

はじめに

■コース概要と目的

Oracle データベースのパフォーマンス問題の分析方法、解決方法を説明します。

■受講対象者

データベース管理者の方を対象としています。

■前提条件

「データベース・アーキテクチャ」「データベース・マネジメント」を受講された方、もしくは同等の知識をお持ちの方。

■テキスト内の記述について

▼構文

[]	省略可能
{ A B }	A または B のどちらかを選択
n	数値の指定
_	デフォルト値

▼マーク

	指定バージョンからの新機能 (左記の場合、Oracle 12cR1 からの新機能)
	Enterprise Edition で使用できる機能
	知っておいたほうが良いテクニック、もしくは注意事項
	参照ページ
	データ・ディクショナリ・ビュー

第 1 章

Oracle パフォーマンス チューニングの基礎知識

この章ではパフォーマンス・チューニングを行う上での基本手順の説明と Oracle の基本アーキテクチャを確認します。

1. チューニング概要
2. チューニングのステップ
3. ベースライン
4. 待機イベント
5. パフォーマンス情報の収集
6. Oracle アーキテクチャ (SQL 処理と Oracle データベース)
7. 代表的なチューニングポイント
8. 代表的なアクセス・パス

7. 代表的なチューニングポイント

Oracle データベースでパフォーマンスの問題が発生している場合、次のような原因が考えられます。

(1) SQL

非効率な SQL によって、大量のディスク I/O やメモリー消費が引き起こされているケースが数多くあります。そのため、高負荷 SQL を特定し、その SQL の記述方法や実行計画に問題がないかどうかを調査します。実行計画とは、SQL を実行するための一連の処理のことであり、アクセス・パス（データベースからデータを取り出す経路）などが含まれます。代表的なアクセス・パスとして以下の2つがあります。

- ・全表スキャン (1-23)
- ・索引スキャン (1-25)

例えば、全表スキャンの SQL を索引スキャンに変更することで、ディスク I/O 量を削減できる場合があります。

※実行計画には、上記以外にも表の結合方法や結合する表の順序などがあります。詳細は「SQL パフォーマンス・チューニング」研修で解説しています。

(2) メモリー領域

システム・グローバル領域 (SGA) とプログラム・グローバル領域 (PGA) を適切に管理します。

1) システム・グローバル領域 (SGA)

SGA では、メモリー上に保持されているデータの共有率を高めることが重要です。

- ・共有プール
ライブラリ・キャッシュにおける SQL 解析結果の共有率の向上、断片化の防止・解消が重要です。SQL の記述を統一化し、共有プールを適正なサイズに調整します。
- ・データベース・バッファ・キャッシュ
バッファ・キャッシュに保持されているデータの共有率向上が重要です。まず SQL を改善し、その後、バッファ・キャッシュを適正なサイズに調整します。
- ・REDO ログ・バッファ
サーバー・プロセスの REDO ログ・バッファへの書き込み待機が発生しないようにすることが重要です。REDO ログ・バッファを適正なサイズに調整します。

2) プログラム・グローバル領域 (PGA)

PGAとはプロセスごとに取得される非共有のメモリー領域であり、PGA内のSQL作業領域でソート処理やハッシュ処理が行われます。SQL作業領域のサイズは時間帯などで使用量が変化するため、適切なサイズに設定できているかが問題になります。

自動PGAメモリー管理機能を使用している場合、PGAの合計サイズが適切になるように調整します。

(3) データベース

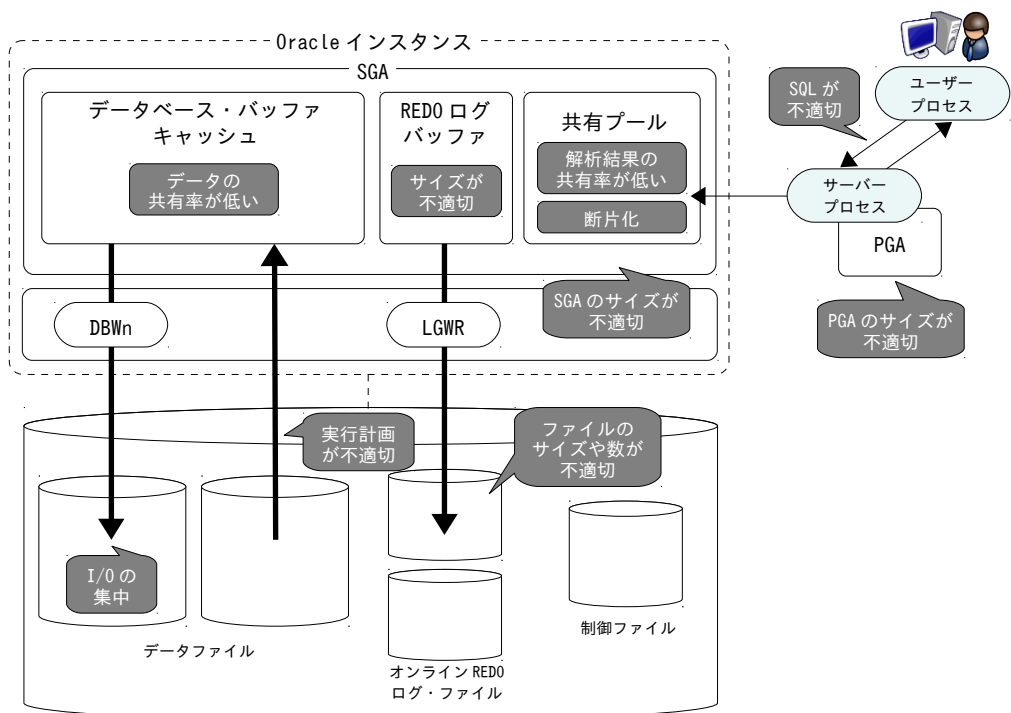
データベース・ファイルに関しては、ファイルへのI/O分散やファイルのサイズ・数の調整が重要です。

・データファイル

特定のデータファイルに対するI/O集中が問題になります。全表スキャンを行っているSQLを索引スキャンに変更してI/O量を削減したり、ディスク追加などでI/O分散を行います。

・オンラインREDOログ・ファイル

オンラインREDOログ・ファイルへのI/Oや、オンラインREDOログ・ファイルに影響を受けるチェックポイントなどが問題になります。ファイルのサイズや数、REDOログ・バッファのサイズなどを調整します。



第3章

代表的なチューニングポイント

Oracle データベースで問題となるケースが多い、代表的なパフォーマンス問題と、その解決方法について解説します。

1. 代表的なチューニングポイント（復習）
2. 共有プールのチューニング
3. データベース・バッファ・キャッシュのチューニング
4. REDO ログ構造のチューニング

4. REDO ログ構造のチューニング

REDO ログ・バッファやオンライン REDO ログ・ファイルなどの REDO ログ構造のチューニングについて解説します。

(1) REDO ログ構造概要

変更処理で発生した REDO エントリは、REDO ログ・バッファに一時的に格納されます。その後、主に以下のタイミングで LGWR プロセスがまとめてオンライン REDO ログ・ファイルに書込みます。REDO ログ・バッファの領域を使い切った後は、書込みが完了した領域を上書きして再利用します。

- ・コミット時
- ・ REDO ログ・バッファの 1/3 が使われた時、または 1MB の領域が使用された時
- ・ DBWn がダーティ・バッファを書込む前

(2) REDO ログ構造の調査

関連するシステム統計や待機イベントを調査し、チューニングが必要かどうかを検討します。

■システム統計を使用して REDO ログ・バッファの待機を確認

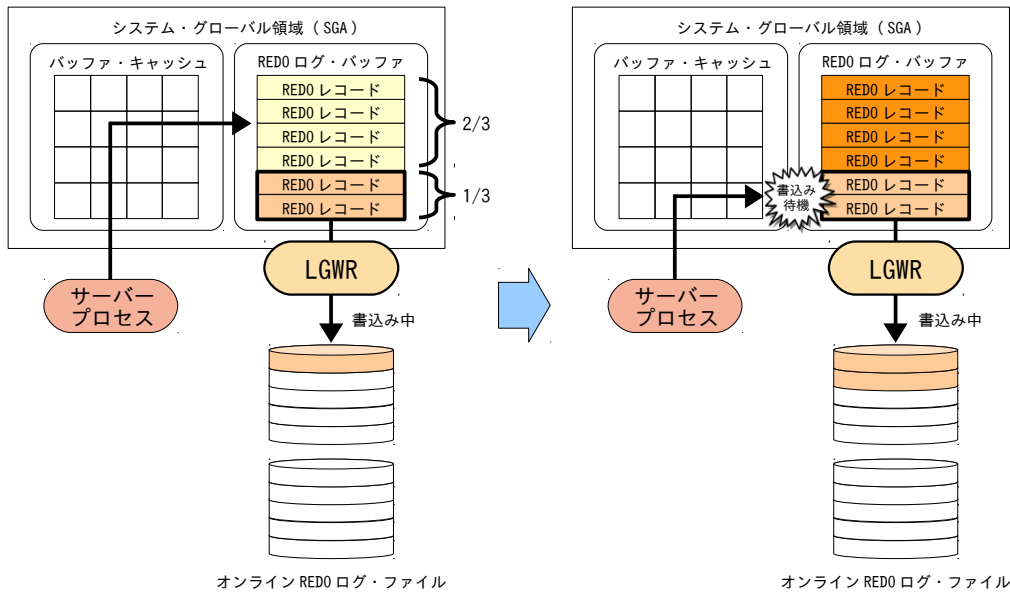
LGWR が REDO エントリをオンライン REDO ログ・ファイルへ書込んでいる間、サーバー・プロセスが REDO ログ・バッファに REDO エントリを書込み、REDO ログ・バッファが満杯になることがあります。このような場合、サーバー・プロセスは REDO ログ・バッファへの書込みを待機します。

この待機発生回数が、全 REDO エントリ数の 1% を上回らないようにします。待機発生回数 (redo buffer allocation retries) と REDO エントリ数 (redo entries) は V\$SYSSTAT ビューで確認できます。

■REDO ログ構造に関連する待機イベント

イベント名	説明
log buffer space	REDO ログ・バッファが満杯でサーバー・プロセスが書込みを待機した。ログ・バッファが小さすぎるか、LGWR プロセスの書込みが遅い。
log file parallel write	LGWR の REDO ログ・ファイルへの書込みに関する待機。I/O 遅延の問題。
log file sync	サーバー・プロセスが COMMIT または ROLLBACK 処理の完了を待機した。通常、LGWR の I/O 遅延が原因。 ※log file parallel write 待機イベントの時間も含まれる
log file switch completion	ログ・スイッチ時に発生する待機。I/O 遅延の問題。
log file switch (checkpoint incomplete)	チェックポイントが完了していないため、オンライン REDO ログ・ファイルを上書きできない場合の待機。
log file switch (archiving needed)	アーカイブが完了していないため、オンライン REDO ログ・ファイルを上書きできない場合の待機。

<REDO ログ・バッファの待機>



REDO ログ・バッファの1/3が使われたため、LGWR が REDO エントリをログ・ファイルに書込む。同時に変更処理が発生したため、サーバー・プロセスが REDO バッファの残りの2/3の領域に REDO エントリを書込む。

サーバー・プロセスの書き込みが2/3の領域を使い切り、残りの領域を上書きしたいが、LGWR の書き込みが終わっていない。そのため、サーバー・プロセスは LGWR の書き込み完了まで待機する。

例) REDO ログ・バッファでの待機状態を調査する。

```
SQL> SELECT sum(decode(name,'redo buffer allocation retries',value,0)) /
2          sum(decode(name,'redo entries',value,0)) RATIO
3 FROM    v$sysstat;

          RATIO
-----
.013731
```

※上記例では、待機は0.01%でほとんど問題になっていません。

第4章

メモリーの管理

バージョンアップごとに進化している、メモリー管理機能の使用方法について説明します。

1. メモリー管理概要
2. 自動共有メモリー管理
3. 自動PGAメモリー管理
4. 自動メモリー管理
5. メモリー・アドバイザー機能

1. メモリー管理概要

Oracle が使用するメモリー領域には、SGA（システム・グローバル領域）と PGA（プログラム・グローバル領域）があります。これらのメモリー領域を適切に管理することは、パフォーマンスにおいて非常に重要です。

(1) 各領域の管理方法

領域ごとのメモリー管理について解説します。

■SGA（4-3～）

SGA 内の各領域（共有プール、データベース・バッファ・キャッシュなど）を個別に調整できます。

また、自動共有メモリー管理機能を使用すると、SGA の総メモリーサイズのみ指定するだけで、Oracle が SGA 内の各領域を自動管理してくれます。

■PGA（4-9～）

PGA はプロセスごとに確保されるメモリー領域で、SQL 処理に必要なデータや制御情報、SQL 作業領域（ソートやハッシュで使用）などが含まれます。

PGA のサイズは、セッション数や処理内容によって増減するため、PGA を監視したり、各 SQL 作業領域を個別にチューニングしたりすることが大きな負担になります。

自動 PGA メモリー管理機能を使用すると、PGA の総メモリーサイズのみ指定するだけで、Oracle が各 PGA を自動管理してくれます。

■メモリー全体（4-13～）

Oracle 11g から自動メモリー管理機能が提供されました。この機能を使用すると、Oracle に割当て可能な総メモリー・サイズを指定するだけで、Oracle が SGA、PGA を自動調整してくれます。

対象	管理方法	初期化パラメータ	概要
SGA	手動共有メモリー管理	SHARED_POOL_SIZE DB_CACHE_SIZE など	共有プールなど、メモリー領域ごとにサイズを調整。
	自動共有メモリー管理	SGA_TARGET	SGA の総メモリー・サイズのみ設定すれば、SGA 内の各領域は Oracle が自動調整。
PGA	手動 PGA メモリー管理	*_AREA_SIZE (SORT_AREA_SIZE など)	セッションごとに SQL 作業領域のサイズを *_AREA_SIZE パラメータで調整。
	自動 PGA メモリー管理	PGA_AGGREGATE_TARGET	PGA の総メモリー・サイズのみ設定すれば、各 SQL 作業領域が自動調整される。
SGA+PGA	自動メモリー管理 (Oracle 11g～)	MEMORY_TARGET	SGA と PGA の総メモリー・サイズを設定すれば、Oracle メモリーが自動調整される。

(2) メモリーサイズの管理

メモリーに割当てる初期値は、経験やテスト、またはガイドラインに基づいて設定します。また、運用中は定期的に監視し、必要に応じて割当てサイズを調整します。

■初期設定のガイドライン

サーバーの物理メモリーから、Oracle 以外のアプリケーションで使用されるメモリーを除いたサイズを SGA と PGA で使用します。

一般的に Oracle メモリーに割当てる初期値は、経験やテスト、以下のガイドラインに基づいて設定します。SGA と PGA での割当ての目安は、OLTP システムと DSS システムでは異なります。一般的に、DSS システムは大規模なソートやハッシュ処理が行われるため、PGA のサイズを OLTP システムに比べて大きく設定します。

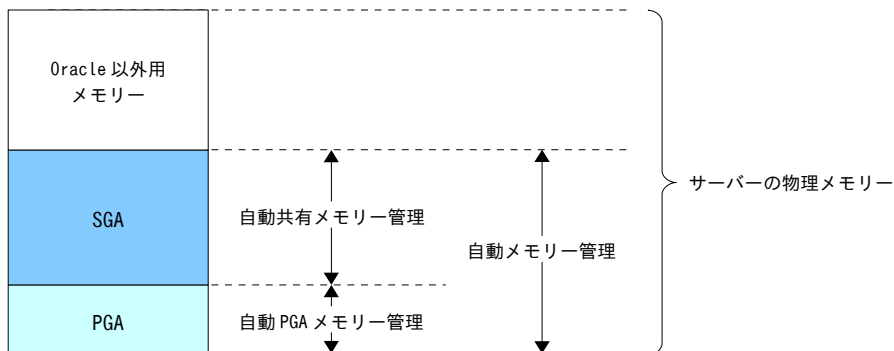
システム	SGA	PGA
OLTP システム	80%	20%
DSS システム	50%	50%

■メモリーの監視 (4-7、4-11、4-15)

運用中は、メモリーに関するデータベース統計を確認し、各領域に割当てたサイズが適切かどうかを監視します。

■適切なサイズに調整 (4-7、4-12、4-15)

最適なメモリー・サイズを見つけるには、テストをする他、メモリー・アドバイザ機能を利用する方法があります。この機能は、実行されたワークロードをもとに分析を行い、様々なメモリー・サイズでどのように動作するかの予測値を提示してくれます。メモリー・アドバイザ機能はメモリー領域ごとに用意されています。



第5章

Statspack によるデータベース診断

データベース診断ツールである Statspack の使用方法と、Statspack によって出力されるレポートファイルの分析方法について説明します。

1. Statspack 概要
2. Statspack レポートの分析
3. Statspack の使用方法
4. スナップショットの取得
5. AWR によるデータベース診断

(2) 負荷特性の確認 (Load Profile セクション)

1 秒単位、1 トランザクション単位でシステム統計を確認できます。

■参照時のポイント

ベースラインの中心的な情報が記録されているため、複数のレポートでこのセクションを比較し、負荷の変化を確認します。

<主な列>

・ Per Second 統計

1 秒あたりのスループットを表します。この値が高くなるほど1秒あたりの作業量が多くなります。

・ Per Transaction 統計

1 トランザクションあたりの処理負荷を表します。この値が高くなるほど、1 トランザクションで行った処理負荷が高くなります。

<システム統計>

統計名	詳細
DB time(s)	SQL の実行に費やした時間
DB CPU(s)	SQL の実行に費やした CPU 時間
Redo size	生成された REDO のサイズ
Logical reads	アクセスしたブロック数
Block changes	変更されたブロック数
Physical reads	ディスクから読込んだブロック数
Physical writes	ディスクに書込んだブロック数
User calls	ログイン、解析、フェッチ、実行などのユーザー・コール数
Parses	SQL の解析回数
Hard parses	ハード解析の回数
W/A MB processed	SQL 作業領域の使用量
Logons	ログオン数
Executes	SQL 実行数 (再帰 SQL を含む)
Rollbacks	ロールバック数
Transactions	トランザクション数

例) Load Profile セクション

Load Profile	Per Second	Per Transaction	Per Exec	Per Call
DB time(s): ①	0.2	3.0	0.00	0.01
DB CPU(s):	0.1	1.8	0.00	0.00
Redo size:	482,067.6	9,659,207.1		
Logical reads:	4,488.1	89,927.7		
Block changes:	3,778.3	75,705.7		
Physical reads: ②	45.7	916.2		
Physical writes:	25.8	516.2		
User calls:	28.0	561.4		
Parses: ③	9.0	180.4		
Hard parses:	3.9	77.8		
W/A MB processed: ⑤	0.6	12.1		
Logons:	0.0	0.7		
Executes:	874.1	17,515.2		
Rollbacks:	0.0	0.0		
Transactions:	0.1			

- ① 「DB time」でユーザー・コールの処理に要した時間、「DB CPU」でCPU処理に要した時間を確認できます。この2つの隔たりが大きい場合、待機時間が問題になっている可能性があるため、後続セクションの「Top 5 Timed Events」で主な待機イベントを確認します。また、チューニング前後で「DB time」を比較し、チューニングの効果を確認できます。
- ② ディスク I/O の状態を確認できます。「Physical reads」が増加している場合、データベース・バッファ・キャッシュのヒット率が低下している可能性があります。ヒット率のセクションも確認した上で、SQL チューニングやデータベース・バッファ・キャッシュのサイズ増加などを検討します。
- ③ 解析の状態を確認できます。「Hard parses」が増加している場合、共有プールのヒット率が低下している可能性があります。ヒット率のセクションも確認した上で、バインド変数の使用や共有プールのサイズ増加などを検討します。
- ④ 変更処理の量を確認できます。これらの値が増えている場合、ログの書込み待機などが発生している可能性があります。待機イベントのセクションも確認した上で、ログ・バッファのサイズやログ・ファイルの I/O などのチューニングを検討します。
- ⑤ SQL 作業領域の使用量を確認できます。この値が増加している場合、ソートやハッシュ処理が増えている可能性があります。PGA 関連のセクションも確認した上で、PGA のサイズ増加を検討します。
- ※Oracle 10g までは「Sorts」と表示されます。



5. AWR によるデータベース診断

AWR (自動ワークロード・リポジトリ) とは、Statspack を進化させた機能です。

Oracle Enterprise Manager (OEM) と API で使用できますが、使いやすさの観点から一般的に OEM で使用されます。

※本機能を使用するには、Enterprise Edition かつ有償オプション (Diagnostics Pack) のライセンスが必要です。

(1) AWR と Statspack の比較

AWR は Statspack と比較して、主に以下の点が優れています。

■管理者が手動でスナップショットを取得する必要がない

以下の2つの情報が自動的に取得されるため、手動でスナップショットを取得する必要がありません。

・AWR スナップショット

AWR スナップショットには Statspack のスナップショットと同じ情報に加え、稼働統計やワークロード情報などが含まれます。

デフォルトでは1時間に1回取得され、8日間 (Oracle 10g では7日間) 保存されます。

※取得時の負荷も Statspack より低くなっています。

・ASH (アクティブ・セッション履歴)

ASH は、1秒ごとに V\$SESSION ビューをサンプリングした情報の集合体です。

AWR スナップショットは Statspack のスナップショット同様、1時間ごとのサマリー情報のため、瞬間的に発生した問題分析には不向きです。ASH を利用することで、そのような問題を解消できます。

ASH は AWR スナップショット同様、8日間保存されます。

■パフォーマンス・レポートの改善

テキスト形式と HTML 形式を選べます。HTML 形式は見やすく分析も容易にできます。

また、Statspack よりも多くのセクションがパフォーマンス・レポートに出力されます。

■期間比較レポートが作成できる

ある2つのパフォーマンス・レポートを並べて比較するレポートを作成できます。

通常時と問題発生時のレポートを比較して問題点を分析したり、チューニング前後のレポートを比較して効果を確認したりすることが容易にできます。

例) HTML 形式のパフォーマンス・レポートの一部抜粋 (Load Profile セクションと Instance Efficiency Percentages セクション (Statspack の Instance Efficiency Indicators セクションに相当))

Load Profile					
	Per Second	Per Transaction	Per Exec	Per Call	
DB Time(s):	0.4	9.0	0.04	2.95	
DB CPU(s):	0.0	0.0	0.00	0.01	
Redo size (bytes):	2,438.7	56,745.1			
Logical read (blocks):	132.2	3,076.3			
Block changes:	7.0	162.4			
Physical read (blocks):	2.9	68.1			
Physical write (blocks):	0.7	16.8			
Read IO requests:	2.6	59.5			
Write IO requests:	0.6	14.6			
Read IO (MB):	0.0	0.5			
Write IO (MB):	0.0	0.1			
User calls:	0.1	3.1			
Parses (SQL):	4.0	92.3			
Hard parses (SQL):	0.6	13.6			
SQL Work Area (MB):	0.1	3.2			
Logons:	0.1	1.1			
Executes (SQL):	10.2	237.4			
Rollbacks:	0.0	0.0			
Transactions:	0.0				

Instance Efficiency Percentages (Target 100%)					
Buffer Nowait %:	100.00	Redo NoWait %:	100.00		
Buffer Hit %:	97.72	In-memory Sort %:	100.00		
Library Hit %:	95.92	Soft Parse %:	98.07		
Execute to Parse %:	11.00	Latch Hit %:	99.25		
Parse CPU to Parse Elapsed %:	25.64	% Non-Parse CPU:	8.42		

例) 期間比較レポートの一部抜粋

Top Timed Events											
<ul style="list-style-type: none"> Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes 											
1st						2nd					
Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time	Event	Wait Class	Waits	Time(s)	Avg Time(ms)	%DB time
control file sequential read	System I/O	7,977	142.95	17.92	10.41	db file sequential read	User I/O	4,978	27.59	5.54	96.63
db file sequential read	User I/O	9,052	104.49	11.54	7.61	control file sequential read	System I/O	9,000	18.34	2.04	64.24
Disk file operations I/O	User I/O	301	23.85	79.24	1.74	CPU time			9.02		31.58
cursor: pin S wait on X	Concurrency	56	20.47	365.48	1.49	Disk file operations I/O	User I/O	61	2.40	39.38	8.42
control file parallel write	System I/O	1,217	12.86	10.57	0.94	control file parallel write	System I/O	246	1.22	4.97	4.28
CPU time			6.60		0.48	db file parallel write	System I/O	388	0.52	1.33	1.80