=1

```
execute :my_sql_id := '<調査した SQL_ID 値>'
```

```
-- SQL_ID を指定して、SQL チューニングアドバイザー・タスクを作成  
exec :my_task_name := DBMS_SQLTUNE.CREATE_TUNING_TASK( sql_id  
=> :my_sql_id)
```

```
-- SQL チューニングアドバイザー・タスクの実行  
/* ※ SQL チューニングアドバイザー・タスクを動作させるだけであって、  
パフォーマンス改善用の処理を実行しているのではない */  
exec DBMS_SQLTUNE.EXECUTE_TUNING_TASK( :my_task_name )
```

```
-- レポートの出力  
SELECT DBMS_SQLTUNE.REPORT_TUNING_TASK( :my_task_name )  
FROM dual;
```

```
-- SQL チューニングアドバイザー・タスクの削除  
exec DBMS_SQLTUNE.DROP_TUNING_TASK( :my_task_name )
```

実行結果

```
DBMS_SQLTUNE.REPORT_TUNING_TASK(:MY_TASK_NAME)  
-----  
GENERAL INFORMATION SECTION  
-----  
Tuning Task Name           : タスク_396  
Tuning Task Owner          : KOZUE  
Workload Type              : Single SQL Statement  
Scope                      : COMPREHENSIVE  
Time Limit(seconds)       : 1800  
Completion Status          : COMPLETED  
Started at                 : 05/25/2018 10:06:25
```

Completed at

: 05/25/2018 10:06:25

Schema Name: KOZUE

SQL ID : 85m0sfskacbqj

SQL Text : select * from emp2 where empno=1

FINDINGS SECTION (1 finding)

1- Statistics Finding

表"KOZUE"."EMP2"は分析されませんでした。

Recommendation

指 摘 事 項

- この表に対するオプティマイザ統計の収集を検討してください。

```
execute dbms_stats.gather_table_stats(ownname => 'KOZUE', tabname =>
'EMP2', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt => 'FOR ALL COLUMNS SIZE AUTO');
```

Rationale

適切な実行計画を選択するには、最新のオプティマイザ統計が必要です。

EXPLAIN PLANS SECTION

1- Original

Plan hash value: 2941272003

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	58	3 (0)	00:00:01
* 1	TABLE ACCESS FULL	EMP2	1	58	3 (0)	00:00:01

Predicate Information (identified by operation id):

1 - filter("EMPNO"=1)

チューニング・アドバイザの指摘事項の実行

```
execute dbms_stats.gather_table_stats(ownname => 'KOZUE', tabname =>
'EMP2', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt
=> 'FOR ALL COLUMNS SIZE AUTO');
```

PL/SQL プロシージャが正常に完了しました。