

**19<sup>c</sup>** ORACLE<sup>®</sup>  
Database

# Oracle Advanced Compression

Oracle ホワイト・ペーパー | 2019 年 2 月



ORACLE<sup>®</sup>

## 目次

概要.....	2
データ圧縮.....	3
移行とベスト・プラクティス.....	5
非構造化データ圧縮.....	7
バックアップ圧縮.....	8
索引圧縮.....	10
ネットワーク圧縮.....	11
Oracle Data Guard REDO 転送の圧縮.....	11
情報ライフサイクル管理.....	11
ヒート・マップ.....	12
自動データ最適化.....	12
その他の機能.....	13
Flashback Data Archive 履歴表の最適化.....	13
ストレージ・スナップショット最適化.....	14
Hybrid Columnar Compression の行レベル・ロック.....	14
Exadata Flash Cache Compression.....	15
表/パーティション/サブパーティションのオンラインでの移動（圧縮形式へ）.....	15
結論.....	1
免責事項.....	1

## 概要

企業が保存し管理しているデータ量は急増しています。各種業界の概算によると、データ量は2、3年ごとに倍増しています。このデータ量の急激な増加によって、ITは困難な課題に直面しています。第1の課題は、ストレージ・コストです。各ストレージのコストは激減しているにもかかわらず、データ量が急増していることにより、ストレージは引き続き大部分のIT予算における最大のコスト要因の1つになっています。さらに、データベースがますます拡大していることから、予算内に収めながらパフォーマンスに対する要件に応え続けることが難しくなっています。

オラクルは、データベース圧縮技術において先駆的な存在です。Oracle Advanced Compression と Oracle Database はともに、この複雑な環境でITマネージャーを成功に導く、一連の強力な圧縮機能、パフォーマンス機能、およびデータ・ストレージ最適化機能を提供します。Oracle Advanced Compression は、顧客のパフォーマンス向上とストレージ・コスト削減を支援する、包括的な一連の圧縮機能を提供します。これにより、リレーショナル・データ（表）、非構造化データ（ファイル）、索引、ネットワーク・データ、バックアップ・データなどのあらゆるタイプのデータの圧縮が可能になり、IT管理者は、データベース記憶域全体のフットプリントを大幅に削減できます。

多くの場合、サーバー（本番、開発、QA、テスト、バックアップなど）のストレージ・コストの削減や最適化がもっとも具体的な利益をもたらす方法として見られていますが、Oracle Advanced Compression の全機能は、メモリ、ネットワーク帯域幅、ストレージなどのITインフラストラクチャのあらゆるコンポーネントのパフォーマンスを向上させるように設計されています。Oracle データベース環境がクラウドかオンプレミスかに関係なく、Oracle Advanced Compression を利用すると、アプリケーションを変更せずに、複数の異なる環境全体で強力な圧縮機能を利用できます。Oracle Advanced Compression で得られるメリットには、データベース記憶域のフットプリントの削減、バックアップの節約、システム・パフォーマンスの向上などがあります。

Oracle Advanced Compression は、コスト削減とパフォーマンス向上を目的とした、包括的な一連の圧縮機能を提供します。これらの機能では、構造化データ、非構造化データ、索引、データベース・バックアップ、ネットワーク・トラフィックのための圧縮や Data Guard REDO 用の圧縮を実現できます。本書では、これらの Oracle Advanced Compression の機能について、それぞれ説明します。

## データ圧縮

Oracle Database 11g Release 1 で OLTP 表圧縮（現在の高度な行圧縮）が導入されました。この機能を使用することで、INSERT や UPDATE などの従来の DML を含むあらゆるタイプのデータ操作中に、データを圧縮できます。さらに、高度な行圧縮では、圧縮データに対する書き込み操作のオーバーヘッドが最小限に抑えられます。そのため、トランザクション環境や OLTP 環境に加え、データウェアハウスにも適しており、圧縮の利点がすべてのアプリケーションのワークロードへと拡張されます。

高度な行圧縮では、OLTP/DW アプリケーションで動作するよう特別に設計された、独自の圧縮アルゴリズムが使用されます。このアルゴリズムは、データベース・ブロック内や複数の列における重複値を排除することによって動作します。圧縮されたブロックには、圧縮メタデータを維持する記号表と呼ばれる構造体が含まれます。ブロックが圧縮されると、最初に重複値のコピーが記号表に1つ追加されることにより、重複値が排除されます。そして、各重複値が、記号表内の適切なエントリへの短い参照に置き換えられます。

この革新的な設計では、圧縮されたデータを元の状態へ変換するために使用されるメタデータがブロック・ヘッダー内に保存されるため、圧縮されたデータはデータベース・ブロック内で自己完結します。グローバルなデータベースの記号表を維持する競合他社の圧縮アルゴリズムと比較すると、圧縮されたデータにアクセスする際、追加の I/O（グローバル記号表とともに必要）が発生しないオラクル独自のアプローチでは、大幅なパフォーマンス上の利点が得られます。

### 高度な行圧縮の利点

ある特定の環境で得られる圧縮率は、圧縮されたデータ、特にデータのカーディナリティによって異なります。通常、高度な行圧縮を使用することにより、ストレージ領域の消費の 2~4 倍の削減が期待できます。つまり、非圧縮データ量が消費する領域量は、圧縮されたデータ量が消費する領域量の 2~4 倍になります。

高度な行圧縮の利点は、ディスク上のストレージ節約の域のみにとどまりません。重要な利点の1つは、ブロックを解凍することなく、圧縮されたブロック（データと索引）を直接メモリ内で読み取れることです。そのため、I/O が削減され、I/O 操作に関連するシステム・コール数が削減されるため、パフォーマンスが向上します。さらに、メモリの追加を必要とせずにより多くのデータを保存することにより、バッファ・キャッシュの効率が向上します。

### 最小限のパフォーマンス・オーバーヘッド

すでに説明したように、高度な行圧縮には、読取り操作に対する悪影響はありません。データの書き込み中には作業が追加で実行されることがありますが、書き込み操作のパフォーマンス・オーバーヘッドを完全に排除することは不可能です。このような高度な行圧縮のオーバーヘッドを最小限に抑えるための最適化がいくつかあります。

おもな最適化は、Oracle Database が、書き込み操作が発生するたびにデータを圧縮するのではなく、バッチ・モードでブロックを圧縮することです。新しく初期化されたブロックは、ブロック内のデータが内部で制御されるしきい値に達するまで、圧縮されずに維持されます。トランザクションによってブロック内のデータがこのしきい値に達すると、ブロックのすべてのコンテンツが圧縮されます。さらに、より多くのデータがブロックに追加されて再びしきい値に達すると、ブロック全体が再圧縮されて、圧縮の最高レベルに到達します。

このプロセスは、圧縮を続けてもそれ以上ブロックに利点が得られないと Oracle が判断するまで繰り返されます。ブロックの圧縮を実行するトランザクションのみ、圧縮のオーバーヘッドはわずかで済みます。したがって、圧縮されたブロック上の DML トランザクションのほとんどのパフォーマンスは、非圧縮ブロックの場合のパフォーマンスと同様になります。



高度な行圧縮のパフォーマンスのその他の最適化には、以下のものがあります。

### 部分圧縮

高度な行圧縮を使用すると、一杯になったブロックが圧縮されます。圧縮後、より多くの行が追加され（ブロックに追加できる行が増えるため）、ブロック内の行が圧縮できなくなるまで、再圧縮のプロセスが何度か繰り返されます。ブロックは通常、全体が圧縮され再フォーマットされますが、ブロックを部分的に圧縮して、CPU を節約し、圧縮に余裕を持たせることが可能な場合があります。

部分圧縮は、すでに圧縮された（つまり、高度な行圧縮で圧縮された）ブロックに対して使用します。未圧縮行を探して圧縮形式に変換し、ブロック・ディクショナリからの記号を追加または再使用します。この方法を使えば、ブロック全体を再圧縮するよりも速く圧縮できます。完全なブロック再圧縮では、ブロック内の行がいずれもロックされていないこと、または行をブロックに挿入するトランザクションによってブロック内の全行がロックされていることが求められます。部分圧縮は未圧縮でロック解除された行しかロック、圧縮しないので、これらの要件を回避します。部分圧縮は、ブロック内のコミットされていない他のトランザクションがある場合に実行されます。

### バックグラウンド圧縮

通常、高度な行圧縮のブロック圧縮は、ダイレクト・ロード操作または DML の insert/update 操作を使わないとトリガーされませんでした。Database 12c Release 2 以降、バックグラウンド領域タスクにより、圧縮候補と見なされたブロックを圧縮できます。以前の DML 操作を基に、またはブロックの領域使用を評価する他の領域タスクにより、ブロックを圧縮候補としてマークすることができます。バックグラウンドで実行された圧縮は、高度な行圧縮と同じですが、アクティブな SQL セッションではなく、バックグラウンド・プロセスによってトリガーされます。バックグラウンドでトリガーされる圧縮のもう 1 つの例は、ADO（自動データ最適化）の行レベルの圧縮ポリシーを使用する場合です。

### 配列挿入

Oracle Database 12c Release 2 より前のリリースでは、配列挿入によってブロックあたり複数の再圧縮が行われました。バッチ行がブロックに挿入された後、ブロックが圧縮され、次のバッチ行が挿入されると、ブロックが再圧縮されるという操作が続きます。Database 12c Release 2 以降、Oracle Database では圧縮済みブロックに格納される行数が見積もられます。これらのすべての行がバッファおよび圧縮され、完全なブロック・イメージが生成されます。つまり、Database 12c Release 2 より前のリリースでは、圧縮が何度も行われる可能性があったのとは対照的に、通常はブロックごとに 1 回か 2 回しか行われません（1 回の圧縮で圧縮比を見積もります）。

この最適化は、経過時間に多大なメリットをもたらします。Oracle Database の場合、より多くの行をまとめて圧縮する（したがって、共通の記号をより効果的に抽出できる）ので、はるかにすぐれた圧縮比も得られます。また、これらのアルゴリズムは、バッファに格納される行数を適宜変更し、1つのブロックに入る圧縮行数に関する見積もりに応じて、バッファに入れる行数を増やします。この強化により、高度な行圧縮を使った表は、圧縮されていない表よりもスキャン速度が大幅に上がるだけでなく、挿入もより高速に行われます。これは、論理ブロックと物理ブロックの取得数が減るためです。

## 移行とベスト・プラクティス

新しい表やパーティションに対して高度な行圧縮を有効化する方法は簡単です。単純に表またはパーティションを CREATE 構文で作成し、“ROW STORE COMPRESS ADVANCED”と指定します。たとえば、次のような文を使用します。

```
CREATE TABLE emp (emp_id NUMBER, first_name VARCHAR2(128), last_name  
VARCHAR2(128)) ROW STORE COMPRESS ADVANCED;
```

既存の表やパーティションに対して高度な行圧縮を有効化する上で、推奨される方法はいくつかあります。

### 1. ALTER TABLE ... ROW STORE COMPRESS ADVANCED

- 今後のすべての DML について高度な行圧縮を有効化しますが、表の既存のデータについては非圧縮のままにします。

### 2. オンライン再定義 (DBMS\_REDEFINITION)

- 今後の DML について高度な行圧縮を有効化し、さらに既存のデータを圧縮します。DBMS\_REDEFINITION を使用すると、移行中に、読取り/書込みの両方のアクティビティに対して、表がオンライン状態で維持されます。最適なパフォーマンスを得るには、DBMS\_REDEFINITION をパラレルで実行します。
- オンライン再定義では、操作の実行中に索引が仮表にクローンされます。クローンされたすべての索引の増分が同期（リフレッシュ）操作中に維持されるため、オンライン再定義の実行中も実行後も、索引の利用が中断されることはありません。ただし、オンライン再定義をパーティションの再定義に使用する場合に限り、索引の利用は中断されます。この場合、グローバル索引がすべて無効化され、オンライン再定義の実行後にそのグローバル索引を再作成する必要があります。

### 3. ALTER TABLE ... MOVE ROW STORE COMPRESS ADVANCED

- 今後の DML について高度な行圧縮を有効化し、さらに既存のデータを圧縮します。表の移動中、読取りアクティビティに対しては表がオンライン状態で維持されますが、排他 (X) ロックがかかるため、移動コマンドが完了するまですべての DML がブロックされます。最適なパフォーマンスを得るには、ALTER TABLE ... MOVE をパラレルで実行します。
- ALTER TABLE ... MOVE により、パーティションまたは表にある索引がすべて無効化されます。ALTER TABLE ... MOVE の実行後、これらの索引を再作成する必要があります。パーティションの移動の場合、ALTER TABLE ... MOVE PARTITION を UPDATE INDEXES 句とともに使用すると、索引が維持されます（排他 (X) ロックがかかるため、移動コマンドが完了するまで、すべての DML がブロックされます）。パーティション化されていない表に対しては、この句は使用できません。
- ALTER TABLE...MOVE 文を使用すると、パーティション化されていない表のデータやパーティション化された表のパーティションのデータを新しいセグメントに再配置したり、オプションとして

異なる表領域に再配置したりできます。ALTER TABLE...MOVE ROW STORE COMPRESS ADVANCED により、圧縮データ用の新しいエクステントが移動先の表領域内に作成され、データが圧縮されます。ここで、新しいセグメントは、データファイルの末尾や先頭に配置されるとは限らず、あらゆる場所に配置される可能性がある点に注意が必要です。そのため、元のセグメントが解放されるとき、エクステントの位置によっては、データファイルが縮小されない場合もあります。

#### 4. ALTER TABLE ... MOVE TABLE/PARTITION/SUBPARTITION ... ONLINE ROW STORE COMPRESS ADVANCED

今後の DML について高度な行圧縮を有効化し、さらに既存のデータを圧縮します。ALTER TABLE...MOVE TABLE/PARTITION/SUBPARTITION ... ONLINE を使用すると、移動中の表/パーティション/サブパーティションで、DML 操作を中断せずに実行し続けることができます。索引は移動操作中に維持されるため、手動による索引の再作成は不要です。

Oracle Advanced Compression の機能を使用する際のベスト・プラクティスと考慮事項は次のとおりです。

- 一般には、データベース内のすべてのアプリケーション関連表を圧縮することが推奨されますが、例外が 1 つあり、それは表がキューとして使用される場合です。つまり、行が表に挿入された後、大部分またはすべての行が削除され、その後さらに多くの行が挿入されてそれも削除される場合です。このタイプのアクティビティは、一時的な行を絶えず圧縮するためのオーバーヘッドが生じるため、圧縮に適したユースケースではありません。
- Oracle Advanced Compression の各機能の最適なテスト環境は、本番環境に合わせて再現した環境です。この環境で、もっとも現実的な（圧縮前および圧縮後の）パフォーマンス比較と機能比較を行うことができます。
- 保存されるデータの重複度が高い（カーディナリティが低い）場合に、高度な行圧縮による領域使用の削減効果ももっとも高くなります。これは、特にバックアップに当てはまります。圧縮率が高いほどバックアップされるデータ量が少なくなり、そのためリカバリ時間も短くなります。バルク・ロードの前に（重複度のもっとも高い列の）データをソートすると、圧縮率が高くなる場合があります。
- プリフィックス圧縮（索引）は、Oracle Database Enterprise Edition に付属しています。
- CPU オーバーヘッドは通常は最小限で済みますが、Advanced Row Compression は、CPU サイクルが空いているシステムに対して導入することが適しています。圧縮により、一部の DML 操作では、非常に小さなものではありませんが、追加のオーバーヘッドが発生するためです。
- PL/SQL パッケージである Compression Advisor は、データ・サンプルの分析に基づいて、高度な行圧縮によるストレージ削減効果を見積もるために使用します。このパッケージにより、高度な行圧縮の導入後、実際の圧縮率を適切に見積もることができます。Oracle9i Database Release 2 から Oracle Database 11g Release 1 までをサポートする Compression Advisor のバージョンは、Oracle.com の Advanced Compression のページより無償で入手できます。Compression Advisor (DBMS\_COMPRESSION) は、Oracle Database 11g Release 2 以降に含まれています。
- LONG データ型を含む表での高度な行圧縮の使用は、サポートされていません。
- ブロック・サイズが大きいくほど、高度な行圧縮の圧縮率が高くなるわけではありません。ブロック・サイズの増減が高度な行圧縮の圧縮率に影響を及ぼすかを確認したい場合は、独自のデータを使用してテストすることを推奨します。
- サイズが 4K 以上の LOB は SecureFiles を使って管理することをお奨めします。Oracle Advanced Compression の機能である高度な LOB 圧縮と重複排除は、LOB に必要なストレージ量を減らします。

- Data Pump 圧縮は、高度な行圧縮とは関係ありません。Data Pump のダンプファイルは、インポート・プロセス中にインラインで圧縮されないため、そのデータは、表の圧縮特性に基づいてターゲット表にインポートされます。
- Oracle Database 12c Release 2 以前では、多くの種類の連鎖行を含むブロックは圧縮できませんでした。Oracle Database 12c Release 2 以降、この制限が取り除かれています。
- 索引構成表 (IOT) は本質的に索引であるため、高度な行圧縮または基本圧縮で圧縮することはできません。IOT は、プリフィックス圧縮で圧縮できます。
- 索引のプリフィックス列がリーフ・ブロック内で何回も繰り返される場合は、索引キー (プリフィックス) 圧縮が非常に有益な可能性があります。ただし、先頭列が非常に選択的な場合や、プリフィックス列に対応して繰り返される値がそれほど多くない場合は、索引キー圧縮は有益ではありません。
- 高度な行圧縮は、表領域レベルの暗号化で適切に機能します。暗号化の前に表が圧縮されるため、圧縮率は暗号化の影響を受けません。列レベルの暗号化では、圧縮前に暗号化が実行されるため、圧縮率が低下します。
- ADO ポリシーをより柔軟にカスタマイズする必要がある場合は、カスタム ADO ポリシーを使用できます。カスタム ADO ポリシーでは、ユーザーが指定した関数を使用して、該当する各セグメントを評価します。
- ある層から別の層へできるだけ迅速にデータを移動する必要があり、次のメンテナンス期間まで待てないような場合があります。ADO ポリシーを実行することで、既存のポリシーの内容を問わず、データの移動や圧縮をオンデマンドで実行できます。

## 非構造化データ圧縮

SecureFiles は、ドキュメント、イメージ、スプレッドシート、XML ファイルなどの非構造化コンテンツを保存するための、両方の長所を備えたアーキテクチャを提供します。特に Oracle Database の利点を維持しながら、従来のファイル・システムと同等かそれを上回る高いパフォーマンスをファイル・データにもたらしように設計されています。

SecureFiles は、ANSI 規格の LOB データ型のスーパーセットとして設計されており、SecureFiles の前段階である既存の BasicFiles LOB からの移行を容易にします。SecureFiles によって、組織は、単一のセキュリティ/監査モデルや統合されたバックアップおよびリカバリ・プロセスを使用して、すべてのリレーショナル・データおよび関連するファイル・データを Oracle Database で管理でき、全情報にわたってシームレスな検索を実行できるようになります。

Oracle Advanced Compression には、SecureFiles データのストレージのフットプリントを大幅に削減し、パフォーマンスも向上させる、高度な LOB 圧縮および高度な LOB 重複排除機能が搭載されています。

## 高度なLOB重複排除

アプリケーションにおいて、ファイルの完全なレプリカを保存することは極めて一般的です。その例として、複数のユーザーが同一の添付ファイルを受信する電子メールのアプリケーションがあります。高度な LOB 重複排除は、SecureFiles データの重複コピーを排除します。Oracle Database では、SecureFiles データのイメージを 1 つ保存し、重複したコピーをそのイメージへの参照に置き換えます。

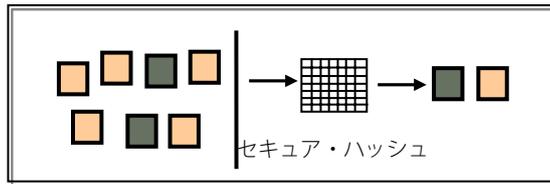


図 3：高度な LOB 重複排除

たとえば、1 MB の同一のファイルが添付されている電子メールを 10 人のユーザーが受信する、電子メール・アプリケーションがあるとします。高度な LOB 重複排除がない場合、システムは 10 人のユーザーそれぞれに対してファイルのコピーを 1 つずつ保存するため、10 MB のストレージが必要になります。この例の電子メール・アプリケーションで高度な LOB 重複排除を使用した場合、1 MB の添付ファイルを一度保存するだけで済みます。これは、ストレージ要件の 90 % の節約になります。

ストレージの節約に加え、高度な LOB 重複排除によってアプリケーションのパフォーマンスも向上します。特に、SecureFiles データへの参照のみが書き込まれるため、書き込み操作およびコピー操作の効率が向上します。さらに、重複した SecureFiles データがすでにバッファ・キャッシュにある場合、読取り操作が改善されます。

## 高度な LOB 圧縮

Oracle Advanced Compression には、SecureFiles データのサイズを抑制する別のメカニズムもあります。高度な LOB 圧縮は、業界標準の圧縮アルゴリズムを使用して、SecureFiles データに必要なストレージをさらに削減します。高度な LOB 圧縮を使用すると、ドキュメントや XML ファイルなどのファイルは通常、2~3 倍の圧縮率になります。

高度な LOB 圧縮は、圧縮しても利点のないデータの圧縮を自動的に回避します。たとえば、SecureFiles ファイルとしてデータベースに挿入される前に、サード・パーティのツールを用いて圧縮されたドキュメントです。圧縮データはデータの小さなチャンクに内部的に分類されるため、アプリケーションは圧縮された SecureFiles データ上でもランダム読取りおよびランダム書き込みを実行できます。これにより、データベースへの挿入前にファイル全体を圧縮する場合と比較して、パフォーマンスが大幅に向上します。

高度な LOB 圧縮には、LOW、MEDIUM、HIGH の 3 つのレベルがあります。デフォルトでは、高度な LOB 圧縮では MEDIUM レベルが使用されます。MEDIUM レベルでは通常、3%~5%という少ない CPU オーバーヘッドで適度な圧縮が実行されます。高度な LOB 圧縮の LOW は、高パフォーマンス向けに最適化されています。高度な LOB 圧縮の LOW では、通常 1/3 の CPU 使用量で、MEDIUM によって達成される圧縮の約 80% が維持されます。高度な LOB 圧縮の HIGH では、ストレージの節約が最大になりますが、CPU オーバーヘッドも最大になります。

## バックアップ圧縮

Oracle Advanced Compression には、データベース内に保存されたデータの圧縮に加えて、バックアップされたデータを圧縮する機能も含まれています。Oracle Recovery Manager (Oracle RMAN) と Data Pump は、Oracle Database 内に保存されたデータのバックアップにもっとも一般的に使用されるツールです。

Oracle RMAN は、データベース・データのブロックごとのバックアップを実行します。このバックアップは“物理バックアップ”とも呼ばれ、データベース、表領域、またはブロック・レベルのリカバリを行うために使用できます。Data Pump は、1 つ以上の表からフラット・ファイルへデータをオフロードすることによって、“論理”バックアップを実行するために使用されます。

Oracle Advanced Compression には、これら 2 つのツールによって作成されたバックアップ・データを圧縮する機能が含まれています。

## Oracle Recovery Manager (Oracle RMAN) による圧縮

エンタープライズ・データベースが増大し続けることにより、データベース管理者にとって重大な問題が生じています。データベース・バックアップを維持するためのストレージ要件およびバックアップ・プロシージャのパフォーマンスは、データベースのサイズに直接影響を受けます。Oracle Advanced Compression には、バックアップ・データのストレージ要件を大幅に削減できる RMAN 圧縮テクノロジーが含まれます。

Oracle RMAN が Oracle Database と密接に統合されていることにより、バックアップ・データはディスクまたはテープに書き込まれる前に圧縮され、リカバリ前に解凍する必要がありません。そのため、ストレージ・コストが大幅に削減されます。また、バックアップとリストアにかかる時間も大幅に短縮される可能性があります。

高度な圧縮に含まれる Oracle RMAN 圧縮には、LOW、MEDIUM、HIGH の 3 つのレベルがあります。一般に、これら 3 つのレベルは次のように分類されます。

- **HIGH** - ネットワーク速度に制約がある、速度が遅いネットワークを介したバックアップに最適
- **MEDIUM** - ほとんどの環境に推奨。圧縮比と速度のバランスに優れている
- **LOW** - バックアップ・スループットに一番影響が少なく、CPU リソースに制約がある環境に最適

上記に示すように、I/O に制約があるが CPU がアイドル状態の場合は、HIGH が最適な場合があります。HIGH の場合、CPU をより多く使用しますが、領域をもっとも多く節約するので、バックアップ・ファイルの書き込みに必要な I/O 数をもっとも少なくなります。その一方で、CPU に制約がある場合は、LOW または MEDIUM がより最適かもしれません。CPU の使用量が少なめで、通常、約 80% の領域が節約されます (Basic と比較した場合)。

## Oracle Data Pumpによる圧縮

Data Pump ジョブに関連するメタデータの圧縮機能が最初に導入されたのは、Oracle Database 10g Release 2 でした。Oracle Database 11g では、この圧縮機能は、表データをエクスポートで圧縮できるように拡張されています。この拡張機能は、Oracle Advanced Compression の機能の 1 つです。

Data Pump による圧縮は、インライン操作であるため、ダンプ・ファイル・サイズが削減されることにより、ディスク領域が大幅に節約されます。オペレーティング・システムまたはファイル・システムの圧縮ユーティリティとは異なり、Data Pump 圧縮は、インポート側でも完全にインラインであるため、ダンプ・ファイルをインポートする前に解凍する必要がありません。圧縮されたダンプ・ファイル・セットは、データベース管理者が手順を追加することなく、インポート中に自動的に解凍されます。

Data Pump の機能は、圧縮ファイルを使用することによって、完全に利用できます。通常のファイルで使用されるコマンドは、圧縮ファイル上でも動作します。以下のオプションを使用して、ダンプ・ファイル・セットのどの部分を圧縮するかを指定します。

- **ALL** - エクスポート操作全体に対して、圧縮が有効化されます。
- **DATA-ONLY** - 圧縮された形式で、すべてのデータがダンプ・ファイルに書き込まれます。
- **METADATA-ONLY** - 圧縮された形式で、すべてのメタデータがダンプ・ファイルに書き込まれます。これがデフォルトです。
- **NONE** - エクスポート操作全体に対して、圧縮が無効化されます。



Oracle Data Pump エクスポートの expdp コマンドライン・オプションを使用して、Oracle Data Pump ダンプ・ファイルに対して使用する圧縮の度合い（BASIC、LOW、MEDIUM、または HIGH）を制御できます。PL/SQL DBMS\_DATAPUMP パッケージにも同じオプションを指定できます。

圧縮の度合いが強いほど、待機時間も長くなりますが、圧縮率が高くなります。つまり、HIGH オプションでは、より多くのオーバーヘッドが生じますが、データの圧縮率が高くなります。DBA は、これらのオプションを使用して、データの圧縮にかかる時間と Oracle Data Pump ダンプ・ファイルのサイズとのバランスを図ることができます。

ダンプ・ファイルの削減サイズは、データ型や他の要素によって異なります。Data Pump を使用してインポートする際には、CREATE TABLE 文に、エクスポート・ファイル内の定義と一致する圧縮関連の句を含める点に注意してください。圧縮関連の句を指定していない場合、その表では、表が保存されている表領域の COMPRESSION 属性が継承されます。

## 索引圧縮

索引は、リレーショナル表に格納されたデータへのさまざまなアクセス・パスを効率的にサポートできるため、OLTP データベース内で幅広く使用されています。OLTP アプリケーションの多数のアクセス・パスをサポートするために、大量の索引が1つの表に作成されることがよくあります。そのため、ベース表単体のサイズと比べて、データベースのストレージ全体に占める索引の割合が大きくなる場合があります。

高度な索引圧縮は、索引インデックス圧縮の形式です。高度な索引圧縮を使用して索引を作成すると、インデックスへの効率的なアクセスを提供しつつ、サポートされているすべての一意索引と非一意索引のサイズを縮小できます。高度な索引圧縮は、サポート対象のすべての索引に対して機能します。既存のプリフィックス圧縮機能で適切な候補にならない索引（索引の主要な列に重複値がほとんど、またはまったくない索引）に対しても有効です。

高度な索引圧縮は、ブロック・レベルで行われるため、各ブロックに対して最適な圧縮が行われます。高度な索引圧縮によってブロックごとに適切な圧縮が自動で選択されるため、データの特性に関する知識は必要ありません。高度な索引圧縮の"HIGH"レベルにより領域が大幅に節約される一方で、索引を使って実行される問合せのパフォーマンスも向上します。

HIGH 圧縮レベルは、LOW 圧縮レベルに比べ、次の利点をもたらします。

- ほとんどの場合でより高い圧縮率を達成
- 高度な LOW レベル圧縮よりも複雑な圧縮アルゴリズムを採用
- 特殊なオンディスク形式である圧縮単位でデータを保管

## ネットワーク圧縮

高度なネットワーク圧縮は、SQL ネットワーク・データ圧縮とも呼ばれ、送信側で送られるネットワーク・データを圧縮し、その後受信側で解凍して、ネットワーク・トラフィック量を削減する目的で使用できます。高度なネットワーク圧縮により、データ接続経由で送信されるセッション・データ・ユニット（SDU）のサイズが削減されます。データのサイズが削減されることで、SDU の送信にかかる時間が短縮されます。

高度なネットワーク圧縮の利点は、以下のとおりです。

- **有効ネットワーク・スループットの向上：**圧縮により、大きなデータの送信にかかる時間を短縮できます。送信時間が短縮されるため、SQL 問合せの応答が速くなります。帯域幅が限られた環境では、この点を利用して、問合せの応答時間を短縮できます。
- **帯域幅利用率の軽減：**圧縮によって、送信するデータ量が削減され、帯域幅が節約されるため、解放された帯域幅を他のアプリケーションで利用できます。また、ネットワーク帯域幅にかかるコストの削減にもつながります。

高度なネットワーク圧縮は、SQL 問合せの応答を速くするだけでなく、帯域幅も節約します。帯域幅接続が狭く、CPU が高速な場合、パフォーマンスを大幅に改善できる可能性があります。圧縮処理は、クライアント・アプリケーションに対して透過的に実行されます。

## Oracle Data Guard REDO 転送の圧縮

Oracle Data Guard は、管理、監視、自動化のためのソフトウェア・インフラストラクチャで、企業のデータを故障、障害、エラー、破損から保護するために、1 つまたは複数のスタンバイ・データベースを作成、保守、および監視します。Oracle Data Guard は、REDO データ（トランザクションのリカバリに必要な情報）を使用して、プライマリ・データベースおよびスタンバイ・データベースの同期を維持します。プライマリ・データベースでトランザクションが発生すると、REDO データが生成され、ローカル REDO ログ・ファイルに書き込まれます。

Oracle Data Guard REDO 転送サービスは REDO データをスタンバイ・サイトに転送します。Oracle Advanced Compression を使用すると、REDO データは圧縮された形式で送信されて、ネットワーク帯域幅の消費が減少します。REDO データの送信時間が減少する場合があります。Oracle Data Guard 構成で、同期 REDO 転送（SYNC）または非同期 REDO 転送（ASYNC）のいずれかが使用される場合、REDO データは圧縮された形式で送信できます。

## 情報ライフサイクル管理

データベース内のすべてのデータのビジネス要件が同じというわけではありません。

データはその情報ライフサイクルのさまざまな段階をたどります。データはまず、アクティブ・データとして始まります。アクティブ・データは初回の作成後、頻繁に問合せや変更が行われるデータです。このようなデータは、高度な圧縮の候補として適しています。ある程度の期間が過ぎたデータは通常、以前よりもアクティブ度が低下します。これは、レポート生成などの目的で、ある期間は頻繁に問合せが行われますが、変更はほとんど発生しないデータです。このようなデータは、Hybrid Columnar Compression（問合せ）のウェアハウス圧縮の候補として適しています。最終段階のデータはほぼ休止状態になります。これは、更新されなくなり、問合せが行われるとしても頻度が非常に低くなりますが、規制遵守の目的で保管する必要のあるデータです。このようなデータは、Hybrid Columnar Compression のアーカイブ圧縮を使用して圧縮できます。



Oracle Database 19c では、Oracle Advanced Compression の複数の機能が Oracle Database の情報ライフサイクル機能を強化します。ヒート・マップは、変更および問合せのタイムスタンプを自動的に追跡します。そのため、データがどのようにアクセスされているかについて詳細に把握できます。自動データ最適化（ADO）は、ヒートマップによって収集された情報に基づいて、データの移動と圧縮を自動的に行います。これらの機能を組み合わせて、自動化した情報ライフサイクル管理（ILM）戦略の実施に役立てることができます。

## ヒート・マップ

ヒートマップは、利用情報をブロック・レベルとセグメント・レベルで収集する Oracle Advanced Compression の機能です。ヒート・マップを自動データ最適化（以下の「自動データ最適化」の項を参照）と併せて使用することで、Oracle Database では、データの使用状況に基づいて圧縮ポリシーや保管ポリシーを自動化できるため、ストレージ・コストの削減、パフォーマンスの向上、ストレージの最適化を実現できます。

セグメント・レベルのヒートマップは、データベース内の各表および各パーティションに対する最新の変更や問合せのタイムスタンプを追跡します。ブロック・レベルのヒートマップは、最新の変更のタイムスタンプを追跡します。これらのタイムスタンプは、自動データ最適化により、データのライフサイクル全体を通じて自動的に維持される圧縮ポリシーと保管ポリシーを定義するために使用されます。ヒート・マップでは、システム・タスクに対して実行された内部的な操作は無視され、統計情報収集、DDL、表再定義などの操作は自動的に除外されます。さらに、ヒートマップはセッション・レベルで無効化することも可能です。そのため、手作業によるメンテナンスを DBA が除外して、ヒートマップ・データの汚染を防止できます。

Oracle Database は、ヒート・マップにより収集されたデータを使用して、ヒート・マップ・データに基づいて、個別に各パーティション/表を自動圧縮できます。これにより、圧縮階層化が実現し、現在の圧縮レベルが現在の表/パーティションの使用状況に最適になるようにします。この圧縮階層化では、すべての Oracle データ圧縮形式を使用できます。基盤となるストレージで HCC がサポートされている場合は、高度な行圧縮と Hybrid Columnar Compression（HCC）の全レベルを使用できます。また、Oracle Database では、ヒートマップ・データに基づいて個々のデータベース・ブロックを圧縮することもできます。

## 自動データ最適化

自動データ最適化（ADO）では、データ圧縮とストレージ階層化を自動化するポリシーを作成できます。ADO ポリシーは、特定のオブジェクトに適用する条件および対応する処理を定義します。Oracle Database は、ヒート・マップによって維持される情報を利用して、リクエストされたオブジェクトの登録済み ADO アクションを実行し、オブジェクトを目的の状態へと透過的かつ自動的に移行します。

- ADO ポリシーは、表またはパーティションに対してセグメント・レベルまたは行レベルで指定できます。
- セグメントおよび行レベルの ADO ポリシーは、メンテナンス期間にバックグラウンドで自動的に評価および実行されます。または、オンデマンドで実行することもできます。
- ストレージ階層化（Tier To）はセグメント・レベルで指定し、セグメントが現在存在する表領域内の領域の圧縮によってトリガーされます。

ADO ポリシーの仕様は次のとおりです。

- 圧縮を起動する条件 -- アクセスなし、変更なしなど。カスタム条件は、特定のビジネス要件に基づいて作成できます。
- ポリシーが適用されるタイミング -- 変更なしの状態が"x"日（あるいは数か月、数年）続いた後、表 / パーティションを作成してから"x"日後、あるいはオブジェクトを含む表領域が事前定義済みの表領域使用率のしきい値を満たしたときなど。

#### ADO ポリシーの例：

以下の最初の例では、高度な行圧縮を使用して、30 日以上変更されていない表全体を自動的に圧縮するためのセグメント・レベルの ADO ポリシーを作成します。

```
ALTER TABLE employee ILM ADD POLICY ROW STORE COMPRESS ADVANCED SEGMENT AFTER 30 DAYS OF NO MODIFICATION;
```

次の例では、高度な行圧縮を使用して、ブロック内のどの行も 3 日以上変更されていない場合に、表内のブロックを自動的に圧縮するための行レベルの ADO ポリシーを作成します。

```
ALTER TABLE employee ILM ADD POLICY ROW STORE COMPRESS ADVANCED ROW AFTER 3 DAYS OF NO MODIFICATION;
```

その他の ADO ポリシーのアクションには、低コストのストレージ層や Hybrid Columnar Compression (HCC) などの他の圧縮機能を使用するストレージ層など、別のストレージ層 (TIER TO) へのデータ移動もあります。HCC を使用するには、Oracle Storage (Exadata、SuperCluster、Pillar Axiom、Sun ZFS Storage Appliance (ZFSSA)、FS1、または Oracle Database Appliance (ODA)) を使用する必要があります。

次の例では、表領域レベルの ADO ポリシーにより、現在オブジェクトを含む表領域が事前定義済みの表領域使用率のしきい値を満たす場合に、表を別の表領域に自動的に移動します。

```
ALTER TABLE employee ILM ADD POLICY tier to lowcost;
```

セグメントを別の表領域に移動する場合、オブジェクトの移動後にターゲットの表領域を READ ONLY に設定するオプションもあります。このオプションは、データベース・バックアップ中に履歴データに対して使用すると便利です。その後のデータベースのフル・バックアップでは、READ ONLY の表領域はスキップされます。

## その他の機能

以下は、Oracle Advanced Compression に搭載されているその他の機能です。

## Flashback Data Archive 履歴表の最適化

Flashback Data Archive は、表のライフサイクル全体にわたって、トランザクションの変更点を追跡し、保存できます。Flashback Data Archive は、レコード・ステージ・ポリシーへの準拠や監査レポートに非常に役立ちます。Oracle Database の Flashback Data Archive 機能 (旧称は、Oracle Database 11g の Total Recall 機能) は、本番データベースの変更を追跡するメカニズムを備えています。このメカニズムはセキュアで、効率がよく、使いやすく、アプリケーション側から特に意識することなく利用できます。Flashback Data Archive によってアーカイブされたデータは、履歴表で維持されます。Flashback Data Archive 履歴表機能の最適化によって、圧縮する履歴表を指定できます。

## ストレージ・スナップショット最適化

Recovery Manager では、Oracle Database バックアップを実行するためのもっとも一般的な方法が残されていますが、データベース・バックアップを行うための方法は他にもあります。それは、データベース内のすべてのファイル（データファイル、制御ファイル、オンライン REDO ログ）について、ストレージ・スナップショットを作成し、そのスナップショットを別のシステム（本番データベースが稼働するシステム以外）にマウントし、データをその 2 次システムから、テープなどの 3 次ストレージにコピーするというものです。

ストレージ製品が、Oracle ドキュメントで説明している固有のガイドラインに従っているとすると、この方法で取得したスナップショットには“クラッシュ一貫性”があります。そのようなスナップショットをリストアした場合、Oracle では、そのスナップショットと、スナップショットが取得された時点でクラッシュしたデータベースとを区別できません。クラッシュ一貫性のあるバックアップは、オープンして、標準的なクラッシュ（インスタンス）リカバリを実施した後に使用できます。ただし、クラッシュ一貫性のあるバックアップは、ポイント・イン・タイム・リカバリに使用するものとしては信頼できません。データファイルの非一貫性の除去を行うためのメディア・リカバリに必要な情報が、REDO ログに含まれていないためです（スナップショットの取得時に、データベースが書き込み用にオープンされていたため）。ポイント・イン・タイム・リカバリを実行するには、要件に従い、さらに、Support Note 604683.1 で説明している手動の手順に厳格に従う必要があります。

または、バックアップ・モードで取得したスナップショット（ALTER DATABASE [BEGIN | END] BACKUP）では、そのようなポイント・イン・タイム・リカバリの制約はなくなります。リカバリの実行時にデータファイルの非一貫性を除去するための追加情報が、REDO ログに書き込まれるためです。スナップショットを取得する前にそれぞれのデータベースをこのモードに設定し、スナップショットの完了時にこのモードを終了する必要があります。数十、数百、あるいは数千ものデータベースに対してこの手順を実行しなければならない場合には、この複雑性が増します。また、このモードの実行中、ブロック・イメージが変更されると、ブロック・イメージ全体が REDO ログに書き込まれ、追加の I/O アクティビティが発生します。同じアレイで多数のデータベースが稼働している場合に、このオーバーヘッドが増加します。

Oracle Database の RECOVER .. SNAPSHOT TIME コマンド（Oracle Advanced Compression で利用可能）によって、バックアップ・モードのデータベースなしに取得したストレージ・スナップショットを、1 つの手順でリカバリできます。この機能は、現時点へのリカバリにも、スナップショット取得後の特定の時点へのリカバリにも対応しており、追加の手順は必要ありません。この最適化では、これらのスナップショットを使用するあらゆるタイプのリカバリ操作をサポートすることで、バックアップ・モードやそれに関連する複雑性とオーバーヘッドは事実上なくなり、DBA は、より重要な本番タスクに時間を費やせるようになります。

## Hybrid Columnar Compressionの行レベル・ロック

オラクルの Hybrid Columnar Compression（HCC）テクノロジーは、一連のデータベース・ブロック内にデータを構成するための手法です。HCC では、行を使用した手法と列を使用した手法を組み合わせることでデータを格納します。圧縮単位（CU）と呼ばれる論理的な構成体は、HCC で圧縮された一連のデータを格納します。データがロードされると、行グループが列形式で保存され、特定の列の値が一緒に格納されて圧縮されます。行セットの列データは、圧縮されてから圧縮単位に格納されます。

Hybrid Columnar Compression は、1CU につき 1 ロック使用します。オプションで、圧縮単位の行レベル・ロックを有効化することもできます。HCC でのデフォルト設定は、NO ROW LEVEL LOCKING で、ROW LEVEL LOCKING は、CREATE TABLE または ALTER TABLE MOVE 操作時に明示的に指定します。

## Exadata Flash Cache Compression

Exadata Flash Cache Compression は、フラッシュ・キャッシュにロードされるユーザー・データを透過的に圧縮することによって、フラッシュ・キャッシュの論理容量を動的に増加させます。これにより、フラッシュ内に大量のデータを保持できるため、ディスク・ドライブ上のデータにアクセスする必要性が減少します。フラッシュ内のデータへの I/O は、ディスク上のデータへの I/O と比べて格段に速い実行速度です。圧縮および圧縮解除操作はアプリケーションやデータベースに対して完全に透過的に実行されます。また、1 秒あたり数百万 I/O の速度で動作している場合であってもパフォーマンス・オーバーヘッドは発生しません。

ユーザー・データの圧縮率によって異なりますが、Oracle Exadata Storage Server Software は、フラッシュ・キャッシュを動的に最大で 2 倍に拡大します。圧縮による効果は、データ内の冗長性によって異なります。圧縮されていない表と索引では、領域を最大に削減できます。OLTP 表圧縮を使用した表と索引では、大幅に領域を削減できます。Hybrid Columnar Compression を使用する表では、領域の削減は最小になります。

## 表/パーティション/サブパーティションのオンラインでの移動（圧縮形式へ）

ALTER TABLE ... MOVE TABLE/PARTITION/SUBPARTITION ... ONLINE により、同時 DML 操作をブロックすることなく、表、パーティション、またはサブパーティションをオンライン操作として移動できます。パーティション/サブパーティションの移動操作では、移動の一部として自動的な索引メンテナンスがサポートされるため、索引の再構築は不要です。MOVE 操作の“online”機能は、Oracle Advanced Compression に付属しています。

## 結論

企業が直面しているデータ量の急増は、重大な課題をもたらしています。企業は、収益に影響を与えることなく変化の激しいビジネス状況に迅速に適応する必要があります。IT マネージャーは、既存のインフラストラクチャを効率的に管理してコストを制御しながら、優れたアプリケーション・パフォーマンスを提供し続けていく必要があります。

Oracle Advanced Compression と Oracle Database はともに、この複雑な環境で、IT マネージャーを成功に導く、一連の堅牢な圧縮、パフォーマンスおよびデータ・ストレージ最適化機能を提供します。

企業は、Oracle Advanced Compression を利用することにより、データセンターのすべてのコンポーネントを通して、増大するデータ要件を効率的に管理でき、最高レベルのアプリケーション・パフォーマンスを実現しながら設備投資と運用コストを最小限に抑えられます。

## 免責事項

下記事項は、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。マテリアルやコード、機能の提供をコミットメント（確約）するものではなく、購買を決定する際の判断材料になさらないでください。オラクルの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®**

Oracle Corporation, World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065, USA

海外からのお問い合わせ窓口  
電話：+1.650.506.7000  
ファクシミリ：+1.650.506.7200

CONNECT WITH US

 [blogs.oracle.com/oracle](https://blogs.oracle.com/oracle)  
 [facebook.com/oracle](https://facebook.com/oracle)  
 [twitter.com/oracle](https://twitter.com/oracle)  
 [oracle.com](https://oracle.com)

### Hardware and Software, Engineered to Work Together

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. 本文書は情報提供のみを目的として提供されており、ここに記載されている内容は予告なく変更されることがあります。本文書は、その内容に誤りがないことを保証するものではなく、また、口頭による明示的保証や法律による黙示的保証を含め、商品性ないし特定目的適合性に関する黙示的保証および条件などのいかなる保証および条件も提供するものではありません。オラクルは本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクルの書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

Oracle および Java は Oracle およびその子会社、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です。

Intel および Intel Xeon は Intel Corporation の商標または登録商標です。すべての SPARC 商標はライセンスに基づいて使用される SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴおよび AMD Opteron ロゴは、Advanced Micro Devices の商標または登録商標です。UNIX は、The Open Group の登録商標です。0219



Oracle is committed to developing practices and products that help protect the environment