

Oracle Direct Seminar



ORACLE®

**実践!!パフォーマンスチューニング
～ Statspack解析Tips ～**

日本オラクル株式会社

Oracle Direct

アジェンダ

- Statspackとは？
- Statspackの解析方法
- データベース
自動チューニング機能

Oracle Directの無償技術サービス

- SQL Serverからの移行アセスメント
- MySQLからの移行相談
- PostgreSQLからの移行相談
- Accessからの移行アセスメント
- Application Server 移行相談
- Oracle Database バージョンアップ支援
- Oracle Developer/2000 Webアップグレード相談
- パフォーマンス・クリニック
- Oracle Database 構成相談
- Oracle Database 高可用性診断
- システム連携アセスメント

<http://www.oracle.com/lang/jp/direct/services.html>

Statspackとは？

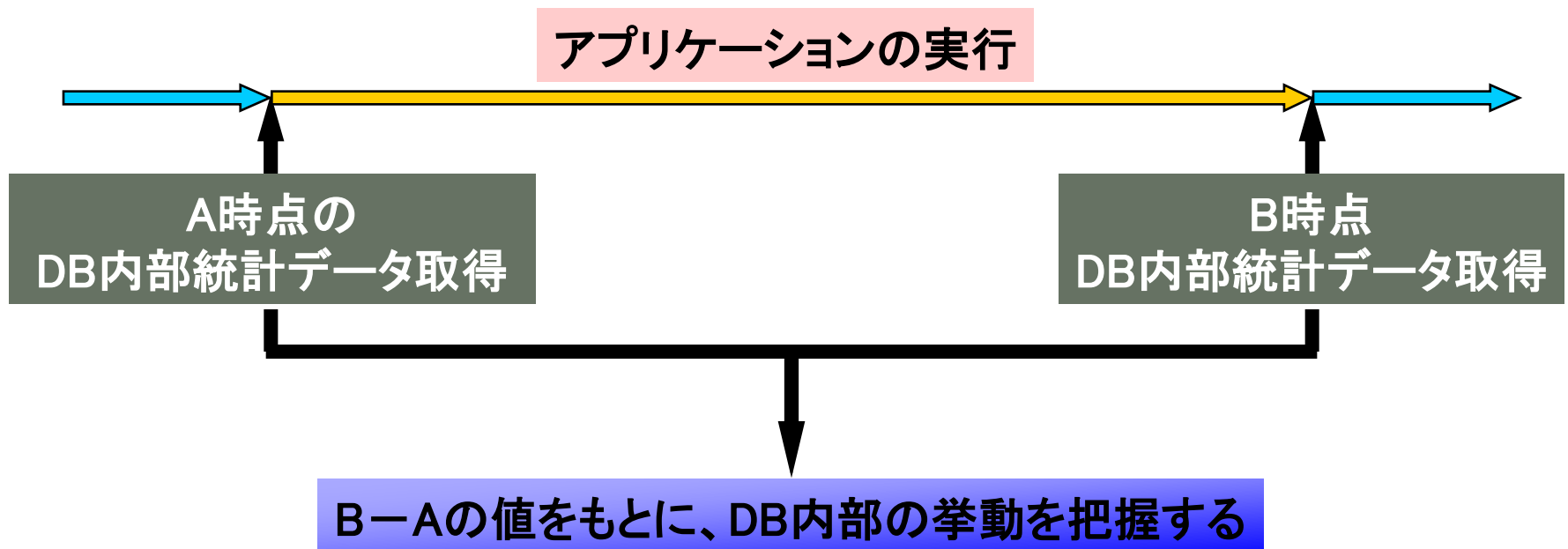
- ある期間での**Oracleが行なった処理の統計情報**
 - キャッシュヒット率
 - 待ち時間の内訳
 - トランザクション統計など
 - DBサーバー内部で、何が行われているかわかる。

**パフォーマンス・チューニングに役立つ情報を
レポートという形で提供するツール**

- システムのボトルネックが**DBであった時に活躍!!**

Statspackとは？

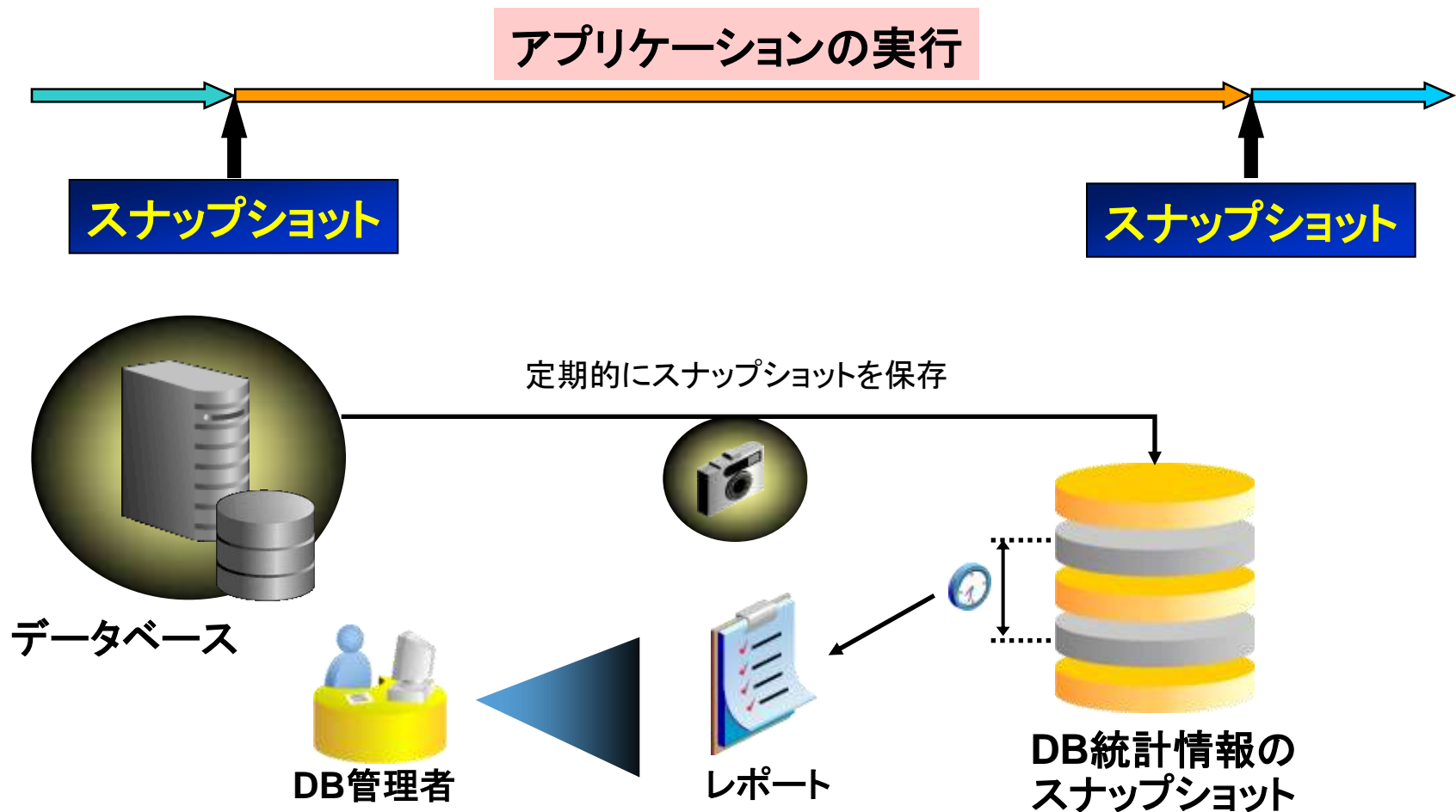
～ **STATISTICS PACKAGE** ～



ある2時点で取得した内部統計データの差分を元に、その間のパフォーマンス統計データを結果レポートに出力

Statspackとは？

～ STATISTICS PACKAGE ～



Statspack レポートの中身は・・・

F:\Oracle\Script\sp_161_162.LST - Microsoft Internet Explorer

ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

STATSPACK report for

DB Name	DB Id	Instance	Inst Num	Release	Cluster	Host
ORA9I	2338399647	ora9i	1	9.2.0.5.0	NO	KKABATA-JP

	Snap Id	Snap Time	Sessions	Curs/Sess	Comment
Begin Snap:	161	07-Oct-04 10:50:27	43	3.8	
End Snap:	162	07-Oct-04 10:55:30	43	3.7	
Elapsed:		5.05 (mins)			

Cache Sizes (end)

Buffer Cache:	24M	Std Block Size:	8K
Shared Pool Size:	48M	Log Buffer:	512K

Load Profile

	Per Second	Per Transaction
Redo size:	10,331.04	703.78
Logical reads:	2,962.57	201.81
Block changes:	70.19	4.78
Physical reads:	2.26	0.15
Physical writes:	1.24	0.08
User calls:	138.97	9.47
Parses:	64.60	4.40
Hard parses:	0.00	0.00
Sorts:	0.40	0.03
Logons:	0.01	0.00
Executes:	65.08	4.43
Transactions:	14.68	

% Blocks changed per Read:	2.37	Recursive Call %:	35.54
Rollback per transaction %:	0.00	Rows per Sort:	86.35

Instance Efficiency Percentages (Target 100%)

Buffer Nowait %:	100.00	Redo NoWait %:	100.00
Buffer Hit %:	99.92	In-memory Sort %:	100.00
Library Hit %:	100.00	Soft Parse %:	100.00
Execute to Parse %:	0.74	Latch Hit %:	100.00
Parse CPU to Parse Elapsed %:	88.54	% Non-Parse CPU:	90.84

F:\Oracle\Script\sp_161_162.LST - Microsoft Internet Explorer

ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

Memory Usage %: 94.50 94.50
% SQL with executions>1: 75.19 75.19
% Memory for SQL w/exec>1: 69.75 69.75

Top 5 Timed Events

Event	Waits	Time (s)	% Total
db file sequential read	678	10	30.93
CPU time		9	29.51
log file sync	4,449	8	24.92
log file parallel write	4,278	3	10.67
control file sequential read	84	1	1.60

Wait Events for DB: ORA9I Instance: ora9i Snaps: 161 -162

-> s - second
-> cs - centisecond - 100th of a second
-> ms - millisecond - 1000th of a second
-> us - microsecond - 1000000th of a second
-> ordered by wait time desc, waits desc (idle events last)

Event	Waits	Timeouts	Total Wait Time (s)	Avg wait (ms)	Waits /txn
db file sequential read	678	0	10	14	0.2
log file sync	4,449	0	8	2	1.0
log file parallel write	4,278	4,161	3	1	1.0
control file sequential read	84	0	1	6	0.0
control file parallel write	98	0	0	3	0.0
db file parallel write	62	0	0	3	0.0
buffer busy waits	8	0	0	13	0.0
latch free	41	10	0	1	0.0
db file scattered read	2	0	0	20	0.0
undo segment extension	1,302	1,302	0	0	0.3
SQL*Net message from client	22,617	0	9,065	401	5.1
wakeup time manager	9	9	276	30721	0.0
jobq slave wait	44	42	134	3048	0.0
SQL*Net message to client	22,617	0	0	0	5.1

Background Wait Events for DB: ORA9I Instance: ora9i Snaps: 161 -162

-> ordered by wait time desc, waits desc (idle events last)

	Total Wait	Avg wait	Waits
--	------------	----------	-------

Statspackのインストール方法・取得方法に関しては以下のURLをご参照ください。

http://www.oracle.com/lang/jp/direct/service/doc/how_to_get_statspack.pdf

ORACLE

Statspackで取れる情報(9iR2,10g,11g)

私の(個人的な)お奨め

通常稼動時は**Level 5(デフォルト)**、「最近ちょっと遅いなあ」と感じ始めたら**Level 6** or **Level 7**、Level 10はOracle サポート等から指示された時のみ(高負荷)

スナップ ショット レベル	収集データ					
	基本統計 情報	アドバイ ス情報	SQL統 計情報	SQL詳細 情報	セグメント 情報	ラッチ詳細 情報
Level 0	○	○				
Level 5	○	○	○			
Level 6	○	○	○	○		
Level 7	○	○	○	○	○	
Level 10	○	○	○	○	○	○

実行計画の出力 ～Level6以上、9.0.1以上～

- Statspackレポートの中には、SQLの**実行計画は出ません!!**SQLも全文表示される訳ではありません(最初の5行のみ)!!
 - StatspackでDB全体感を把握し、個別のSQL Tuningを実施したい時に別途実行計画出力手順を実施します。
1. Statspackレポートの確認
 - 実行計画を確認したいSQLが載っている**Snapshot ID**確認
 - 実行計画を確認したいSQL文の**Hash Value**を確認
 2. スクリプトの実行
 - SQL> @\$ORACLE_HOME/rdbms/admin/sprepsql.sql
 - 実行中に、**Snapshot ID**, **Hash Value**を入力
 3. カレントフォルダにレポートが出力される

実行計画取得手順

```
SQL> @$ORACLE_HOME/rdbms/admin/sprepsql.sql
```

～中略～

Instance	DB Name	Snap Id	Snap Started	Snap Level	Comment
ora9204a	ORA9204	126	11 May 2004 19:44	6	
		127	11 May 2004 19:45	6	

Specify the Begin and End Snapshot Ids

~~~~~

begin\_snapに値を入力してください: 126

Begin Snapshot Id specified: 126

end\_snapに値を入力してください: 127

End Snapshot Id specified: 127

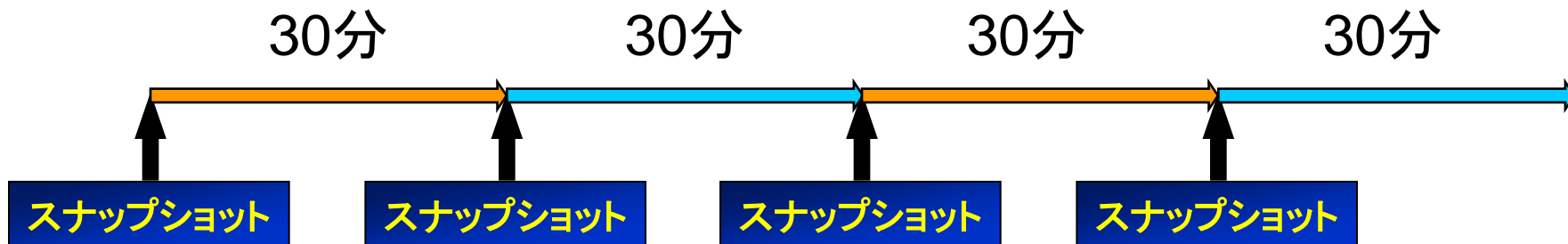
Specify the Hash Value

~~~~~

hash_valueに値を入力してください: 3328979786

Hash Value specified is: 3328979786

スナップショットをとるタイミング



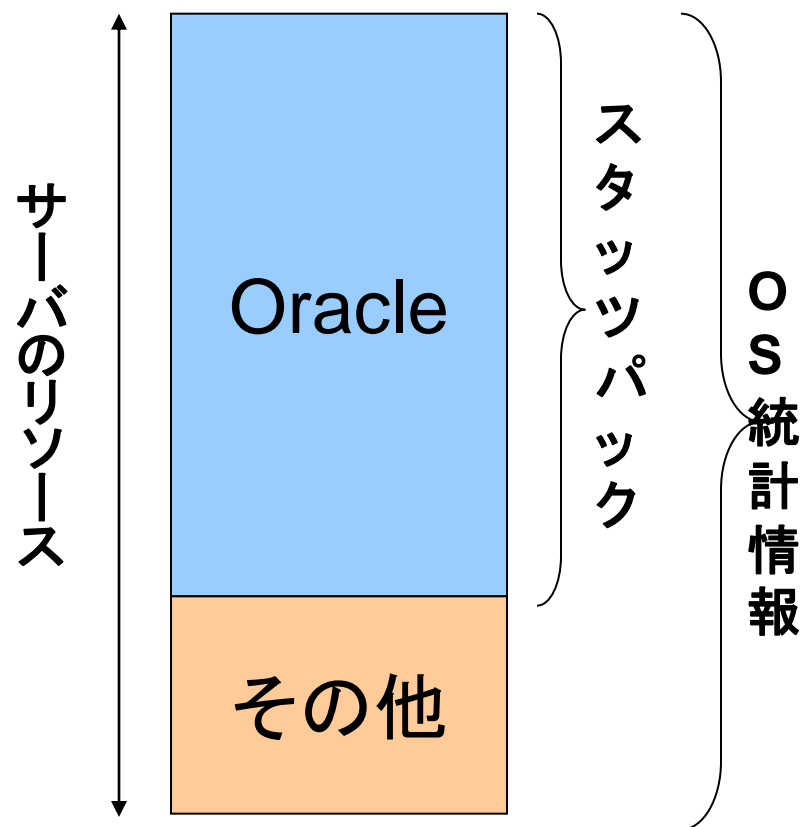
- 30分間隔ぐらいで常時取得がのぞましい
- 保存期間を決め古くなったものは定期的に削除

※ご参考 AWRの場合

- デフォルトで60分間隔で常時取得(変更可)
- 保存期間はデフォルトで7日間(変更可)

Statspack以外に取得すべき情報

- OS統計情報
 - CPU統計情報
 - ディスク統計情報
 - メモリー統計情報
 - ネットワーク統計情報



10gR2～ OS統計も一部表示される

```
Host Name: keabe-jp1 Num CPUs: 4 Phys Memory (MB):3,071
~中略~
Host CPU (CPUs: 4)
~~~~~
Load Average
Begin End User System Idle WIO WCPU
-----
80.05 7.67 12.28

Instance CPU
~~~~~
% of total CPU for Instance: 65.99
% of busy CPU for Instance: 75.23
%DB time waiting for CPU - Resource Mgr:

Memory Statistics
~~~~~
Begin End
-----
Host Mem (MB): 3,070.9 3,070.9
SGA use (MB): 400.0 400.0
PGA use (MB): 61.3 73.1
% Host Mem used for SGA+PGA: 15.0 15.4
```

Statspack解析前の心得

- Statspackはある期間内のサマリー情報である
 - 値は合計値と平均値であり、取得できる情報が決まっているので、瞬間風速的な遅延や詳細のトレースは難しい
 - Statspackを見て判断が難しい場合は、瞬間風速的な遅延が発生している可能性があるのでv\$sessionやv\$session_waitで探る
 - Statspackで大まかな原因を特定して詳細に落とす場合は、別途パフォーマンス・ビューなどで確認する必要がある

Statspack解析前の心得

- チューニング効果の一般論
 1. ビジネス・ルールのチューニング
 2. データ設計のチューニング
 3. アプリケーション設計のチューニング
 4. データベースの論理構造のチューニング
 5. データベース操作のチューニング
 6. アクセス・パスのチューニング
 7. メモリー割当てのチューニング
 8. I/O および物理構造のチューニング
 9. リソースの競合のチューニング
 10. 基礎を形成するプラットフォームのチューニング

SQL文や論理、物理設計
の改善の方が効果が高い

StatspackのSQL統計は必ずチェックすべし

アジェンダ

- Statspackとは？
- Statspackの解析方法
- データベース
自動チューニング機能

Oracle Directの無償技術サービス

- SQL Serverからの移行アセスメント
- MySQLからの移行相談
- PostgreSQLからの移行相談
- Accessからの移行アセスメント
- Application Server 移行相談
- Oracle Database バージョンアップ支援
- Oracle Developer/2000 Webアップグレード相談
- パフォーマンス・クリニック
- Oracle Database 構成相談
- Oracle Database 高可用性診断
- システム連携アセスメント

<http://www.oracle.com/lang/jp/direct/services.html>

Statspackの解析方法

1. セッション数チェック
2. アプリケーションの傾向をざっくり確認
 - Load Profileのチェック

どういう使われ方??

3. インスタンスの効率をチェック
 - Instance Efficiency Percentageのチェック
4. Top 5 Wait Event
 - 待機イベントとは?
 - 主要な待機イベントとドリルダウン
5. Memory Advisoryをチェック
 1. Buffer Pool Advisory
 2. PGA Memory Advisory etc.
6. 実行されているSQL文をチェック

DB内で行われている
処理は何??
手をつけるポイントは
どこ??

1.セッション数チェック

	Snap Id	Snap Time	Sessions	Curs/Sess
Begin Snap:	502	26-4月 -05 20:13:28	107	1,335.4
End Snap:	503	26-4月 -05 20:35:28	106	1,352.7
Elapsed:		22.00 (mins)		
DB Time:		37.65 (mins)		

- スナップショット取得時のセッション数を確認する
 - セッションが増えるとDBが使用する時間が増えるため
 - スナップショットを取るタイミングによって変わってくるのであくまで参考のため確認する

2.アプリケーションの傾向をざっくり確認

Load Profile

~~~~~

|                         | Per Second | Per Transaction |
|-------------------------|------------|-----------------|
|                         | -----      | -----           |
| Redo size:              | 321,514.13 | 5,605.82        |
| <b>Logical reads:</b>   | 29,800.08  | <b>519.59</b>   |
| Block changes:          | 1,926.31   | 33.59           |
| <b>Physical reads:</b>  | 117.19     | <b>2.04</b>     |
| <b>Physical writes:</b> | 137.81     | <b>2.40</b>     |
| User calls:             | 1,488.21   | 25.95           |
| Parses:                 | 439.28     | 7.66            |
| <b>Hard parses:</b>     | 0.00       | <b>0.00</b>     |
| Sorts:                  | 172.12     | 3.00            |
| Logons:                 | 0.03       | 0.00            |
| Executes:               | 729.22     | 12.71           |
| Transactions:           | 57.35      |                 |

- トランザクションあたりの統計を確認
  - 平常時にも取得しておき、ベースラインとして保存しておく
  - 平均値だが、同一業務であれば異変に気づきやすい

### 3. インスタンス効率をチェック

Instance Efficiency Percentages (Target 100%)

~~~~~

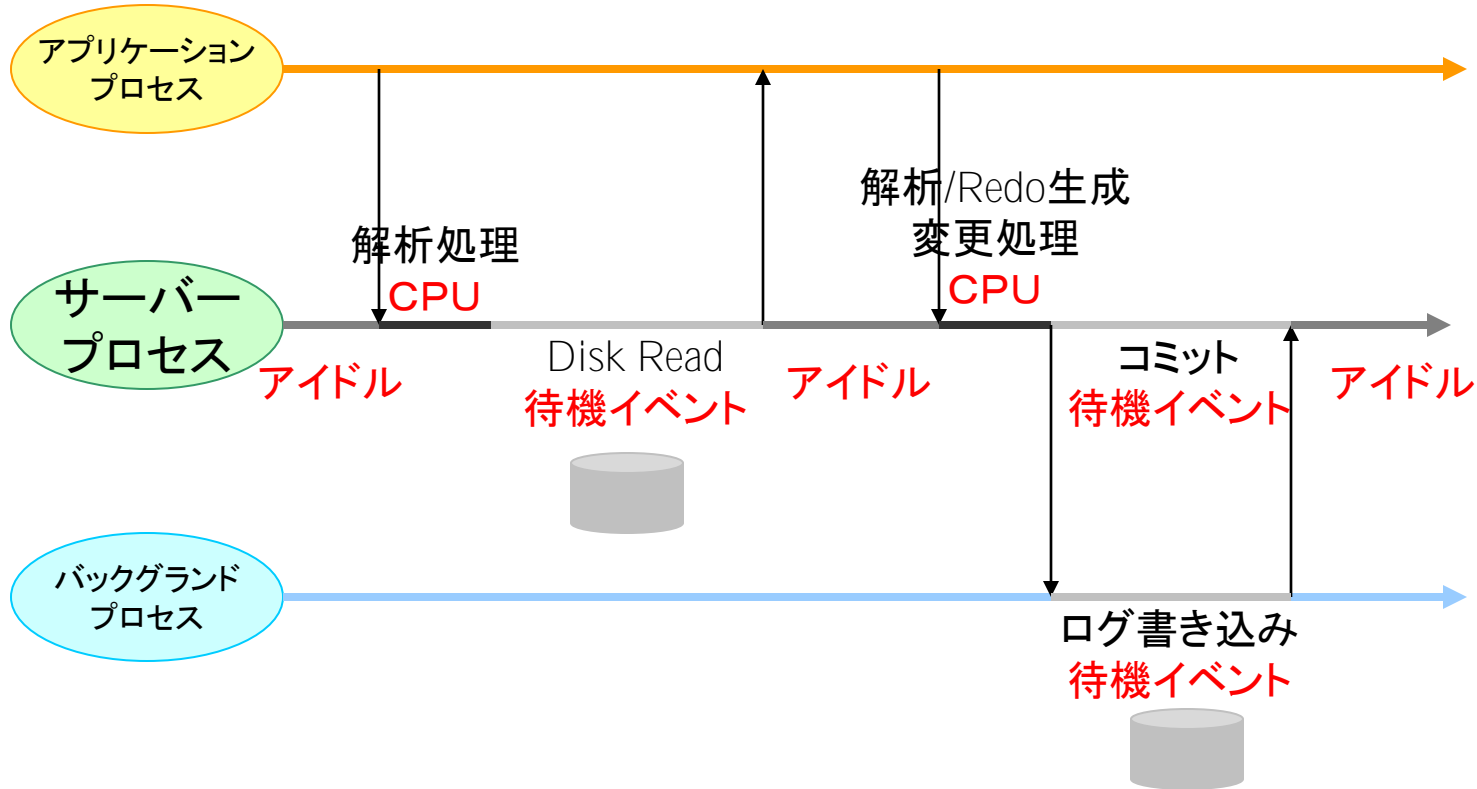
Buffer Nowait %:	100.00	Redo NoWait %:	100.00
Buffer Hit %:	99.61	In-memory Sort %:	100.00
Library Hit %:	99.87	Soft Parse %:	100.00
Execute to Parse %:	39.76	Latch Hit %:	99.62
Parse CPU to Parse Elapsed %:	80.98	% Non-Parse CPU:	95.09

- 全ての値を100%に近づけることが目標
 - 80%以上・・・ %Non-Parse CPU
 - 90%以上・・・ Buffer Hit%, In-memory Sort%, Soft Parse%
 - 95%以上・・・ Library Hit%, Redo Nowait%, Buffer Nowait%
 - 98%以上・・・ Latch Hit%

各値の解説

	解説
Buffer Hit%	必要なデータがバッファ上にあった割合
Library Hit %	必要なSQL、PL/SQLがライブラリ・キャッシュにあった割合
Soft Parse %	全ての解析のうち再利用可能なものの割合
In-Memory Sort%	ソートがメモリ内で行われた割合
Latch Hit%	全てのラッチのヒット率
Parse CPU to Parse Elapsed %	解析CPU時間/ 解析の合計時間
Execute to Parse%	SQL実行に対し解析が行われなかった割合
%non-parse CPU	解析以外で使されたCPU時間の割合
Buffer Nowait%	バッファに要求を出したときに、即座に使用可能だった割合
Redo Nowait%	redo logに要求を出したときに、即座に使用可能だった割合

4.待機イベント



待機イベントとCPU使用時間をチューニングすることでレスポンスを早くすることができる

待機イベント

- プロセスがCPUを使用していない時間
 - アイドル待機イベント
 - ボトルネックが存在する場合に、原因がデータベースリソースではないことを意味します
 - その他の待機イベント
 - データベースリソース(バッファ競合、I/O競合、ラッチ競合など)に関連する待機時間

トップ5待機イベント (Oracle10gR2,11g)

Top 5 Timed Events

~~~~~

Event

CPU timeより上位の  
イベントは??

Waits Time (s) Avg wait (ms) %Total Call Time

CPU time

SQL\*Net more data to client

db file sequential read

SQL\*Net message from dblink

control file sequential read

|           |       |       |      |      |
|-----------|-------|-------|------|------|
|           |       | 5,899 |      | 57.1 |
| 1,035,190 | 1,390 | 1     | 13.5 |      |
| 80,242    | 1,105 | 14    | 10.7 |      |
| 89,050    | 777   | 9     | 7.5  |      |
| 162,720   | 266   | 2     | 2.6  |      |

- ・待機イベントのうち、**総待ち時間(Wait Time)**の長い順にトップ5までを表示
- ・**上位にきているイベント**=チューニングをして効果が大きいイベント=**手をつけるべきチューニングポイント**

Waits : イベントのために待機した合計回数

Time(s) : イベントの合計待機時間および合計CPU時間(秒)

Avg wait(ms) : イベントの平均待機時間

% Total Call Time:

total call time = total CPU time + total wait time for non-idle events

% Total Call Time = time for each timed event / total call time

# 待機イベント: 良くある質問

Q. 待機イベントの値がどれくらいだと、ボトルネックと判断するのでしょうか？

基準となる値、目安値はありますか？

A. ボトルネックと判断する基準値はありません。  
CPU Timeよりも上位に来たイベントが、ボトルネックポイントとして解析を進めます。また、目安値というものも特にはないですが、いつもと比べて急激な変化があったという場合は注意してください。

| Top 5 Timed Events           |         |          | Avg  | %Total |
|------------------------------|---------|----------|------|--------|
| ~~~~~                        |         |          | wait | Call   |
| Event                        | Waits   | Time (s) | (ms) | Time   |
| -----                        |         |          |      |        |
| db file scattered read       | 876,341 | 910      | 20   | 50.2   |
| db file sequential read      | 624,391 | 724      | 16   | 35.7   |
| CPU time                     |         | 11       |      | 10.2   |
| control file sequential read | 229     | 5        | 20   | 3.1    |
| log file parallel write      | 8       | 0        | 6    | .3     |

db file scattered readが「XXであれば」という考え方ではなく、CPU Timeが上位にくることが望ましい。但し、CPU TimeがTOPであっても効率の悪いSQLがCPUを消費している可能性もある。



# 主要な待機イベント

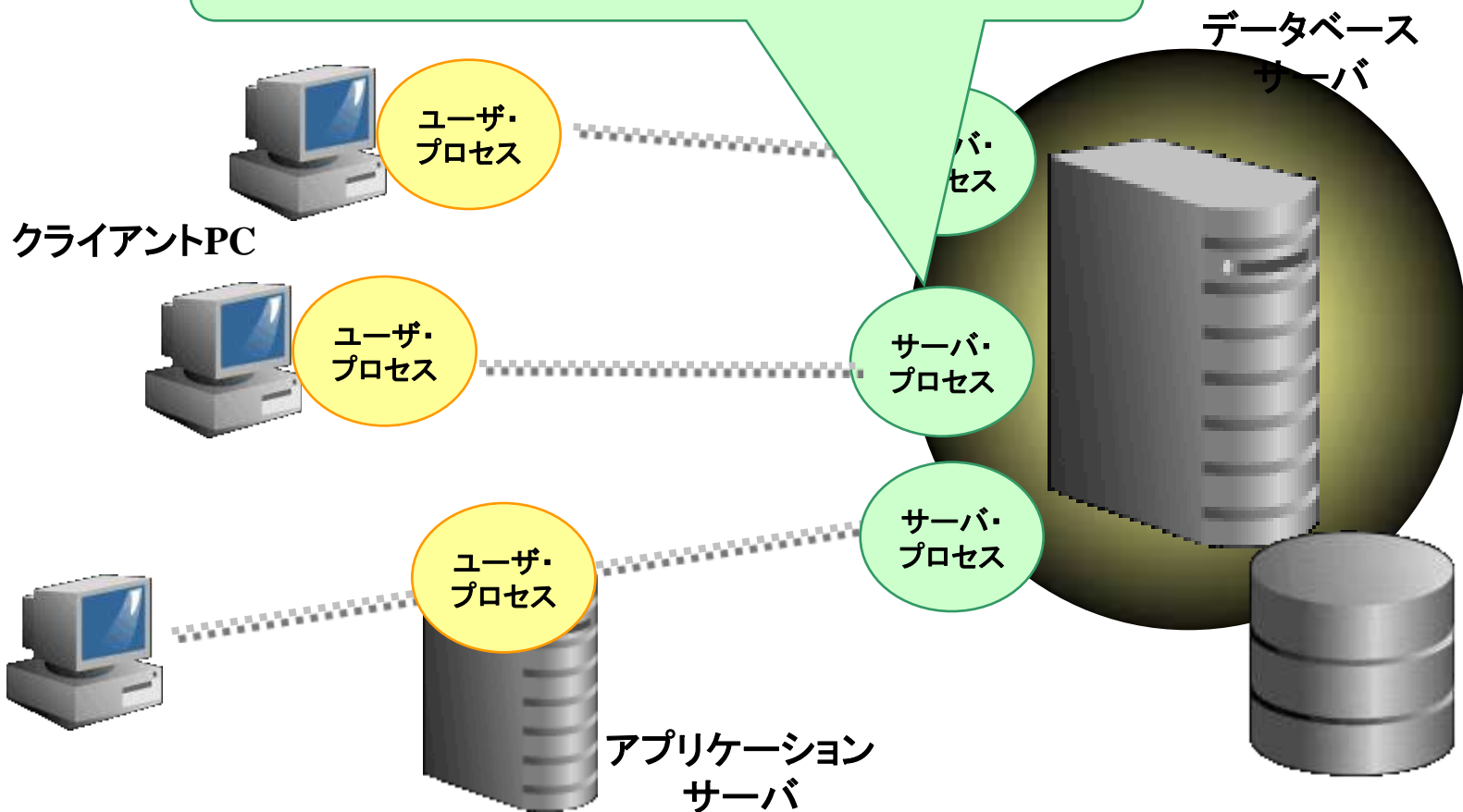
- db file sequential read
- db file scattered read
- buffer busy waits
- free buffer waits
- db file parallel writes
- log file sync
- log file parallel write
- log buffer space
- enqueue
- latch free
- アイドル待機イベント

巻末資料にて  
詳細ご紹介します

# アイドル待機イベント

待機イベント : SQL\*Net message from client

クライアントからの処理要求待ち



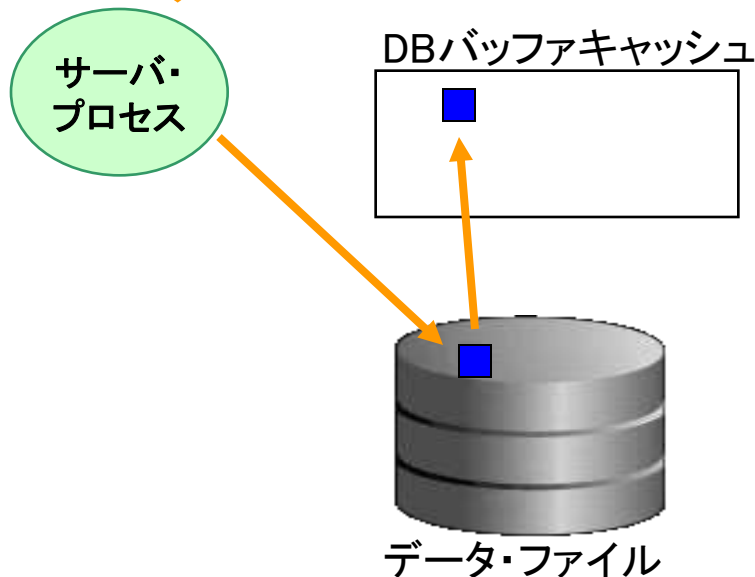
# db file sequential read

## db file scattered read

### 待機イベント:

#### • db file sequential read

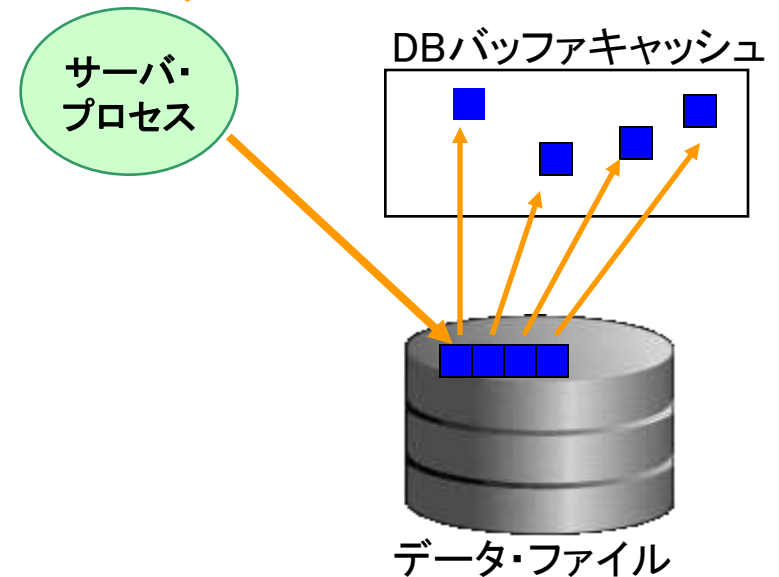
単一ブロック読み込み  
(インデックス検索)



### 待機イベント:

#### • db file scattered read

マルチブロック読み込み  
(全表検索、索引高速スキャン)



# この2つが待機イベントの上位にきたら

- バッファキャッシュヒット率 (buffer hit %)を確認
  - メモリに余裕があればサイズを大きくすることを検討する
- I/Oネックの可能性があるのでOS統計も確認
- db file scattered read
  - 統計・・・table scans (long tables)を確認
  - SQL ordered by GetsからBuffer Getsが多いSQL文を確認
  - SQL文のチューニング
- db file sequential read
  - SQL ordered by GetsからBuffer Getsが多いSQL文を確認
  - SQL文のチューニングができれば行う

# Disk I/Oがどこで発生しているのか？

File IO Stats DB/Inst: ORCL01/ORCL01 Snaps: 766-768

Tablespace                      Filename

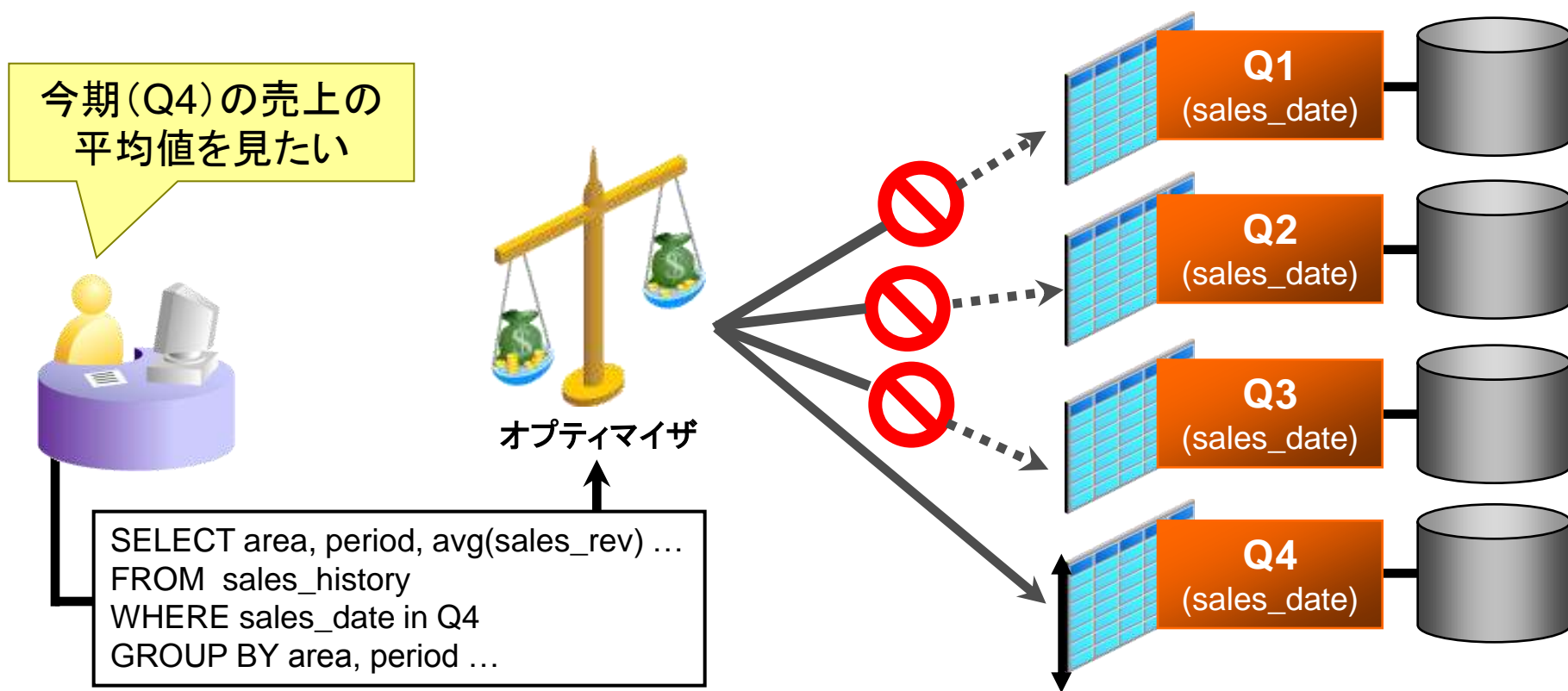
|  |  | Av      | Mx     |  |  |  |  | Av     |  |  |
|--|--|---------|--------|--|--|--|--|--------|--|--|
|  |  | Rd      | Rd     |  |  |  |  | Buffer |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)   |  |  |  |  | Waits  |  |  |
|  |  | Reads/s | (ms)</ |  |  |  |  |        |  |  |

# 考えられる解決策: DiskI/Oを減らせるように

- バッファヒット率が低ければメモリ割り当ての拡張
- SQLが効率よくできるように
  - 索引の作成
  - 意図した索引が使用されているかの確認  
(ヒント句、SQLプロファイル)
- データを分割して管理する
  - Partitioningの利用で、SQL変更なくデータを分割
  - 表領域の割り当て変更

# <参考>データを分割して管理する：パーティション

- 対象のデータが格納されているパーティションだけにアクセスし、不要なパーティションを読み飛ばす



# 5.Advisory Statistics

PGA Memory Advisory for DB: AAA Instance: aaa End Snap: 20555

| PGA Target<br>Est (MB) | Size<br>Factr | W/A MB<br>Processed | W/A MB Read/<br>Written to Disk | Cache<br>Hit % | Overalloc<br>Count |
|------------------------|---------------|---------------------|---------------------------------|----------------|--------------------|
| 16                     | 0.1           | 13,406,708.5        | 1,150,524.0                     | 92.0           | 98,500             |
| 32                     | 0.3           | 13,406,708.5        | 1,149,545.5                     | 92.0           | 98,500             |
| 64                     | 0.5           | 13,406,708.5        | 1,149,545.5                     | 92.0           | 98,500             |
| 96                     | 0.8           | 13,406,708.5        | 1,149,545.5                     | 92.0           | 98,500             |
| 128                    | 1.0           | 13,406,708.5        | 370,864.9                       | 97.0           | 98,343             |
| 154                    | 1.2           | 13,406,708.5        | 358,442.9                       | 97.0           | 73,884             |
| 179                    | 1.4           | 13,406,708.5        | 345,671.0                       | 97.0           | 51,419             |
| 205                    | 1.6           | 13,406,708.5        | 325,909.7                       | 98.0           | 34,441             |
| 230                    | 1.8           | 13,406,708.5        | 208,594.9                       | 98.0           | 8,993              |
| 256                    | 2.0           | 13,406,708.5        | 158,403.9                       | 99.0           | 4,272              |
| 384                    | 3.0           | 13,406,708.5        | 105,314.7                       | 99.0           | 826                |
| 512                    | 4.0           | 13,406,708.5        | 99,935.0                        | 99.0           | 176                |
| 768                    | 6.0           | 13,406,708.5        | 98,714.6                        | 99.0           | 22                 |
| 1,024                  | 8.0           | 13,406,708.5        | 98,433.7                        | 99.0           | 0                  |

ディスクアクセス量の見積もり値

一般的にはこの列の数値が小さい  
つまりディスクアクセスが低い方が  
パフォーマンス上は好ましい

キャッシュヒット率

PGA Memory Advisory以外の  
Advisory Statisticsもチェックする  
例.Buffer Pool Advisoryなど

現在の設定

## • 最適なメモリの割り当て量を確認

- PGA\_AGGREGATE\_TARGET パラメータを増減させた場合の、ディスクアクセス量の見積もり値の変化を確認



# 5.実行されているSQL文をチェック

物理読み込みブロック数(Physical Reads)の

多い順にSQL文をリスト

"SQL ordered by Reads"

~~~~~

Physical Reads	Executions	Reads per Exec	% Total	Hash Value
----------------	------------	----------------	---------	------------

19,649,016	1,732	11,344.7	60.9	797383422
------------	-------	----------	------	-----------

```
SELECT "CALL_NUMBER","SUPPMGR","STAT_CODE" FROM "CALLS" C WHERE  
("STAT_CODE"='NEW' OR "STAT_CODE"='CFU' OR "STAT_CODE"='1CB' OR  
"STAT_CODE"='2CB' OR "STAT_CODE"='IRR') AND "SUPPMGR"=:1||'.JP'
```

5,377,470	473	11,368.9	16.7	3848244512
-----------	-----	----------	------	------------

```
SELECT COUNT(DECODE(A1."STAT_CODE",'HCL',NULL,'SCL',NULL,'1MN',N  
ULL,1)) FROM "CALLS" A1 WHERE A1."SUPPMGR"=:B1
```

1,220,850	110	11,098.6	3.8	1212026352
-----------	-----	----------	-----	------------

```
SELECT "CALL_NUMBER","LAST_UPDATED","SUPPMGR","SUMMARY","STAT_CO  
DE" FROM "CALLS" C WHERE "STAT_CODE"<>'SCL' AND "STAT_CODE"<>'HC  
L' AND "STAT_CODE"<>'ASG' AND "SUPPMGR"=:1||'.JP'
```

表示される列:

Physical Reads(物理読み込み)

Executions(実行回数)

Reads per Exec(実行ごとの読み込み)

%Total(合計%)

Hash Value(ハッシュ値)

リソース消費の多いSQL文

・「Physical Reads」、「%Total」の割合が大きい

- 「SQL ordered by ~」の項目でリソース消費の多いSQL文を確認
 - リソース消費の多いSQLの実行計画を確認してチューニングできる箇所があるかを確認
 - 「SQL ordered by read」以外にも「SQL ordered by CPU」、「SQL ordered by get」等も確認する

アジェンダ

- Statspackとは？
- Statspackの解析方法
- データベース
自動チューニング機能

Oracle Directの無償技術サービス

- SQL Serverからの移行アセスメント
- MySQLからの移行相談
- PostgreSQLからの移行相談
- Accessからの移行アセスメント
- Application Server 移行相談
- Oracle Database バージョンアップ支援
- Oracle Developer/2000 Webアップグレード相談
- パフォーマンス・クリニック
- Oracle Database 構成相談
- Oracle Database 高可用性診断
- システム連携アセスメント

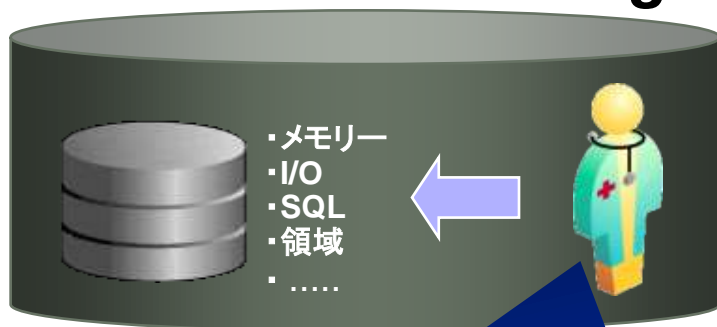
<http://www.oracle.com/lang/jp/direct/services.html>

Oracleの機能を用いた監視 / 管理



- 10gからは**ボトルネックの診断**から**問題解決のアドバイス**までを**自動化**する機能が追加されています。

Oracle Database 10g～



Database Control



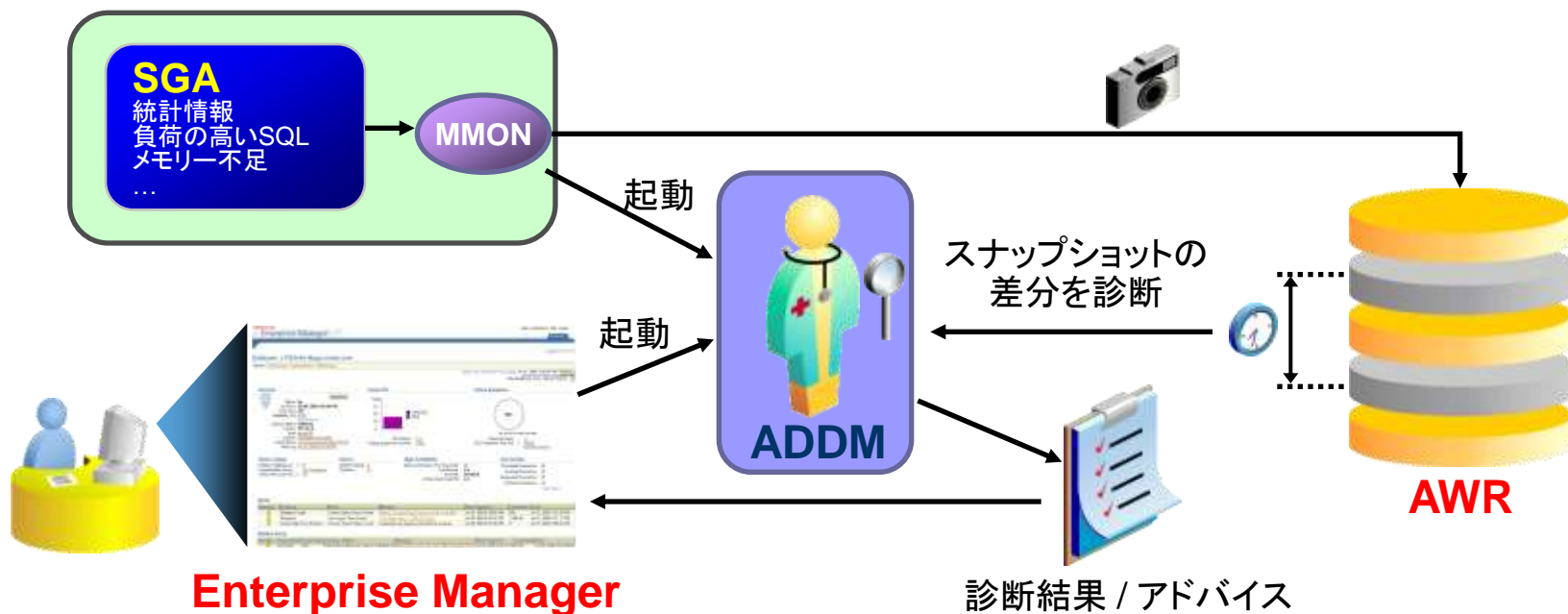
DBAはOracleのアドバイスを
受け入れるかを判断



Oracle自身がデータベースを診断/監視

Automatic Workload Repository(AWR)&Automatic Database Diagnostic Monitor(ADDM)

- 自動的に収集されたデータを定期的に分析し、データベースのパフォーマンスを監視 / 診断する

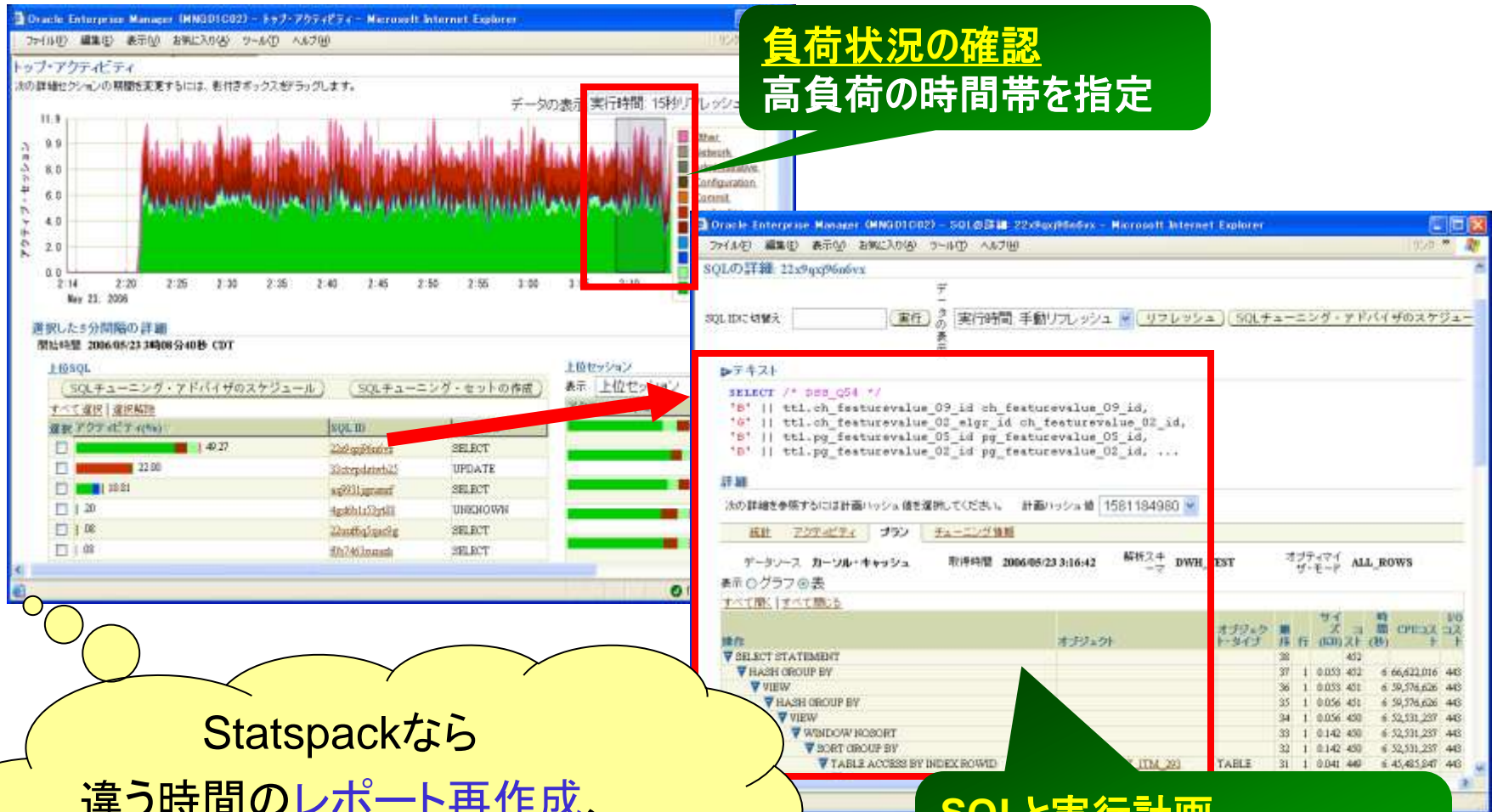


DB稼動状況分析方法 AWRとStatspack

- AWRでスナップショットを自動収集していればSTATSPACKを使用する必要はほとんどありません。

	AWR	Statspack
対応バージョン	10.1.0～	8.1.6～
ライセンス	Enterprise Edition+Diagnostic Pack Option	Standard Edition One～
スナップショット取得	自動/手動	手動
レポート形式	HTML/TEXT	TEXTのみ
レポート解析	ADDMが実行	なし
操作	GUI/コマンド	コマンドのみ

例：負荷の高いSQL→実行計画もワンクリック



Statspackなら

違う時間のレポート再作成、
SQL全文と実行計画表示
のためのレポート再作成

SQLと実行計画

SQL文と実行計画(実行手順、アクセスパス)が表示

ORACLE

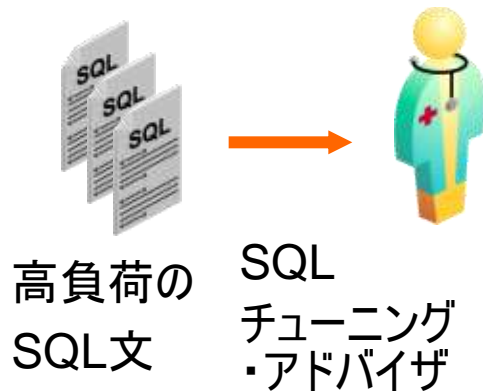
SQLチューニング・アドバイザーによる 自動チューニング

EE

Diag

Tun

- Oracle Database 10gから実装されたアドバイス機能
- 高負荷で問題となるSQL文や実行計画を診断し、アドバイスを提示
 - 統計の再取得
 - SQL文の問題点を探し、SQL文の修正方法
 - 必要な索引の作成をアドバイス
 - SQLプロファイルの作成



Enterprise Managerが
負荷を軽減する最適な対
処方法を提示

失効・欠落している
統計の収集



Index の作成



SQL文の
再構成



SQLプロファイル
の作成



ORACLE

SQLチューニング・アドバイザーの実行例

EE

Diag

Tun

推奨の選択					ベネフィット(%)	新規実行計画	実行計画の比較
元の実行計画(注釈付き)							
実装							
選択	タイプ	結果	推奨	論理			
<input type="radio"/>	統計	表"SH", "SALES"の最適化統計は失効しています。	この表およびその索引に対する最適化統計の収集を検討してください。	適切な実行計画を選択するには、表およびその索引の最新の最適化統計が必要です。			
<input checked="" type="radio"/>	SQLプロファイル	この文により適している可能性のある実行計画が見つかりました。	推奨されるSQLプロファイルの承認を検討してください。		99.36	👁	👁
<input type="radio"/>	索引	索引を1つ以上作成すると、この文の実行計画を改善できます。	物理スキーマ設計を改善するAccess Advisorの実行が、推奨される索引の作成を検討してください。	推奨される索引を作成すると、この文の実行計画が大きく改善されます。ただし、単一の文ではなく代理SQLワークロードを使用した"Access Advisor"の実行が適切な場合があります。この処理により、索引メンテナンス・オーバーヘッド	66.74	👁	👁

実行計画の比較

元の実行計画(注釈付き)

- SQLチューニング・アドバイザーによる元のプランからの調整を示します
計画ハッシュ値 **4005616876**

すべて開く | すべて閉じる

操作	ラインID	オブジェクト	オブジェクト・タイプ	順序	行	バイト	コスト	時間	CPUコスト	I/Oコスト
SELECT STATEMENT	0			6		0.013	9,398	113	1,895,178,624	9,312
HASH GROUP BY	1			5		0.013	9,398	113	1,895,178,624	9,312
NESTED LOOPS	2			4		0.013	9,397	113	1,873,027,712	9,312
TABLE ACCESS FULL	3	SH.CUSTOMERS	TABLE	1		0.005	405	5	21,682,460	404
PARTITION RANGE ALL	4			3		0.008	8,992	108	1,851,345,152	8,908
TABLE ACCESS FULL	5	SH.SALES	TABLE	2		0.008	8,992	108	1,851,345,152	8,908

SQLプロファイルのある新しい実行計画

計画ハッシュ値 **1458810583**

すべて開く | すべて閉じる

操作	ラインID	オブジェクト	オブジェクト・タイプ	順序	行	バイト	コスト	時間	CPUコスト	I/Oコスト
SELECT STATEMENT	0			6		0.013	5	1	22,182,248	4
HASH GROUP BY	1			5		0.013	5	1	22,182,248	4
NESTED LOOPS	2			4		0.013	4	1	31,336	4
TABLE ACCESS BY GLOBAL INDEX ROWID	3	SH.SALES	TABLE	2		0.008	4	1	29,386	4
INDEX RANGE SCAN	4	SH.SALES_CUST_ID_IDX	INDEX	1		3	1	21,764	3	
INDEX UNIQUE SCAN	5	SH.CUSTOMERS_PK	INDEX (UNIQUE)	3		0.005	0	1	1,950	0

コストと時間が大幅に改善されることが分かる

ORACLE

まとめ

- Statspackとは？
 - パフォーマンス・チューニングに役立つ情報をレポートという形で提供するツール
- Statspackの解析方法
 - セッション数チェック
 - アプリケーションの傾向をざっくり確認
 - インスタンスの効率をチェック
 - Top5 待機イベントをチェック
 - Memory Advisoryをチェック
 - 実行されているSQL文をチェック
- データベース
 - 自動チューニング機能

OTN×ダイセミ でスキルアップ!!



- ・技術的な内容について疑問点を解消したい！
- ・一般的なその解決方法などを知りたい！
- ・ 세미나資料など技術コンテンツがほしい！

Oracle Technology Network(OTN)を御活用下さい。

<http://otn.oracle.co.jp/forum/index.jspa?categoryID=2>

技術的な疑問点は、OTN揭示版の
「データベース一般」へ

※OTN揭示版は、基本的にOracleユーザー有志からの回答となるため100%回答があるとは限りません。
ただ、過去の履歴を見ると、質問の大多数に関してなんらかの回答が書き込まれております。

<http://www.oracle.com/technology/global/jp/ondemand/otn-seminar/index.html>

過去の 세미나資料、動画コンテンツはOTNの
「OTNセミナー オンデマンドコンテンツ」へ

※ダイセミ事務局にダイセミ資料を請求頂いても、お受けできない可能性がございますので予めご了承ください。
ダイセミ資料はOTNコンテンツ オン デマンドか、 세미나実施時間内にダウンロード頂くようお願い致します。

ORACLE

OTNセミナー オンデマンド コンテンツ

期間限定にて、ダイセミの人気セミナーを動画配信中!!

ダイセミのライブ感はそのままに、好きな時間で受講頂けます。

最新のコンテンツ

エンジニアのためのITIL実践術 再生時間: 60分	ここからはじめよう Oracle PL/SQL入門 再生時間: 60分	実践!!高可用システム構築 -RAC基本 再生時間: 60分	お悩み解決! Oracleのサイジング 再生時間: 60分

Database

今さら聞けない!!バックアップ・リカバリ入 再生時間: 60分	意外と簡単!? Oracle Database 11g -セ 再生時間: 60分	実践!!バックアップ・リカバリ 再生時間: 60分	意外と簡単!? Oracle Database 11g -デ 再生時間: 60分

>> もっと見る

OTN オンデマンド

検索

※掲載のコンテンツ内容は予告なく変更になる可能性があります。

期間限定での配信コンテンツも含まれております。お早めにダウンロード頂くことをお勧めいたします。

ORACLE

オラクル クルクルキャンペーン

あの**Oracle Database Enterprise Edition**が超おトク!!

おトクな買い方
オラクル5年分

- ライセンス使用期間 を**5年**間に設定
- 初期のライセンスコストがなんと**67%OFF** !
- テクニカル・サポート価格も**53%OFF** !

Oracle Databaseの
ライセンス価格を**大幅に抑えて**
ご導入いただけます

多くのお客様でサーバー使用期間とされる
5年間にライセンス期間を限定

- 期間途中で永久ライセンスへ差額移行
- 5年後に新規ライセンスを購入し継続利用
- 5年後に新システムへデータを移行



Enterprise Editionはここが違う!!

- 圧倒的な**パフォーマンス**!
- データベース**管理がカンタン**!
- データベースを**止めなくていい**!
- もちろん**障害対策**も万全!

この機能でこの価格
ライセンスパック

- Oracle Databaseの機能を**存分に使える**!
- **2ノードRAC**構成も可能!
- サーバー構成によって計4種類のバックから**選べる**!

詳しくはコチラ

<http://www.oracle.co.jp/campaign/kurukuru/index.html>

Oracle Direct 0120-155-096

お問い合わせフォーム

http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=28

ORACLE

パフォーマンス・クリニック・サービス

Oracle Directが**無償で**Oracleデータベースの稼動状況をチェック。

パフォーマンスのボトルネックを診断し、チューニングすべき項目をガイドします。

? CPU... ?
? メモリ... ?
? SQL... 索引...
? ディスクI/O... ?

お客様



レスポンスが遅いと言われているが、どう手をつけていいかわからない。
システムの性能問題が多く、原因を特定するのも困難である。
対応しなくちゃいけないけど、今からデータベースのチューニング・スキルを習得するにも時間もコストもかかる。

ヒアリングシート、
Statspack送付

H/W情報、
OS、
ユーザ数
etc...

簡単なシステムのヒアリングシートと、Statspackを頂きます。

それを基に、無償でボトルネックの診断と、それに対するチューニングのガイドレポートを送付します。

Statspackの取得方法も、ご希望に応じてご紹介致します。

診断レポート
送付



Oracle Direct

サービス詳細はこちら <http://www.oracle.co.jp/direct/service/pc.html>

お問い合わせ方法

電話によるお問い合わせ: **0120-155-096**

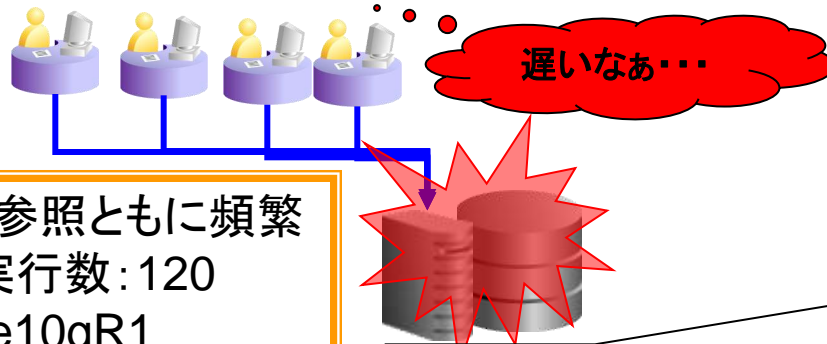
WEBによるお問い合わせ: http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=1

※ガイドさせて頂く指標が最良であることを
保証するものではありません。

※ご不明点はお気軽にお問い合わせ下さい。

本サービスに関するお問合せの場合には、その旨お伝えください。
例:「パフォーマンス診断サービスを受けたいのだけど。。。」「
無償サービスの詳細について聞きたい。」

パフォーマンス・クリニック・サービス 対応事例



- ・更新・参照ともに頻繁
- ・同時実行数: 120
- ・Oracle10gR1
- ・データ量: 40GB
- ・メモリ: 2GB
- ・CPU: 1.2GHz × 1枚



システム担当者様

利用者からレスポンスが遅いと言われている。けど、WEBサーバー・データベース・ネットワーク、社内のインフラを少ないリソースで管理しているので、手がいっぱい!! データベース内のどこに問題があるのか判別するノウハウが社内になくて...

診断結果: 問題の原因

- ・CPUボトルネックが主原因。
- ・I/Oネックもある程度発生している

診断結果: ご提案した対応方法

- 要求される負荷がCPUの処理能力を超えていると考えられるため
CPU追加orサーバ・リプレースが望ましい
- 今すぐの増強が難しい場合
I/Oネック回避のためSQLチューニングを推奨

診断結果
レポート送付

ヒアリングシート、
Statspack送付



Oracle Direct

ご希望によりStatspackの取得方法もお伝え、
無償で分析レポートを送付します。

あなたにいちばん近いオラクル



Oracle Direct

まずはお問合せください

Oracle Direct

検索

システムの検討・構築から運用まで、ITプロジェクト全般の相談窓口としてご支援いたします。

システム構成やライセンス/購入方法などお気軽にお問い合わせ下さい。

Web問い合わせフォーム

専用お問い合わせフォームにてご相談内容を承ります。

http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=28

※フォームの入力には、Oracle Direct Seminar申込時と同じ
ログインが必要となります。

※こちらから詳細確認のお電話を差し上げる場合がありますので、ご登録されている連絡先が最新のものになっているか、ご確認下さい。

フリーダイヤル

0120-155-096

※月曜～金曜 9:00～12:00、13:00～18:00

(祝日および年末年始除く)

ORACLE

巻末資料：主な待機イベントと対応方法

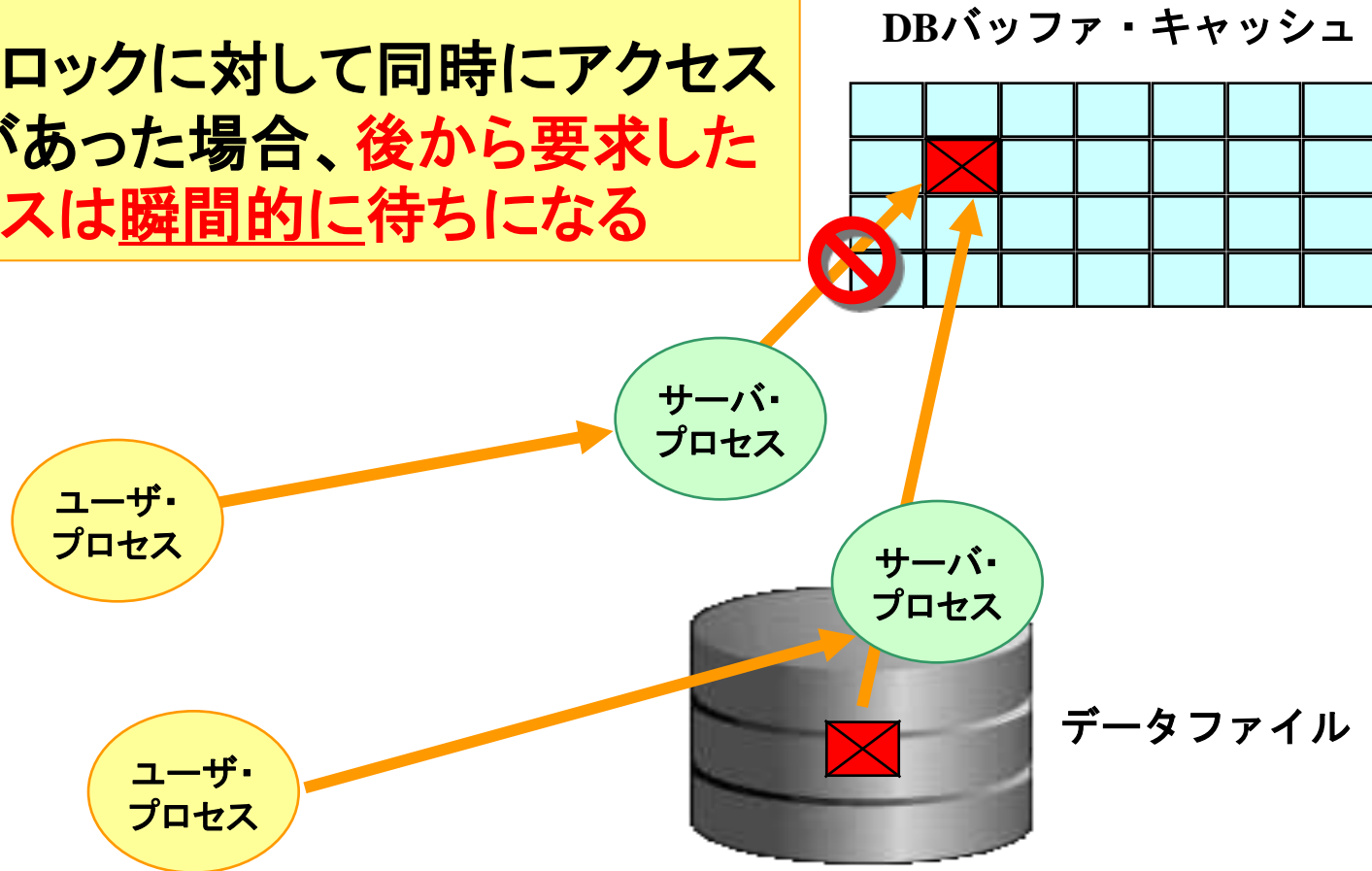
主要な待機イベント:

- (db file sequential read) セミナ内にて解説
- (db file scattered read) セミナ内にて解説
- buffer busy waits
- free buffer waits
- db file parallel writes
- log file sync
- log file parallel write
- log buffer space
- enqueue
- latch free
- (アイドル待機イベント)

buffer busy waits

待機イベント : buffer busy waits

同じブロックに対して同時にアクセス
要求があった場合、**後から要求した
プロセスは瞬間的に待ちになる**



この待機イベントが上位にきたら

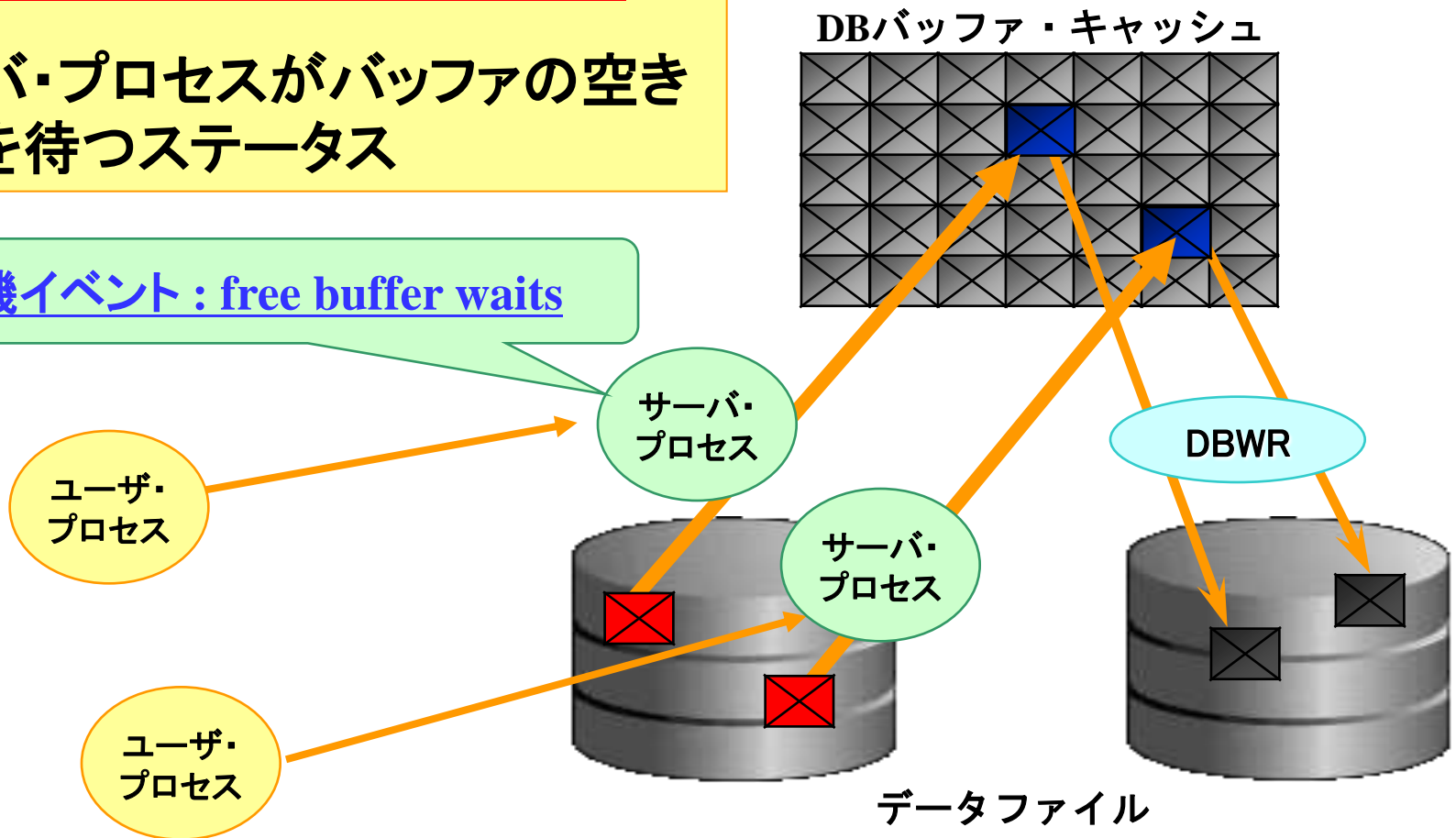
- 特定のセグメント、ブロックがホットになっているので
 - Segments by Buffer Busy Waitsからホットになっているオブジェクトを特定します(レベル7以上)
 - セグメントヘッダー
 - 空きリストの競合が主な原因です。ASSM(自動セグメント領域管理)を利用します
 - データブロック
 - 昇順インデックスが原因であれば逆キー索引やパーティションなどでホットブロックを分散させます
 - ASSMで複数のブロックが同じブロックに挿入するのを防ぎます
 - Pctfreeを調節して1ブロックに格納する行数を減らします
 - Undoヘッダー、Undoブロック
 - 自動Undo管理を利用します

free buffer waits

待機イベント : free buffer waits

サーバ・プロセスがバッファの空き領域を待つステータス

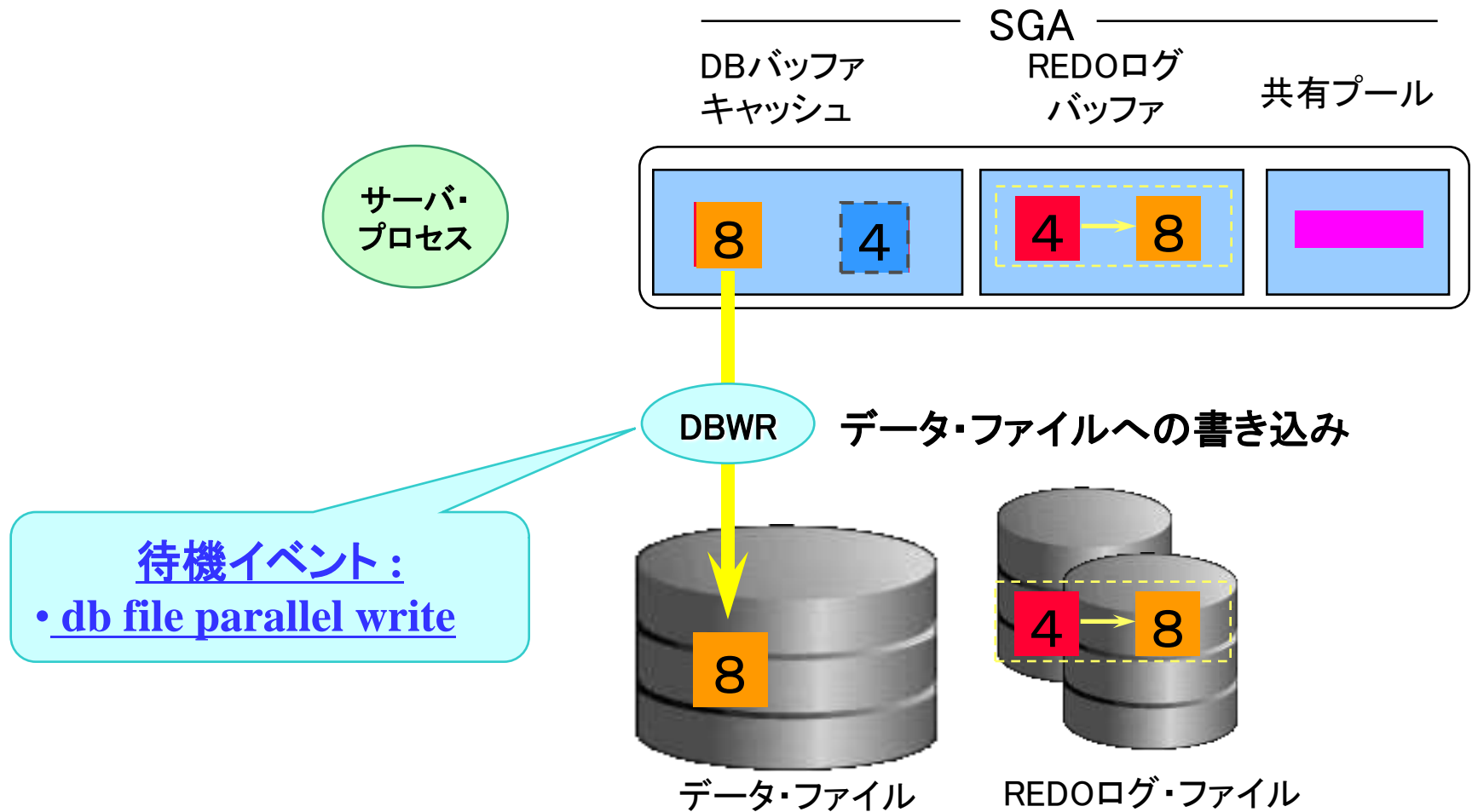
待機イベント : free buffer waits



この待機イベントが上位にきたら

- DBWRの書き込み遅延が原因の可能性あり
 - File IO Statsから書き込みが多いデータファイルを確認
 - OS統計から書き込み時間が妥当かを確認
 - 遅い場合は、I/Oのチューニングが必要
- キャッシュサイズが小さい可能性あり
 - Buffer Hit %が低い場合はサイズをおおきめに
- DBWR関連の設定問題
 - チェックポイントの頻度を増やす
 - DBWRの数がキャッシュのサイズのわりには少ない

db file parallel write

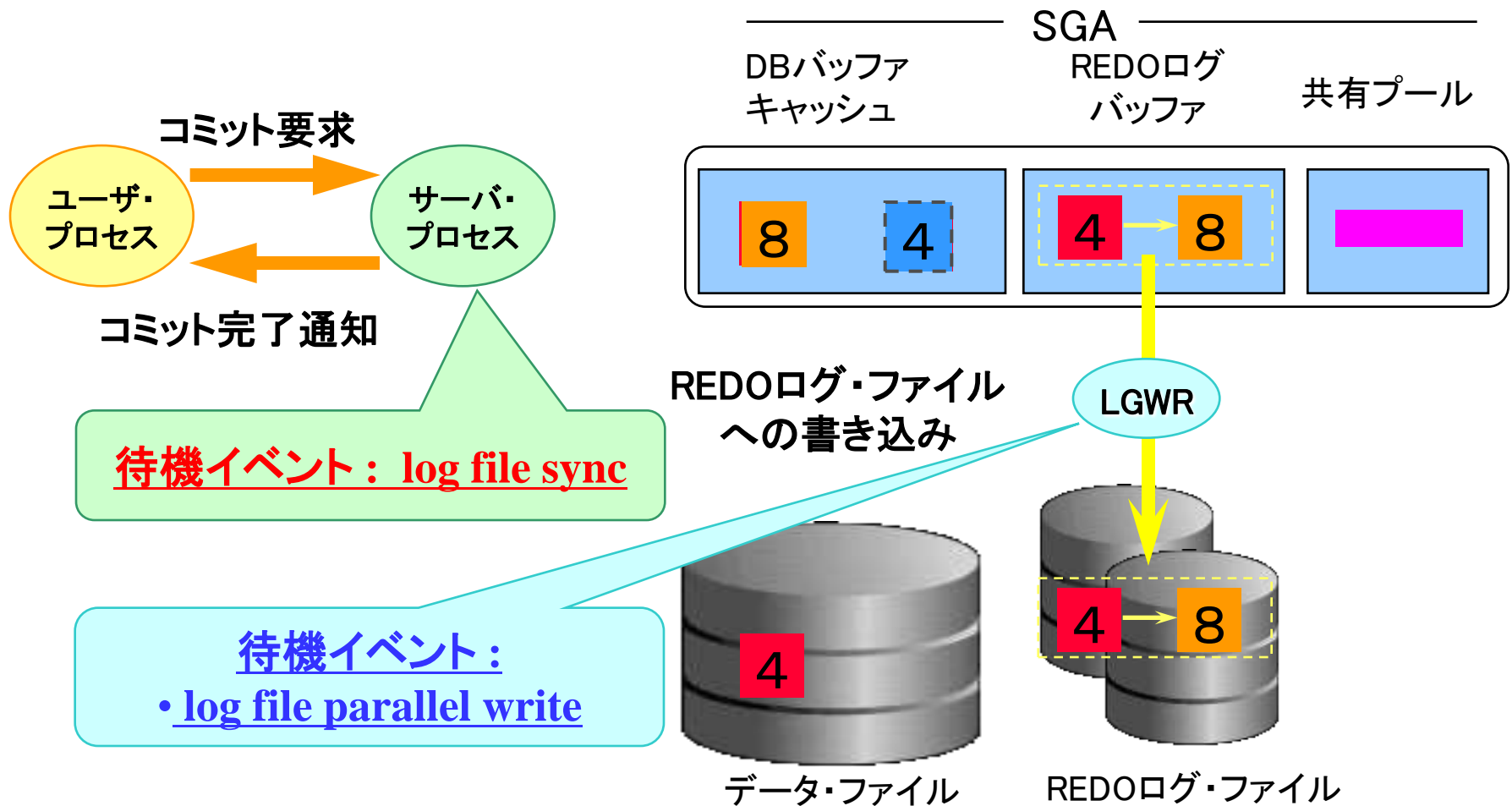


この待機イベントが上位にきたら

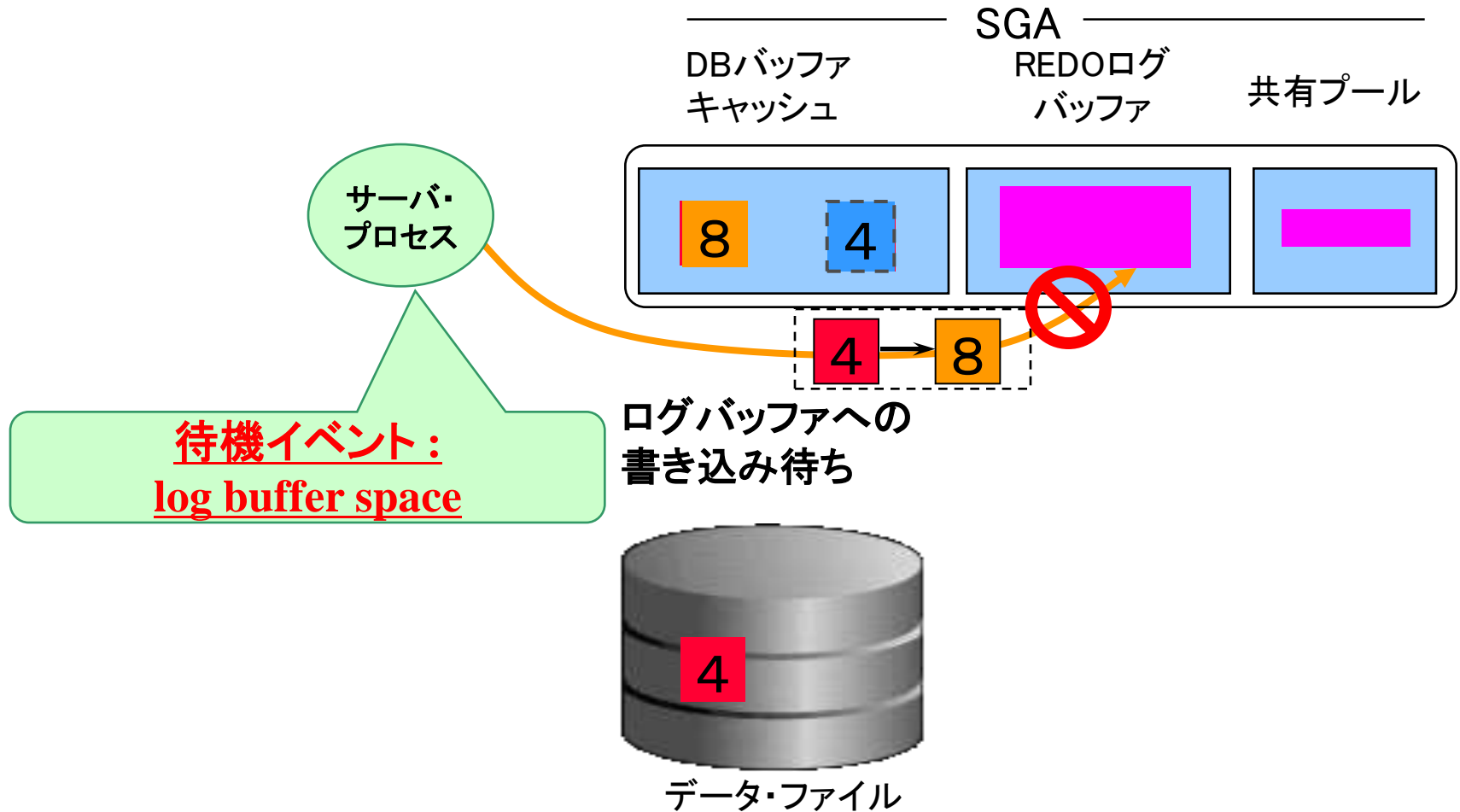
- DBWRの書き込み遅延が原因の可能性あり
 - File IO Statsから書き込みが多いデータファイルを確認
 - OS統計から書き込み時間が妥当かを確認
 - 遅い場合は、I/Oのチューニングが必要

log file sync

log file parallel write



log buffer space



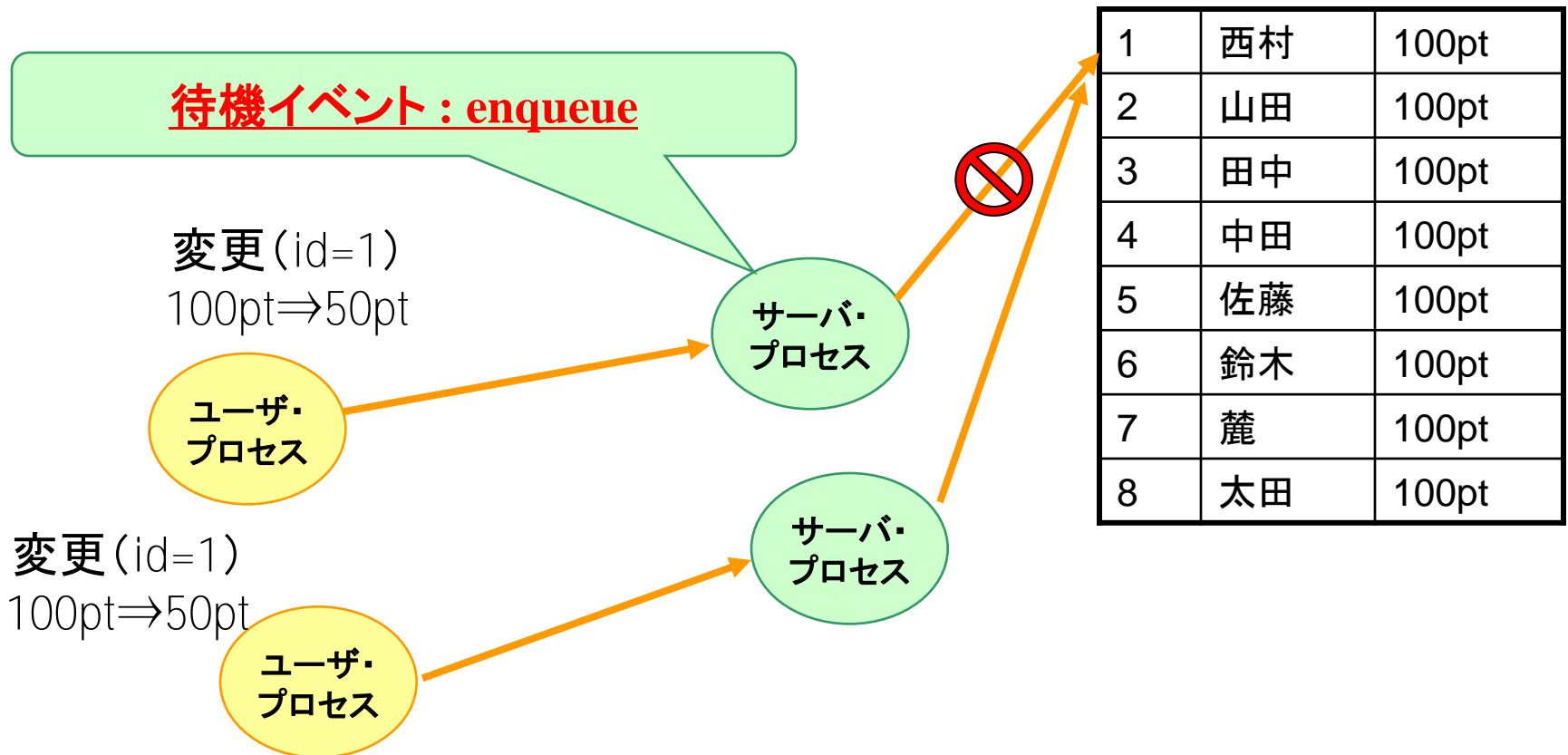
この待機イベントが上位にきたら

- コミットやロールバックに伴うLGWRの書き込み遅延が原因
 - OS統計から書き込み時間が妥当かを確認
 - 遅い場合は、I/Oのチューニングが必要
 - バッチ処理などでコミットのタイミングを変更できる場合は変更する
 - 待機時間が短く回数が多い場合・・・コミット頻度をさげる
- log buffer spaceの場合は、log bufferサイズが適切でない可能性がある

enqueue

共有リソースへアクセスするためのロックングメカニズム

- 例) Txエンキュー



この待機イベントが上位にきたら

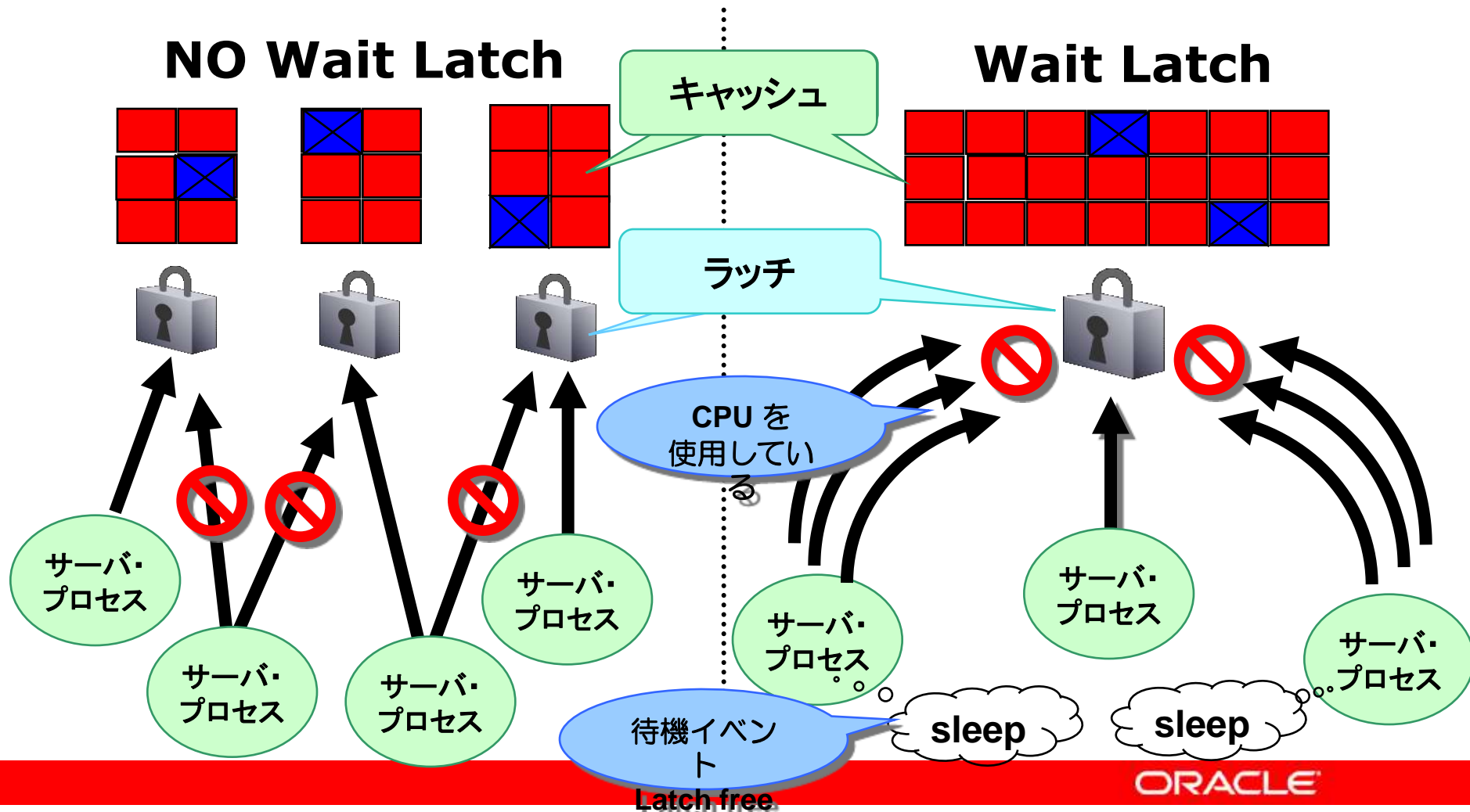
- Enqueue Activityからどのenqueueで待機が多いか確認する(10gから待機イベント名から判別可能)
 - STエンキュー ... enq: ST - contention
 - 動的な領域割り当てが主な原因
 - ローカル管理表領域を使用します
 - HWエンキュー ... enq: HW - contention
 - セグメントのHWMを越える領域の割り当てが原因
 - エクステントの手動割り当て
 - SQエンキュー ... enq: SQ - contention
 - 順序を生成する頻度が高いことが原因
 - 順序のキャッシュ数を増やす
 - TMエンキュー ... enq: TM - contention
 - 制約される列が索引付けされない場合の外部キー制約に関係している傾向
 - 外部キー列を索引付けします

この待機イベントが上位にきたら

- TXエンキュー
 - モード6（行レベルロック）・・・ enq: TX - row lock contention
 - アプリケーションの変更
 - モード4（ITLスロットの待機）・・・ enq: TX - allocate ITL entry
 - INITTRANS, MAXTRANSを変更しITLの個数を増やす
 - モード4（索引ブロック分割待ち）・・・ enq: TX - index contention
 - 逆キー索引やパーティションなどで分割の影響を受けるセッションを減らします
 - モード4（ビットマップ索引の更新待ち）・・・ enq: TX - row lock contention
 - 更新頻度が多い表にビットマップ索引を使用しない

ラッチ

キャッシュにアクセスする際のロック機構の一種



この待機イベントが上位にきたら

- CPUリソースが不足していないか確認する
 - ラッチ関係はCPU不足のケースが多い
- Latch ActivityからどのLatchで待機が多いか確認する
(10gから待機イベント名から重要なものは判別可能)
 - cache buffer chains ... latch: cache buffers chains
 - ホットブロックが存在することが原因
 - 採番表などの存在⇒順序を使用
 - 索引リーフブロックの競合⇒logical readsが多く索引のサイズが小さいものがあるかを調べる
 - cache buffer lru chain ... latch: cache buffers lru chain
 - 多くの全表検索、誤った索引の使用、DBWRの遅延などが原因

この待機イベントが上位にきたら

- Library cache ... latch: library cache
 - 過度な解析が原因
 - 共有されていないSQL
 - バインド変数を使用していない
 - 不必要な解析コールの発行
 - 共有プールサイズが小さい
 - Library Hit%など共有プール関連のインスタンス効率を確認
 - 共有されるようにアプリケーションを変更またはcursor_sharingをforceに設定
 - session_cached_cursorsを増やしsoft parseを減らす



以上の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

Oracle、PeopleSoft、JD Edwards、及びSiebellは、米国オラクル・コーポレーション及びその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標の可能性あります。