



ORACLE

Oracle Data Pumpの ベスト・プラクティス



Oracle Databaseユーティリティ

2021年8月 | バージョン2.00

Copyright © 2021, Oracle and/or its affiliates

本書の目的

本書では、Oracle Data Pumpを使用してオンプレミスからオンプレミス/Oracle CloudにOracleデータベースを移行する際に推奨されるベスト・プラクティスについて説明します。本書は、Oracle Databaseへのアップグレードによる利点の評価とITプロジェクトの計画を支援することのみを目的としています。

免責事項

本文書には、ソフトウェアや印刷物など、いかなる形式のものも含め、オラクルの独占的な所有物である占有情報が含まれます。この機密文書へのアクセスと使用は、締結および遵守に同意したOracle Software License and Service Agreementの諸条件に従うものとします。本文書と本文書に含まれる情報は、オラクルの事前の書面による同意なしに、公開、複製、再作成、またはオラクルの外部に配布することはできません。本文書は、ライセンス契約の一部ではありません。また、オラクル、オラクルの子会社または関連会社との契約に組み込むことはできません。

本書は情報提供のみを目的としており、記載した製品機能の実装およびアップグレードの計画を支援することのみを意図しています。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するものでもありません。本書に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、オラクルの裁量により決定されます。

製品アーキテクチャの性質上、コードが大幅に不安定化するリスクなしに、本書に記載されているすべての機能を安全に含めることができない場合があります。

内容

本書の目的	1
免責事項	1
はじめに	3
SYS as SYSDBAを使用してエクスポートを起動しない	3
パラメータ・ファイルを使用する	3
Data Pumpエクスポートの一貫性を確保する	4
エクスポートとインポートから統計情報を除外する	4
エクスポートおよびインポート中に診断パラメータを含める	4
並列処理と現在の統計情報を使用してパフォーマンスを向上する	5
リソース使用率を適切に設定する	6
異なるOSおよびストレージ上のサーバーにネットワーク・リンクを使用する	7
SecureFile LOBを使用する	7
データベース互換性を設定する	7
スペース使用とパフォーマンスのためにエクスポートで圧縮を使用する	7
Data Pumpジョブ実行前のデータベース・チェック	8
Oracle Cloudデータベース向けのその他のプラクティス	9
結論	9



はじめに

オンプレミスのOracleデータベースを移行する手法はいくつもあります。一般的な方法の1つは、Oracle Data Pumpの使用です。Oracle Data Pumpは、Release 9i以前のExportおよびImportユーティリティ（exp/imp）の後継であり、Release 10g以降ではOracle Databaseの機能になっています。Oracle Data Pumpは、スキーマ間、バージョンの異なるデータベース間、異なるオペレーティング・システム間や、オンプレミスからオンプレミスまたはOracle Cloudへデータを移行する際に役立ちます。以下のような状況で、オンプレミスからOracle CloudにOracleデータベースを移行する場合、Oracle Data Pumpを検討することをお勧めします。

- ソースのOracle Databaseのリリースが10g以上である（古いデータベースの場合は2段階の作業が必要で、まずデータベースに付属する従来のExportおよびImportユーティリティで10gに移行し、次にData Pumpを使用）
- 異なるエンディアン間でデータを移行する
- 非CDBのOracleデータベースからOracle Multitenantに移行する
- 移行時にデータベース構造を変更する（暗号化、圧縮、キャラクタ・セット、IOTからヒープ表、表領域またはスキーマの再マッピング、BasicFile LOBからSecureFile LOBへの変換など）
- 移行時にオブジェクトを含めるか、除外する
- 移行とバージョン・アップグレードを同時に実行する

Oracle Data Pumpを使用してデータを移行するには、次の3つのステップのプロセスがあります。クラウド・インスタンスとPDBが作成されていることが前提です。

- expdpを使用して、オンプレミス・データベースをエクスポート
- ダンプ・ファイルをターゲット・システムまたは必要に応じてOracle Object Storeにコピー
- impdpを使用して、クラウドPDBにインポート
- または、dblinkを使用して上記ステップをネットワーク・モードのインポートに集約すると、ダンプ・ファイルが不要になります。

この技術概要では、Oracle Data Pump使用時に、プロセスを円滑かつ順調に進めるのに役立つベスト・プラクティスを紹介します。

SYS as SYSDBAを使用してエクスポートを起動しない

SYSDBAは内部で使用される特別な関数を持つユーザーであり、一般ユーザーとは動作が異なります。

オラクルのテクニカル・サポートから要求された場合とトランスポータブル表領域セットをインポートする場合を除き、通常はSYSDBAとしてエクスポートを開始しないでください。

パラメータ・ファイルを使用する

“parfile”とも呼ばれるパラメータ・ファイルを使用すると、ファイルにコマンドライン・パラメータを指定して簡単に再利用できます。また、特に値に引用符を必要とするパラメータを使用する場合に、コマンドラインに長い非定型のData Pumpコマンドを入力することによるタイプミス回避できます。

parfileの例は次のとおりです。

```
DIRECTORY=my_data_pump_dir
DUMPFILE=dumpfile.dmp
LOGFILE=logfile.log
SCHEMAS=HR
EXCLUDE=STATISTICS
LOGTIME=ALL
METRIC=YES
FLASHBACK_TIME=SYSTIMESTAMP
```

parfileを実行するためのコマンドは次のようになります。

```
expdp parfile=my_data_pump_parfile.par
```

Data Pumpエクスポートの一貫性を確保する

デフォルトでは、Oracle Data Pumpは単一のデータベース表内で一貫性を維持します。たとえば、1,000のパーティションを含む1つの表をエクスポートする場合、エクスポートされた表は、エクスポートを開始した特定のシステム変更番号（SCN）の時点で一貫性が維持されます。複数の表をエクスポートする場合は、エクスポートされた次の表は異なるSCNの時点で一貫性を維持します。2つ以上の表のエクスポートの場合は、同じSCNの時点のすべてのオブジェクトを表すためにエクスポート・ダンプ・ファイルが必要になる可能性があります。

その場合は、FLASHBACK_SCN=<scn>またはFLASHBACK_TIME=<timestamp>を使用してフラッシュバック問合せユーティリティを有効にします。特に便利なのは、FLASHBACK_TIME=SYSTIMESTAMPを指定する方法です。

FLASHBACK_SCNを使用すると、エクスポート操作では、指定されたSCNまでの一貫性が維持されるデータが使用されます。たとえば、以下のコマンドは既存のSCN値384632が存在することを前提としています。これによって、hrスキーマはSCN 384632までエクスポートされます。

```
expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_scn.dmp FLASHBACK_SCN=384632
```

FLASHBACK_TIME=<timestamp>を使用すると、エクスポート操作では、指定された時刻に最も近いSCNまでの一貫性が維持されるデータが使用されます。たとえば、以下のエクスポート操作は、指定した時間にもっとも近いSCNまで一貫性が維持されるデータを用いて実行されます。

```
FLASHBACK_TIME="TO_TIMESTAMP('27-10-2012 13:16:00', 'DD-MM-YYYY HH24:MI:SS')"
```

FLASHBACK_TIME=SYSTIMESTAMPを指定すると、タイムスタンプは現在のシステム時刻になります。最後に、Release 11.2のレガシー・インタフェースであるCONSISTENT=Yを引き続き使用できます。これは、FLASHBACK_TIME=SYSTIMESTAMPに直接変換されます。

注：FLASHBACK_TIMEまたはFLASHBACK_SCNを使用する場合、エクスポートの継続中は、Data PumpエクスポートにUNDOレコードが保持されていなければなりません。データベースのUNDO保存期間が十分ではない場合、データベースで使用できなくなったUNDOレコードにデータベースがアクセスしようとすると、“snapshot segment too old”エラーが発生します。

注：Oracle Data Guardスタンバイ・データベースを一時スナップショットに変換して、エクスポートを実行することもできます。Data Pumpでは、エクスポートを調整するための表の作成が必要であり、これには読取り/書込みインスタンスが必要です。Active Data GuardとDMLリダイレクトで使用できるのはDMLのみで、CREATE TABLEのDDLコマンドは使用できないため、現時点ではこの方法に対応していません。

エクスポートとインポートから統計情報を除外する

Oracleでは、エクスポート中に統計情報をエクスポートしないことを推奨します。これによって、たとえインポート後に統計情報を収集する必要性があるとしても、エクスポートとインポートの両方のパフォーマンスが向上します。エクスポート操作から統計情報を除外するには、EXCLUDE=STATISTICSパラメータを使用するか、トランスポート表領域にEXCLUDE=TABLE_STATISTICS,INDEX_STATISTICSを指定します。この場合、ベスト・プラクティスとして、ターゲット・データベースで新しい統計情報を作成するか、統計情報をトランスポートするためのDBMS_STATSステージング表を使用することをお勧めします。

エクスポートおよびインポート中に診断パラメータを含める

LOGTIME=ALLを使用して、エクスポート操作中に表示されるメッセージにタイムスタンプを付けます。このパラメータはOracle Database Release 12.1以降で使用できます。エクスポートおよびインポートのパフォーマンスを評価する際に、ログ・ファイルのすべての行にタイムスタンプがあると役に立ちます。

オブジェクトの数およびジョブに関する経過時間をOracle Data Pumpログ・ファイルに記録します。それには、METRICS=YESパラメータを使用します。これにより、パラレルでのエクスポート/インポートで各プロセスにより実行された処理など、さらに詳しい情報が得られます。

LOGTIME=ALLを使用して、インポート操作中に表示されるメッセージにタイムスタンプを付けます。このパラメータはOracle Database Release 12.1以降で使用できます。

METRICS=YESパラメータを使用して、オブジェクトの数およびジョブに関する経過時間をOracle Data Pumpログ・ファイルに記録します。

例：

```
No を指定した場合の診断情報
Processing object type SCHEMA_EXPORT/TABLE/TABLE
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
. . imported "METAL"."ALBUMS"          988.8 KB   28069 rows
. . imported "METAL"."BANDS"          3.444 MB  37723 rows
. . imported "METAL"."REVIEWS"        66.47 MB  21510 rows

ALL を指定した場合の診断情報
16-OCT-20 17:26:57.158: Processing object type SCHEMA_EXPORT/TABLE/TABLE
16-OCT-20 17:26:58.262: Startup took 1 seconds
16-OCT-20 17:26:58.264: Startup took 1 seconds
16-OCT-20 17:26:59.082:      Completed 3 TABLE objects in 1 seconds
16-OCT-20 17:26:59.082:      Completed by worker 1 1 TABLE objects in 1 seconds
16-OCT-20 17:26:59.082:      Completed by worker 2 1 TABLE objects in 0 seconds
16-OCT-20 17:26:59.082:      Completed by worker 3 1 TABLE objects in 0 seconds
16-OCT-20 17:26:59.313: Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
16-OCT-20 17:27:01.943: . . imported "METAL"."ALBUMS"          988.8 KB   28069 rows in 2 seconds using external_table
16-OCT-20 17:27:03.778: . . imported "METAL"."BANDS"          3.444 MB  37723 rows in 2 seconds using external_table
16-OCT-20 17:27:12.644: . . imported "METAL"."REVIEWS"        66.47 MB  21510 rows in 13 seconds using external_table
```

並列処理と現在の統計情報を使用してパフォーマンスを向上する

並列処理の使用

並列処理を使用して、より短い時間でより多くの作業を実行します。新しいData Pumpジョブは、2つ以上のバックグラウンド・プロセス（制御プロセスとワーカー・プロセス）と2つのセッションで構成されています。Data PumpのPARALLELパラメータにより、エクスポート/インポート中に追加のバックグラウンド・プロセスとセッションが作成されます。

PARALLEL=nパラメータは、エクスポート/インポート・ジョブに代わって動作するアクティブ実行の最大プロセス数を指定します。通常、nにはCPUコア数の2倍の値を指定しますが、必要に応じて調整する必要があります。

Oracle Data Pumpのいくつかのリリースでは、PARALLELパラメータにより、パッケージ本体を並列でインポートしていました。Release 12.2以上のダンプ・ファイルのみによるエクスポートおよびインポートで、Oracle Data Pumpは、ほとんどのメタデータとほとんどのデータベース・オブジェクトを並列インポートします。PARALLELパラメータにより、並列で作成される索引の数も決まります。タイプ（継承のため）、スキーマ、プロシージャのアクションなど、依存関係のあるデータベース・メタデータ・オブジェクトは引き続き順次インポートされます。

Oracle Database Release 11.2.0.4または12.1.0.2を使用している場合、バグ22273229用のパッチを適用すると、制約と索引の並列インポートを有効にできます。

注：Data Pumpは2種類の並列処理を使用します。サブパーティションと小さい表には、パーティションまたは表ごとに1つのワーカーを使用します（表間の並列処理）。大規模パーティションやパーティション化されていない大きな表では、1つのパラレル問合せのPXプロセス（旧称PQスレーブ）を使用します（表内の並列処理）。

ワイルドカードの指定による複数ダンプ・ファイルの作成

並列処理を使用する場合、ダンプ・ファイル名の指定時に置換変数%Uまたは%Lを使用します。これにより、同時に複数のログ・ファイルに並列で書き込むことができます。また、ダンプ・ファイルのサイズを小さくして複数にすると管理とコピーが容易になります。

例としては次のようなものがあります。

1から99までのファイルで、%Uを使用（ファイル名の数字は固定幅の2桁、増分する整数）

```
dumpfile=dumpfile%U.dmp
filesize=n
```

1から99を超えるファイルで、%Lを使用（ファイル名の数字は3～10桁、可変幅の整数）

```
dumpfile=dumpfile%L.dmp  
filesize=n
```

これらを指定しない場合、1つのログ・ファイルに平行書き込みが行われる結果となり、ダンプ・ファイルに書き込むそれぞれのワーカーがファイルに排他ロックをかけて、他のワーカー・プロセスを阻止するため、パフォーマンスに影響が及び場合があります。

Data Pump処理前後での正確な統計情報の収集

エクスポート処理の前に正確な統計情報を取得しておく、可能な限り最大のエクスポート・パフォーマンスを実現するのに役立ちます。また、インポート後の統計情報収集にも有用です。最新の統計情報を維持するために必要な統計情報の更新頻度は、データベースがどの程度頻繁に変更されるかによって異なります。統計情報の収集には、ディクショナリ統計とオブジェクト統計の両方が含まれます。ディクショナリ統計は、エクスポートの各種段階で、ダンプ・ファイルがメタデータをフィルタ、順序付け、収集するときに使用されます。オブジェクト統計は表および索引サイズの見積りに使用され、順序付けと並列処理を最適化するのに役立ちます。

メタデータのエクスポートと同時に、平行エクスポートのために表サイズが予測され、最大から最小までランク付けされます。表サイズの予測は、統計情報を使用して行われます。統計情報を収集するには、gather_table_stats、gather_schema_stats、gather_database_stats プロシージャで、dbms_statsパッケージを使用します。

gather_schema_statsは、情報が古いかどうかに関係なくすべてのオブジェクトの統計情報を収集するので、このプロシージャの使用をお勧めします。

例：

```
SQL> BEGIN  
      DBMS_STATS.GATHER_SCHEMA_STATS('SYS');  
      DBMS_STATS.GATHER_SCHEMA_STATS('SYSTEM');  
      END;  
  
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl \  
-l/tmp \  
-b gatherstats -- \  
--x"begin dbms_stats.gather_schema_stats('SYS'); dbms_stats.gather_schema_stats('SYSTEM');  
end;"
```

注：統計情報の収集には時間がかかり、多大なリソースを消費する可能性があるため、特に本番システムでは注意が必要です。カーソルの無効化が発生してハード解析につながるおそれがあります。また、新しい統計情報によりオプティマイザのプランが新しくなることがあります。該当する表の統計情報が古くなっていない場合、自動メンテナンス・タスクでは索引の統計情報が収集されない場合があります。

リソース使用率を適切に設定する

STREAMS_POOL_SIZE初期化パラメータは、64 MBから256 MBまでの範囲内で合理的な値に設定する必要があります。Oracle Data Pumpでは、プロセス間の通信にアドバンスド・キューイング（AQ）機能を使用します。SGA_TARGET初期化パラメータが設定されている場合、STREAMS_POOL_SIZE初期化パラメータは、データベース使用率に対して妥当な最小値に設定する必要があります。STREAMS_POOL_SIZEパラメータの設定に関する詳細については、Oracle Database Referenceを参照してください。

Release 19c以降では、DBAによるData Pumpリソース使用率の制限が可能

マルチテナント環境のプラガブル・データベースに、Data Pumpジョブの最大数および並列処理の最大数を設定します。

MAX_DATAPUMP_JOBS_PER_PDBパラメータは、作成されるジョブ数を制限します。このパラメータをSESSIONSの50%に自動設定するよう指定できます。この値は、各RACインスタンスの場合と同じです。値を設定した後で動的に変更することも可能で、PDBごとに変更できます。

MAX_DATAPUMP_PARALLEL_PER_JOBパラメータは、個々のData Pumpジョブでの並列処理数を制限します。このパラメータをSESSIONSの25%に自動設定するよう指定できます。この値は、各RACインスタンスの場合と異なる可能性があります。値を設定した後で動的に変更することも可能で、PDBごとに変更できます。

注：エラー"ORA-00018: maximum number of sessions exceeded"または"ORA-00020: maximum number of processes (%s) exceeded"が発生した場合は、それぞれ、許可しているジョブ数が多すぎるか、並列処理数が多すぎます。

異なるOSおよびストレージ上のサーバーにネットワーク・リンクを使用する

データベース・リンク経由で、ターゲット・データベースからインポート (impdp) を開始することができます。ダンプ・ファイルは生成されないため、移行中の一時記憶域の必要性を排除できます。Oracle Database Release 12.2以降では、dblink経由でのダイレクト・パス・ロードがサポートされています。これにはACCESS_METHOD=DIRECT_PATHパラメータを使用し、LONGおよびLONG RAWデータも対象に含まれます。

SecureFile LOBを使用する

SecureFile LOBの使用をお勧めします。これは特にパーティション化を使用する場合に当てはまります。SecureFile LOBは、BasicFile LOBよりもパフォーマンス、機能性、スケーラビリティに優れ、以下を提供します。

- LOB列を含む表に対するパラレルI/O
- 圧縮
- 暗号化

impdpでパラメータLOB_STORAGE=SECUREFILEを使用すると、古いLOBを自動的にSecureFile LOBに変換できます。ターゲット・データベース上に、SecureFile LOBを含む表のストレージが自動的に作成されます。

データベース互換性を設定する

データベース互換性レベルは、Data Pumpのエクスポートおよびインポート処理に影響を与えます。ソース・データベースの互換性レベルにより、エクスポート・ダンプ・ファイル・セットの互換性レベルが決まります。Oracle Database 21cより前のリリースでは、それぞれがソース・データベースとターゲット・データベースのバージョンに一致するexpdpおよびimpdpコマンドライン・クライアントを使用してエクスポートおよびインポートが実行されます。Oracle Database 21c以降では、汎用のexpdpおよびimpdpコマンドライン・クライアントが導入されており、Data Pumpをサポートするどのバージョンのデータベースでも使用できます。ネットワーク・モードのインポートは、ソース・データベースの互換性レベルがターゲット・データベースと一致するか、またはメジャー・バージョンが1つだけ違う場合に実行できます。

impdpコマンドでは、常により古いバージョンのData Pumpダンプ・ファイル・セットの読み取りが可能です。

注：ターゲット・データベースの互換性レベルがソース・データベースよりも低い場合、expdpのパラメータVERSIONを使用すると、ターゲット・バージョンを指定できます。

注：Oracle Database 18c以降では、使用するリリースに合わせてCOMPATIBLEを18.0.0または19.0.0に設定する必要があります。18.1.0、19.1.0、19.2.0などの2番目のバージョン番号は指定しません。

注：Data Pumpと従来のExportおよびImportユーティリティの間に相互運用性はありません。impdpを使用して、従来のexpダンプ・ファイルを読み取ることとはできず、impを使用してData Pumpのダンプ・ファイルを読み取ることもできません。

Data Pumpの互換性の詳細については、My Oracle Support Noteの『Export/Import Data Pump Parameter VERSION - Compatibility of Data Pump Between Different Oracle Versions』 ([Doc ID 553337.1](#)) を参照してください。

スペース使用とパフォーマンスのためにエクスポートで圧縮を使用する

エクスポート中にメタデータやデータを圧縮すると、ダンプ・ファイルのサイズとネットワーク・モードのインポートでのデータ・ストリーム・サイズを削減できます。また、一部のケース（特にネットワーク・モード）でパフォーマンスが向上する場合があります。ただし、生データの変換に追加のCPUリソースが必要になるので、テストを実行することをお勧めします。

圧縮の対象として、メタデータ、データ、その両方、どちらでもないを指定できます。デフォルトは、COMPRESSION=METADATA_ONLYです。

ほとんどのケースで推奨される圧縮アルゴリズムは、COMPRESSION=ALLおよびCOMPRESSION_ALGORITHM=MEDIUMです。このアルゴリズムは高速で、デフォルトのBASICと比べて大きなオーバーヘッドの違いもありません。

圧縮アルゴリズム	特性
BASIC	パフォーマンスに影響しない適切な圧縮
LOW	圧縮率よりもCPU使用率の方が重要である場合に使用

MEDIUM	推奨。BASICと似ているが、アルゴリズムは異なる
HIGH	最大の圧縮率とCPU使用率

注：Data Pumpでデータ圧縮を使用するには、Advanced Compressionオプションのライセンスが必要です。
COMPRESSION=METADATA_ONLYの指定と圧縮ダンプ・ファイルのインポートにはライセンスは不要です。

注：Oracle Database 10g Release 1（10.1）では圧縮はサポートされていませんでした。

実際の例 – 12.2 EBS データベースのエクスポート

	ファイル・サイズ (MB)	圧縮率	時間
なし	5.500	1,0	4m 54s
すべて BASIC	622	8,9	4m 58s
すべて LOW	702	7,8	5m 24s
すべて MEDIUM	567	9,7	4m 55s
すべて HIGH	417	13,2	5m 13s

	ファイル・サイズ (MB)	圧縮率	時間
なし	5.800	1,0	2m 33s
すべて BASIC	705	8,2	3m 03s
すべて LOW	870	6,6	8m 11s
すべて MEDIUM	701	8,2	3m 01s
すべて HIGH	509	11,3	12m 16s

Data Pumpジョブ実行前のデータベース・チェック

AQ_TM_PROCESSESデータベース・パラメータ

AQ_TM_PROCESSESをゼロに設定しないでください。ゼロを指定するとアドバンスド・キューの処理速度が低下するおそれがあり、その結果としてアドバンスド・キューイングを使用するData Pumpの処理が遅くなる場合があります。このパラメータはNullのままにするか、0より大きい値を設定してください。

_OPTIMIZER_GATHER_STATS_ON_LOAD隠しパラメータ

Oracle Database 12c以降では、オンライン統計収集により、Create Table as Select（CTAS）またはダイレクト・パス挿入を使用したデータ・ロード処理時に統計情報が自動生成されます。ただし、デフォルト設定である_OPTIMIZER_GATHER_STATS_ON_LOAD=TRUEは、インポート処理のパフォーマンスを低下させる可能性があります。

Data Pumpの処理が完了するか、DBMS_STATS.GATHER_DATABASE_STATSを使用してデータベース統計情報を手動で収集したら、その後で忘れずにパラメータをTRUEにリセットしてください。

Oracle RAC用の_lm_share_lock_opt隠しパラメータ

Oracle Real Application Clusters（Oracle RAC）Release 12.2以降では、Oracle RACのライブラリ・キャッシュでSHAREロック（S-lock）最適化を使用できます。

Release 12.2のOracle RACで、parallelに1より大きい数を指定して開始したimpdp処理中に、'ライブラリ・キャッシュ・ロック'（サイクル）の問題が発生するのを回避するには、すべてのRACインスタンスでインポート時に"_lm_share_lock_opt"=FALSEを設定するか、またはparallel=1を指定してメタデータ・インポートを実行することを検討してください。

詳しくは、My Oracle Support Noteの『'Library Cache Lock' (Cycle) Seen During DataPump Import in 12.2 RAC Environment』（Doc ID 2407491.1）を参照してください。

Oracle Cloudデータベース向けのその他のプラクティス

Oracle Cloudへの移行時のスキーマ・レベルのエクスポート

OracleデータベースからOracle Cloudへ移行する際、スキーマ・レベルのエクスポートを実行します。たとえば、パラメータ`schemas=schema_name,...`を使用します。

```
$ EXPDP SYSTEM DIRECTORY=DP_DIR SCHEMAS=SCOTT LOGFILE=EXPORT_SCOTT.LOG PARALLEL=8 ...
```

これにより、許可されるユーザー・スキーマだけがOracle Cloudデータベースに移行されます。

自律型データベース移行時のネットワーク・リンクの使用

ターゲットの自律型データベースからデータベース・リンク経由で、インポート（`impdp`）を開始できます。この場合、ダンプ・ファイルは生成されません。

アクセス方法

Data Pumpのエクスポート・パラメータ`ACCESS_METHOD`を使用すると、エクスポート時に特定のデータ・アンロード方法を使用するように指定できます。デフォルトでは、`AUTOMATIC`に設定されています。Release 12.2以降では、`ACCESS_METHOD=DIRECT_PATH`と`NETWORK_LINK=<dblink>`を使用すると、ソース・データベース内の`LONG`または`LONG RAW`タイプを`dblink`経由でエクスポートできます。

チェックサムによるデータファイルの検証

Data Pumpは、アプリケーション・データをオンプレミスからOracle Cloudに移行するために使用されます。Oracle Database 21cには、ダンプ・ファイルにチェックサムを追加するチェックサム・パラメータが含まれています。これにより、オブジェクト・ストアに転送後またはオブジェクト・ストアから転送後に、ファイルが有効で、悪意のある変更が加えられていないことを検証できます。チェックサムは、暗号化ハッシュを使用して、ヘッダー・ブロック以降のダンプ・ファイルの内容の整合性と、意図しないエラーがダンプ・ファイルに含まれていないことを確認します。

値を設定することで、Oracle Data Pumpのエクスポート・ダンプ・ファイル・セットでチェックサムを計算するかどうかと、チェックサムの計算に使用するハッシュ・アルゴリズムを指定できます。

パーティション化されていない表へのインポート

ソース・データベースにパーティション表が含まれており、パーティション化を使用しないAutonomous Data Warehouseデータベースにデータを移行する場合は、`DATA_OPTIONS=GROUP_PARTITION_TABLE_DATA`を使用します。これにより、Data Pumpでパラレル問合せエンジンが使用され、より効率的にデータウェアハウスにデータをロードできます。

AL32UTF8データベース・キャラクタ・セットの使用

オラクルは、データベースのキャラクタ・セットとしてAL32UTF8を使用することを推奨しています。これは他のすべてのキャラクタ・セットのスーパーセットであるため、もっとも一般的なキャラクタ・セットです。

結論

Oracle Data Pumpは、Oracle Databaseの移行のための、フル機能の成熟した柔軟なツールです。この技術概要で述べたように、新しいリリースごとにパフォーマンスとリソース使用を最適化するためのオプションが追加されています。本書に記載したベスト・プラクティスに従うことで、Data Pumpを使用したエクスポートおよびインポートの実行が可能な限り円滑かつ迅速になるでしょう。

CONNECT WITH US

+1.800.ORACLE1までご連絡いただくか、oracle.comをご覧ください。

北米以外の地域では、oracle.com/contactで最寄りの営業所をご確認いただけます。



blogs.oracle.com



facebook.com/oracle



twitter.com/oracle

Copyright © 2021, 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。0120

Oracle Data Pumpのベスト・プラクティス

2021年8月

著者：William Beauregard

共著者：Roy Swonger

