

V\$SESSION 動的ビュー（処理中の起動セッション情報）の構成列  
セッションに紐付いた他の V\$動的ビューとの関係  
（関係テーブルをリンク（レコード連結）させる Where 句の指定条件）

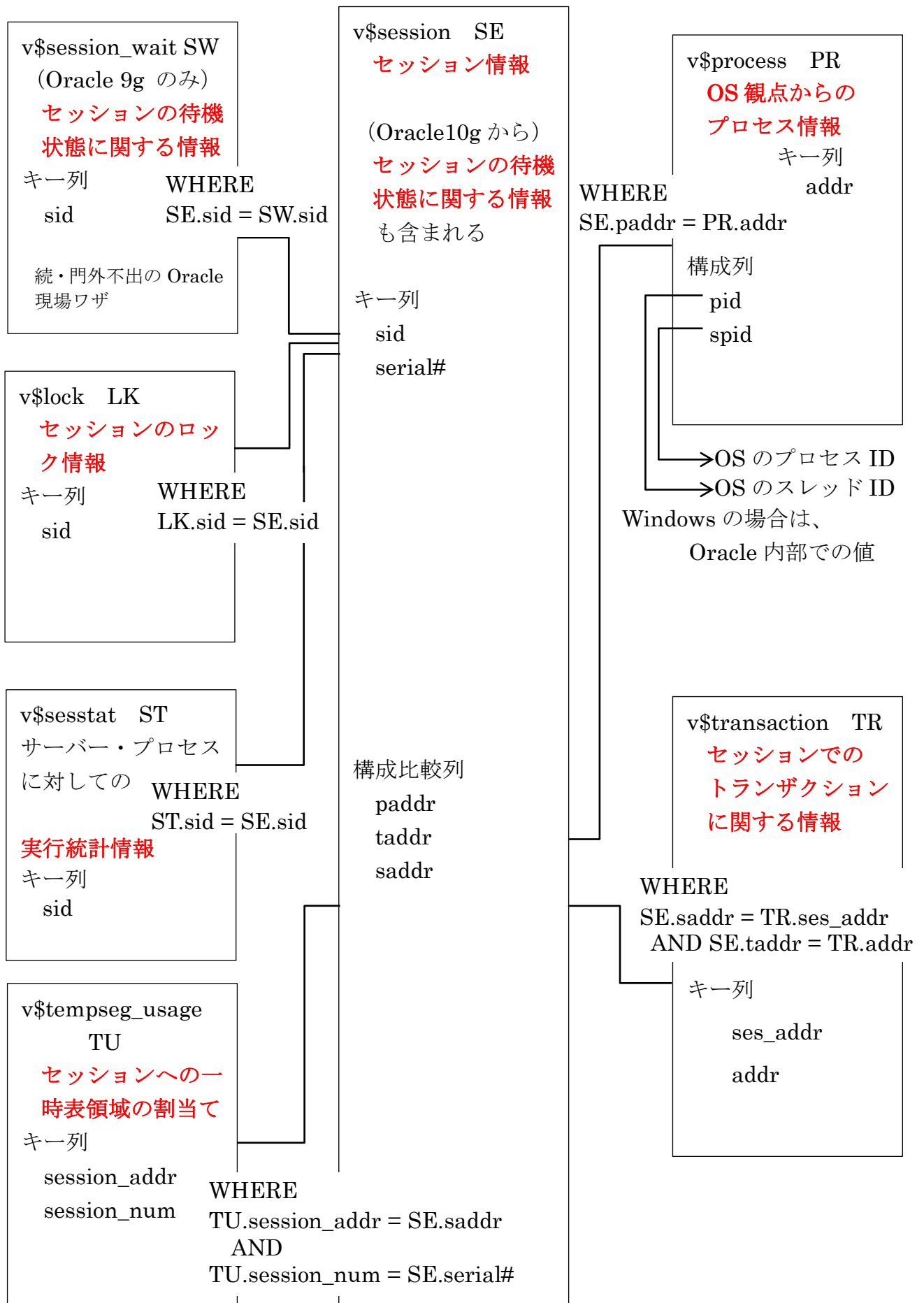
V\$SESSION 動的ビューとそれに紐付いた他の V\$ビューの関係  
（関係レコードを抽出するための比較列 Where 条件）

- ・次ページ参照

セッション別の Oracle データベース・リソース利用状況の関連 V\$ビューから調査出来ること

・ SQL 文の実行統計に関する情報 （ライブラリキャッシュ上の	V\$SQLAREA
・ SQL 文の実行計画に関する情報 （ライブラリキャッシュ上の）	V\$SQL_PLAN
・ セッション別 PGA メモリ使用量の調査	V\$PROCESS
・ 長時間待機しているセッションの確認	V\$SESSION
・ 長時間アクティブなセッションの確認	V\$SESSION
・ 行ロックを行っているセッションの特定と対処法	V\$LOCK
・ SQL 文に対する実行統計情報 CPU リソースを大量に使用しているセッションの確認	V\$SESSTAT
・ 実行している SQL 文についての内容情報 特定の SQL を実行しているセッションの確認	V\$SQL
・ セッションとプロセスの対応確認	V\$PROCESS
・ トランザクション実行中のセッションの確認	V\$TRANSACTION
・ 一時表領域を使用しているセッションの確認	V\$TEMPSEG_USAGE
・ OS 観点から見たセッションのプロセス情報	V\$PROCESS
・ セッション番号とプロセス番号の関連調査 及び、システムリソース利用状況の調査	V\$PROCESS
・ セッション（サーバー・プロセス）の強制終了方法	V\$PROCESS

# V\$SESSION 動的ビューとそれに紐付いた他の V\$ビューの関係 ( 関係レコードを抽出するための比較列 Where 条件 )



## v\$sql、v\$sql\_plan について

10 g 以降

v\$sql SL

**SQL 文に関する  
情報**

WHERE

SL.sql\_id = SE.sql\_id AND  
SL.child\_number  
= SE.sql\_child\_number

（ — 参考情報 —  
実行計画は、  
v\$sql\_plan ビュー）

キー列

sql\_id  
child\_number

WHERE

SL.sql\_id = SE.prev\_sql\_id  
AND  
SL.child\_number  
= SE.prev\_child\_number

1 つ前のステップの SQL 文に  
対する情報を入手する場合の  
V\$SESSION 側だけの指定  
prev\_

v\$session SE  
の続き 1

現在実行中 SQL は、

構成比較列  
(Oracle10g 以降)  
sql\_id  
sql\_child\_number

(Oracle 9g 以前)  
sql\_address  
sql\_hash\_value

直近に実行完了の  
SQL に関しては、

構成比較列  
(Oracle10g 以降)  
prev\_sql\_id  
prev\_child\_number

(Oracle 9g 以前)  
prev\_sql\_addr  
prev\_hash\_value

9 g 以前

v\$sql SL

**SQL 文に関する  
情報**

WHERE

SE.sql\_address  
= SL.address AND  
SE.sql\_hash\_value  
= SL.hash\_value

キー列

address  
hash\_value

WHERE

SE.prev\_sql\_addr  
= SL.address AND  
SE.prev\_hash\_value  
= SL.hash\_value

1 つ前のステップの SQL 文に  
対する情報を入手する場合の  
V\$SESSION 側だけの指定  
prev\_

## v\$sqlarea について

10 g 以降

v\$sqlarea SA

**実行統計の情報**

(ライブラリキャッシュ内)  
すべての実行の合計値

WHERE  
SA.sql\_id = SE.sql\_id

キー列  
sql\_id

WHERE  
SA.sql\_id  
= SE.prev\_sql\_id

1 つ前のステップの SQL 文に  
対する情報を入手する場合の  
V\$SESSION 側だけの指定  
prev\_

v\$session SE  
の続き 2

現在実行中 SQL は、

構成比較列  
(Oracle10g 以降)

sql\_id  
~~sql\_child\_number~~

(Oracle 9g 以前)  
sql\_address  
sql\_hash\_value

直近に実行完了の  
SQL に関しては、

構成比較列  
(Oracle10g 以降)

prev\_sql\_id  
~~prev\_child\_number~~

(Oracle 9g 以前)  
prev\_sql\_addr  
prev\_hash\_value

9 g 以前

v\$sqlarea SA

**実行統計の情報**

(ライブラリキャッシュ内)  
すべての実行の合計値

WHERE  
SE.sql\_address  
= SA.address AND  
SE.sql\_hash\_value  
= SA.hash\_value

キー列  
address  
hash\_value

WHERE  
SE.prev\_sql\_address  
= SA.address AND  
SE.prev\_hash\_value  
= SA.hash\_value

1 つ前のステップの SQL 文に  
対する情報を入手する場合の  
V\$SESSION 側だけの指定  
prev\_

過去のセッションの情報を調査する場合は、  
DBA\_HIST\_ACTIVE\_SESS\_HISTORY を調査する（セッション終了分あり）

サンプル SQL コマンド)

```
SELECT * from v$session SE , v$session_wait SW WHERE SE.sid = SW.sid ;
```

```
SELECT * from v$session SE , v$process PR WHERE SE.paddr = PR.addr ;
```

```
SELECT * from v$session SE , v$lock LK WHERE SE.sid = LK.sid ;
```

```
SELECT * from v$session SE , v$sesstat ST WHERE SE.sid = ST.sid ;
```

```
SELECT * from v$session SE , v$transaction TR  
WHERE SE.saddr = TR.ses_addr AND SE.taddr = TR.addr ;
```

```
SELECT * from v$session SE , v$tempseg_usage TU  
WHERE SE.saddr = TU.session_addr AND SE.serial# = TU.session_num ;
```

( Oracle10 g 以降 )

```
SELECT * from v$session SE , v$sql SL  
WHERE SE.sql_id = SL.sql_id AND SE.child_number = SL.sql_child_number ;
```

```
SELECT * from v$session SE , v$sql SL  
WHERE SE.sql_id = SL.prev_sql_id AND SE.child_number = SL.prev_child_number ;
```

```
SELECT * from v$session SE , v$sqlarea SA  
WHERE SE.sql_id = SA.sql_id ;
```

```
SELECT * from v$session SE , v$sqlarea SA  
WHERE SE.sql_id = SA.prev_sql_id ;
```

( Oracle9 g )

```
SELECT * from v$session SE , v$sql SL  
WHERE SE.sql_address = SL.address AND SE.sql_hash_value = SL.hash_value ;
```

```
SELECT * from v$session SE , v$sql SL  
WHERE SE.prev_sql_addr = SL.address AND SE.prev_hash_value = SL.hash_value ;
```

```
SELECT * from v$session SE , v$sqlarea SA
WHERE SE.sql_address = SA.address AND SE.sql_hash_value = SA.hash_value ;
```

```
SELECT * from v$session SE , v$sqlarea SA
WHERE SE.prev_sql_addr = SA.address AND SE.prev_hash_value = SA.hash_value ;
```

## 【v\$sqlsession のレコードに対する実行統計情報の出力】

注意)

V\$SQLAREA ビューは、対象の SQL 文が複数回実行された場合は、そのすべての実行の実行記録の合計値としての実行統計を示す

実行した 1 回分の実行実績記録ではないので、注意すること

実行中の SQL 文に対しての実行実績数値の調査ならば、V\$SESSION から求める

```
select  SID, SERIAL#, SUBSTR(SA.SQL_FULLTEXT, 0, 40) SQL_TEXT ,
        SA.CPU_TIME , SA.ELAPSED_TIME , SA.EXECUTIONS ,
        SA.DISK_READS , SA.BUFFER_GETS , SA.USER_IO_WAIT_TIME
from    V$SESSION SE, V$SQLAREA SA
where   SA.SQL_TEXT like 'select%'
        and SE.SQL_ID = SA.SQL_ID ;
```

SID	SERIAL#	SQL_TEXT
106	2341	select SID, SERIAL#, SUBSTR(SA.SQL_T
127	605	select substr(SA.SQL_TEXT, 0, 40), S

ELAPSED_TIME	DISK_READS
CPU_TIME     ↓	EXECUTIONS     ↓
15600	12959
31200	27505

EXECUTIONS	DISK_READS	BUFFER_GETS
1	17788	480
2	3911	870

## V\$SQL (V\$SQLTEXT) と V\$SQLAREA の関係

( Oracle10 g 以降 )

```
SELECT * from v$sql SL, v$sqlarea SA
WHERE SL.sql_id = SA.sql_id ;
```

( Oracle9 g )

```
SELECT * from v$sql SL, v$sqlarea SA
WHERE SL.address = SA.address AND SL.hash_value = SA.hash_value ;
```

## V\$SQL (V\$SQLTEXT) と V\$SQL\_PLAN の関係

( Oracle10 g 以降 )

```
SELECT * from v$sql SL, v$sql_plan VP
WHERE SL.sql_id = VP.sql_id ;
```

( Oracle9 g )

```
SELECT * from v$sql SL, v$sql_plan VP
WHERE SL.address = VP.address AND SL.hash_value = VP.hash_value ;
```



## V\$SESSION 動的ビューの構成年列

### v\$session ビューの主な列とその列項目

列	概 要
sid	セッション識別子
serial#	セッションシリアル番号
} この組合せが識別キー	
<p>sid は、(専用 or 共用) サーバー・プロセスに割当てられたセッション番号になる よって、クライアントから何度か再接続を行った場合には、過去に使用したサーバー・プロセスと同一のサーバー・プロセスでの接続となった場合には、セッション番号は同一の番号が割当てられる</p> <p>serial#は、接続したときのシーケンス番号になる これは、クライアントからの再接続時には、前回と異なった番号が必ず割当てられる</p> <p>よって、この2つの組合せが、v\$session ビューのセッション・オブジェクトを一意に識別するキーとなる</p>	
seq#	現在または前回の <b>待機</b> を一意に識別する番号 (待機ごとに増分される) 待機が解消されるごとに、次の待機処理時にはシーケンス番号はカウントアップされていく
username	Oracle のユーザー名
audsid	監査セッション ID 監査情報ビューの列と結合させレコードを特定する
paddr	セッションに対応する <b>プロセスを特定</b> させるアドレス V\$PROCESS ビューの addr 列と結合させレコードを特定する
taddr	<b>トランザクション</b> を示すアドレス V\$TRANSACTION ビューの addr 列と結合させレコードを特定する
saddr	<b>一時表領域との対応</b> を紐づけるアドレス V\$TEMPSEG_USAGE ビューの session_addr 列と結合させレコードを特定する
server	サーバー・プロセスの接続形態 DEDICATED            専用サーバー・プロセス SHARED/NONE        共有サーバー・プロセス
osuser	クライアントアプリケーションを実行している OS のユーザー名
process	クライアントアプリケーションに対する、OS のプロセスが実行されているプロセス ID

machine	Oracle クライアントアプリケーションが実行されているマシン名	
terminal	Oracle クライアントアプリケーションが実行されている端末名	
program	Oracle クライアントアプリケーションが実行されている端末で、実行されているプログラム名	
type	実行されているプロセスの種類 USER : ユーザー・セッション・プロセス (サーバー・プロセス) BACKGROUND : バックグラウンド・プロセス	
logon_time	ログインが開始された時刻	
(Oracle10 g 以降)		
sql_id	現在実行されている SQL 文の SQL 識別子	
sql_child_number	現在実行されている SQL 文の実行計画を決定づけるための識別番号 〔ライブラリ・キャッシュには、1つの SQL 文に対して、複数の実行計画を確保することが可能である その場合の実行計画を決定するための識別番号〕	
prev_sql_id	前回に実行された SQL 文の SQL 識別子	
prev_child_number	前回に実行された SQL 文の子カーソルの番号	
(Oracle9 g 以前)		
sql_address	現在実行している SQL 文のアドレス値	この 2 つの組合せが、v\$session ビューのセッション・オブジェクトを一意に識別するキーとなる
sql_hash_value	現在実行している SQL 文のハッシュ値	
prev_sql_addr	前回に実行された SQL 文のアドレス値	この 2 つの組合せが、v\$session ビューのセッション・オブジェクトを一意に識別するキーとなる
sql_hash_value	前回に実行された SQL 文のハッシュ値	
(Oracle9 g 以前) と (Oracle10 g 以降) 共通		
status	セッションの状態 (SQL 文を実行中かどうか分かる)  ACTIVE : CPU 使用中 or 待機イベント中 INACTIVE : アイドル状態 KILLED : 強制終了の処理中 CACHED : 分散トランザクションを実行するため、一時的にキャッシュに入っている状態 SNIPED : 長時間待機中	
state	待機状態。WAITING (セッションは現在待機中) WAITING : 待機イベント中 もしくは、アイドル状態 ※ 上の status 列との組合せで判断する WAITED UNKNOWN TIME : 前回の待機時間不明	

	<div>現在は処理中</div> <div>WAITED SHORT TIME : 前回の待機時間は、0.01 秒未満</div> <div>現在は処理中</div> <div>WAITED KNOWN TIME : 前回の待機時間は wait_time 列に示されている</div> <div>現在は処理中</div> <div>[重要]</div> <div>WAITING 以外の時は、現在実行中であり、EVENT# の値は意味を持たない</div>															
wait_time_micro	待機時間 (マイクロ秒) (Oracle9 g では、V\$SESSION_WAIT ビュー) 推奨															
wait_time	前回の待ち時間 (単位 : 10 ミリ秒) 但し、現在待機中 (state 列=WAITING) の場合は 0 (Oracle9 g では、V\$SESSION_WAIT ビュー) すぐに次の待機になるので、使用出来ないと考えられる															
seconds_in_wait	同一の待機イベントで待機している延べ待ち時間 一度待機から解除されると、0 にリセットされる <del>(11g からマイクロ秒に変更)</del> (Oracle9 g では、V\$SESSION_WAIT ビュー)															
<div>待機状態とは、</div> <div>status＝active で、state＝wait</div> <div>このときは、wait_time_micro は、経過中の待機時間を表す</div> <div>活動状態とは、</div> <div>status＝active で、state≠wait</div> <div>このときは、wait_time_micro は、前回の待機状態での待機時間を表す</div> <div>アイドル状態とは、</div> <div>status＝inactive で、state＝wait</div> <div>このときは、wait_time_micro は、アイドル状態での経過時間を表す</div>																
<div>state と wait_time の関係</div> <div>すぐに次の待機になるので、使用出来ないと考えられる</div> <table><tr><th>state</th><th>wait_time</th><th>意味</th></tr><tr><td>WAITING</td><td>0</td><td>現在待機中 待機が解消されてから、待機時間の値がセットされる</td></tr><tr><td>WAITED UNKNOWN TIME</td><td>-2</td><td>前回の待機時間が不明</td></tr><tr><td>WAITED SHORT TIME</td><td>-1</td><td>前回の待機時間が 10ms 未満だった</td></tr><tr><td>WAITED KNOWN TIME</td><td>&gt; 0</td><td>前回の待機時間は、wait_time 秒だった</td></tr></table>		state	wait_time	意味	WAITING	0	現在待機中 待機が解消されてから、待機時間の値がセットされる	WAITED UNKNOWN TIME	-2	前回の待機時間が不明	WAITED SHORT TIME	-1	前回の待機時間が 10ms 未満だった	WAITED KNOWN TIME	> 0	前回の待機時間は、wait_time 秒だった
state	wait_time	意味														
WAITING	0	現在待機中 待機が解消されてから、待機時間の値がセットされる														
WAITED UNKNOWN TIME	-2	前回の待機時間が不明														
WAITED SHORT TIME	-1	前回の待機時間が 10ms 未満だった														
WAITED KNOWN TIME	> 0	前回の待機時間は、wait_time 秒だった														
last_call_et	status 列が、「ACTIVE」 or 「INACTIVE」になってからの経過時間 (単位 : 秒)															

SQL の実行時間 (CPU 時間+待機時間) : (単位 : 秒) last_call_et もしくは、 round( (SYSDATE - SQL_EXEC_START)*86300 , 0) ELAPSED_SED	
sql_exec_start	SQL の開始時間 TO_CHAR( SQL_EXEC_START , 'YYYY-MM-DD HH24:MI:SS' ) 経過時間 :
<p>実行中の SQL コマンドに対しての CPU 使用合計時間、待機合計時間を示すような項目列は、V\$SESSION には存在していない</p> <p>この 2 つの値を必要とした時の対処方法としては、</p> <ul style="list-style-type: none"> <li>• SQL トレースを実施して、時間測定を行う方法</li> <li>• V\$SQLAREA から過去の実績値を求める (複数回実行の場合には、平均値を計算しての使用となる)</li> </ul>	
event	待機イベント名 (Oracle9 g では、V\$SESSION_WAIT ビュー)
event#	待機イベント番号 (Oracle9 g では、V\$SESSION_WAIT ビュー)
wait_class	待機クラス名
time_since_last_wait_micro	前回の待機終了後の経過時間 (マイクロ秒)
P1	第 1 パラメータ。待機イベントごとに異なる意味を持つ。 パラメータの意味は各待機イベントのリファレンスに記載
P2	第 2 パラメータ。待機イベントごとに異なる意味を持つ。 パラメータの意味は各待機イベントのリファレンスに記載
P3	第 3 パラメータ。待機イベントごとに異なる意味を持つ。 パラメータの意味は各待機イベントのリファレンスに記載
command	進行中のコマンド (最後に解析された文)
lockwait	セッションが待機しているロックのアドレス。ロックがない場合は NULL
blocking_session_status	ブロッキングセッションがあるかどうかを示す ブロックされている状態 : VALID ブロックされていない状態 : NO HOLDER ※ 他をブロックしているかどうかについては、この列項目は関知していない
blocking_instance	ブロッキングセッションのインスタンス識別子
blocking_session	このセッションを待たせているセッション番号 (原因元セッション)

サンプル SQL コマンド)

SELECT

sid , serial# , username , audsid , paddr , taddr , saddr , server , osuser , process ,  
machine , terminal , program , type , logon\_time , sql\_id , sql\_child\_number ,  
prev\_sql\_id , prev\_child\_number , sql\_address , sql\_hash\_value , prev\_sql\_addr ,  
sql\_hash\_value , status , state , last\_call\_et , event , event# , wait\_class ,  
time\_since\_last\_wait\_micro , P1 , P2 , P3 , wait\_time , wait\_time\_micro ,  
seconds\_in\_wait , command , lockwait , sql\_exec\_start , blocking\_session\_status ,  
blocking\_instance , blocking\_session , seq#

FROM V\$SESSION;

## 長時間待機しているセッションの調査方法

### 10 秒以上待機を継続しているセッションの調査方法

second\_in\_wait : 同一の待機イベントで待機している延べ待ち時間  
(11g からマイクロ秒に変更)  
(Oracle9 g では、V\$SESSION\_WAIT ビュー)

```
select se.sid , se.serial# , se.seq# , se.status , se.state , se.wait_time_micro ,
       se.event , sa.sql_fulltext
from v$session se , v$sqlarea sa
where se.sql_id = sa.sql_id(+)
      and wait_time_micro >= 10000000
      and status = 'ACTIVE'
      and state = 'WAITING'
      and type = 'USER'
;

and username = 'KOZUE'
and terminal = 'BUSINESS2-PC'
```

↑

クライアントを特定できる条件を指定すると、特定セッションの SQL 文の情報が調査しやすくなる

SID	serial#	SEQ#	STATUS	STATE	WAIT_TIME_MICRO
182	2070	25	ACTIVE	WAITING	2508148865

EVENT	SQL_TEXT
enq: TX - row lock contention	Select * from . . . . .

sid	serial#	username	event	st	wt	siw
141	16	SCOTT	enq : TX – row lock co	WAITING	0	120
153	23	SCOTT	SQL*Net message fro	WAITING	0	98

## 長時間アクティブなセッションの確認

(待機状態のものも含む)

### 長時間アクティブなセッションの調査方法

last\_call\_et: status 列が、「ACTIVE」 or 「INACTIVE」になってからの経過時間

```
SELECT  sid , serial# , username , status , state , last_call_et ,
        event
FROM    V$SESSION
WHERE   username = '<ユーザー名>'
        AND last_call_et > <秒数>
        AND status = 'ACTIVE';
```

sid	serial#	username	status	last_call_et
141	16	SCOTT	ACTIVE	1120

### 対象 SQL 文の CPU 処理時間と待機イベント待ち時間の出力

(待機状態のものも含む)

この SQL 文の処理に入ってから (待機時間の合計) と CPU 処理時間の和

last\_call\_et

もしくは、round((SYSDATE - SQL\_EXEC\_START)\*86300,0) ELAPSED\_SEC

### CPU 処理時間の和

V\$SQLAREA から、1 回分の平均値として求めるしか方法がない

cpu\_time / executions / 1000000 "cpu\_time(sec)"

### 現在の待機状態に入ってから時間

SECONDS\_IN\_WAIT

WAIT\_TIME\_MICRO

SQL 文の処理が終了していなくても、一旦待機状態が解消されると次の待機に入った時には 0 からの計測となる

```
SELECT  sid , serial# , username , status , state , last_call_et , event
FROM    V$SESSION
WHERE   username = '<ユーザー名>'
        AND last_call_et > <秒数>
        AND status = 'ACTIVE';
```

sid	serial#	username	status	last_call_et
141	16	SCOTT	ACTIVE	1120