

SQL トレースの取得方法

別セッションからの SQL\_TRACE の取得方法

サーバー・プロセス（ユーザー・セッション）の SID と SERIAL#  
の調査方法

SQL トレースを見るための出力形式変換（コンバート作業）

SQL トレースの出力内容

問題ポイントと原因についての解説

## SQL トレース・ファイルの概要

トレース指示を実施した期間中に実行したすべての SQL 文に対して、**実行計画**と**実行統計**の情報が出力される

ただし、トレースした情報ファイルは、一度コンバートを行わないと人が直接見ることとはできない

## SQL トレースの取得方法

### 方法 A

自身が使っている SQL\*Plus のセッションでの実行 SQL の SQL トレース取得

- PLUSTRACE ロールを持ったスキーマ、もしくは、SELECT ANY DICTIONARY 権限を付与されたスキーマで、SQL\*Plus を立ち上げる

```
sqlplus /nolog
```

```
connect <ユーザー名> / パスワード @ 接続識別子  
            もしくは、@IP アドレス or サーバー名 : 1521 / サービス名
```

```
sql> set lines 100 pages 100
```

- SQL トレースの開始

```
sql> alter session set sql_trace = true ;
```

- 調査する SQL 文 もしくは、無名 PL/SQL ブロックの実行

```
sql> <解析したい SQL 文の実行> ;
```

※ この場合、実行計画の表示とともに SQL 文の処理も同時実行される

```
例) sql> SELECT last_name FROM employ_table WHERE  
        employ_id = 125 ;
```

```
sql> /
```

- SQL トレースの終了

```
sql> alter session set sql_trace = false ;
```

- SQL/Plus のセッション切断

(確実に情報取得するために、切断は行うこと)

```
sql> exit
```

## 方法 B

（別セッションから、SQL\_TRACE の有効化設定をして、SQL トレースを取得する方法）

- PLUSTRACE ロールを持ったスキーマ、もしくは、SELECT ANY DICTIONARY 権限を付与されたスキーマで、SQL\*Plus を立ち上げる

```
sqlplus /nolog
```

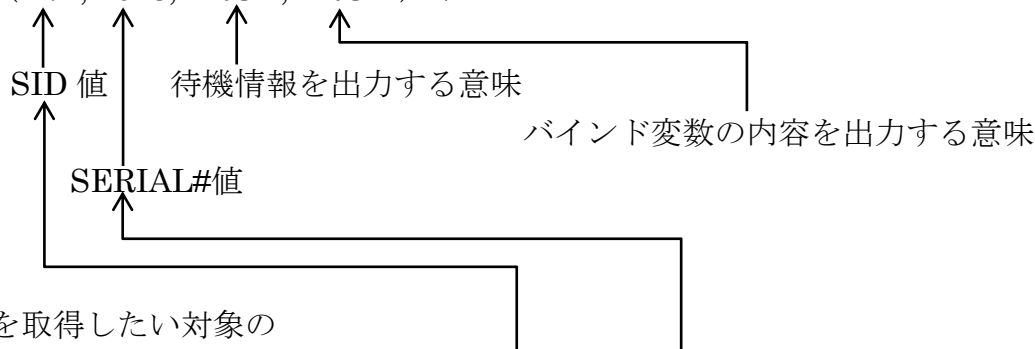
```
connect <ユーザー名> / パスワード @ 接続識別子  
もしくは、@IP アドレス or サーバー名 : 1521 / サービス名
```

```
sql> set lines 100 pages 100
```

- SQL トレースの開始

```
sql> EXECUTE DBMS_MONITOR.SESSION_TRACE_ENABLE
```

```
( 172, 1023, TRUE, TRUE ) ;
```



SQL トレースを取得したい対象の  
サーバー・プロセス（ユーザー・セッション）の SID と SERIAL# の調査方法

```
sql> select ses.sid, ses.serial#, ses.username, ses.program, ses.machine,  
ses.osuser from v$session ses, v$process proc  
where ses.paddr = proc.addr and ses.username is not null ;
```

表示された内容から、対象のセッションを見つけて指定する

SID	SERIAL#	USERNAME	PROGRAM	MACHINE	OSUSER
193	526	SYSTEM	sqlplus@dbse	(TNS V1-V3)	oracle
<b>172</b>	<b>1023</b>	<b>kozue</b>	<b>JDBC Thin Cl</b>	<b>pc-105</b>	<b>oracle</b>
193	526	SYSTEM	sqlplus@dbse	(TNS V1-V3)	oracle

- SQL トレースの終了

```
sql> EXECUTE DBMS_MONITOR.SESSION_TRACE_DISABLE  
( <SID 値> , <SERIAL#値> ) ;
```

## SQL トレースを見るための出力形式変換（コンバート作業）

SQL トレースを実行すると、`user_dump_dest` フォルダ・オブジェクトの中に、トレース情報が入ったデータ・ファイルが出来上がる

しかし、このままでは見ることができないので、見やすいようにファイル変換する  
出来上がったファイルの中で、最新のファイルを選択するとよい

**\$ tkprof** <トレース・ファイル名> <出力コンバート・ファイル名>

例) \$ tkprof ora10\_ora\_2132.trc /home1/userdata/list05.tkp

※ 出来上がったファイルには、SQL 文以外の情報（セッション接続や専用サーバー・プロセスなど）も大量に出力されるので、SQL 文の部分特定して、調査すること

## SQL トレースの出力内容

10053 トレースで取得される情報には、実行計画が変更された時に、その変更理由を調査するための有益な情報が含まれてきます

### 【10053 トレースの結果出力で確認すべきセクション】

確認セクション	確認内容
Current SQL Statement for this session	対象の（解析された）SQL 文
PARAMETERS USED BY THE OPTIMIZER	コスト計算に影響を与える初期化パラメータ
Query transformations (QT)	SQL の問合せ変換内容
SYSTEM STATISTICS INFORMATION	システム統計
Elapsed times include waiting on following events	待機イベントの発生状況
BASE STATISTICAL INFORMATION	オブジェクト統計
SINGLE TABLE ACCESS PATH	表へのアクセスパス計算・比較
GENERAL PLAN	結合順序・結合方法の計算・比較
Plan Table	最終的に選択された実行計画

[実行計画]

実行計画についての内容

[実行統計]



\*\*\*\*\*

```

select department_name from employees e, departments d
where e.department_id = d.department_id
and salary > 1000

```

← 実行した SQL 文

[SQL の実行情報] 部分

call	count	cpu	elapsed	disk	query	current	rows
-----	-----	-----	-----	-----	--- ↑ --	--- ↑ ---	-----
↑	解析数	↑	CPU+待機時間	↑	アクセスしたブロック数		↑
解析種		CPU 時間		ディスク読取りブロック			アクセスレコード数
Parse	1	0.02	0.06	0	0	0	0
Execute	1	0.00	0.00	0	0	0	0
Fetch	4	0.01	0.24	187	292	0	35
-----	-----	-----	-----	-----	-----	-----	-----
total	6	0.03	0.30	187	292	0	35

[実行計画] 部分

Rows(1st)	Rows(avg)	Rows(max)	Rows	Source	Operation
-- ↑ -----	-----	-----	-----	-----	-----
アクセスレコード数 (初回)、(平均)、(最大)					↑
605					実行計画に関する説明
					SORT GROUP BY
					→ (cr=808 pr=300 pw=0 time=8293us)
605	605	605		TABLE ACCESS FULL	TABLE_1
					→ (cr=808 pr=300 pw=0 time=8293us cost=305 card=200)
cr :	処理した論理ブロック数			pr :	処理した物理ブロック数
pw :	ディスクに書き出したブロック数			time :	実行時間 (マイクロ秒)
cost :				card :	

SQL 文のアクセス・データ量見積り

(1行目は、SQL 文全体での合計値)

(2行目以降は、そのステップでのアクセス・データ量)

待機イベント情報（追加）取得のための SQL トレース・コマンド

```
> alter session set event '10046 trace name context forever , level 8';
```

この後、取得したトレース・ファイルに対して、tkprof ユーティリティを使用して、リスト出力を行います

[待機イベント情報] 部分

Elapsed times include waiting on following events:

待機イベントに関する待ち時間統計情報の表示

Event waited on	Times	Max. Wait	Total Waited
SQL*Net message to client	2	0.00	0.00
db file scattered read マルチブロック・リードでの I/O 待機	151	0.04	2.02
db file sequential read シングルブロック・リードでの I/O 待機	4	0.00	0.02
SQL*Net message from client	2	0.00	0.00

1 ブロック当りの I/O レスポンス時間  $2.02 / 151 = 0.013$  秒

この SQL 実行において、マルチブロック・リードが 151 回、シングルブロック・リードが 4 回行われたことがわかる

## レスポンス悪化のボトルネック・ポイントの発見とその処理時間長についての調査

- どの SQL 文に長時間かかっているか
- どの部分（実行ステップ）で時間がかかっているか
- CPU 実行時間と待機イベント時間は、妥当か
- どの種類の待機イベントに時間がかかっているか
- テーブルへのアクセス方法は妥当か（インデックス使用の有無）
- テーブル・ループの種類は妥当か  
（ネステッドループ結合などこの SQL 文に対しての効率が良いテーブル・ループが選択されているか）
- アクセス（読み込み、書き込み）ブロック数の検討  
（テーブル劣化やインデックス劣化によりで、対象ブロック数が多すぎないか）

などを検討して、長時間原因を推測し、時間短縮のための対策を行う