

Oracle Data Pump

ベスト・プラクティス

2024年5月、バージョン2.0 Copyright © 2024, Oracle and/or its affiliates 公開

目次

サポートとData Pump Bundleパッチの取得	4
SYS as SYSDBAを使用してエクスポートを起動しない	4
パラメータ・ファイルを使用する	4
Data Pumpエクスポートの一貫性を確保する	5
エクスポートとインポートから統計情報を除外する	6
診断を使用する	6
常にログ・ファイル診断パラメータを使用する	6
アラート・ログを確認する	6
データベース・ビューでプロセスを監視する	6
Data Pumpトレースを使用する	7
SQLトレースを使用する	7
並列処理と統計情報を使用してパフォーマンスを向上させる	7
並列処理を使用する	7
ワイルドカードを指定して複数のダンプ・ファイルを作成する	9
Data Pump処理前後で統計情報を収集する	10
リソース使用率を適切に設定する	10
ネットワーク・モードのインポート	11
SecureFile LOBを使用する	11
データベース互換性を設定する	11
ユニバーサル・クライアントを使用する	11
スペース効率とパフォーマンスのために圧縮を使用する	12
Data Pumpジョブの前にデータベース・パラメータを確認する	13
IncludeおよびExcludeパラメータを使用する	13
スケジューラ・ジョブをエクスポートから除外する、またはエクスポートに含める	14
大規模な索引を除外し、インポート後に並列処理によって構築する	14
Oracle Autonomous Database向けのその他のプラクティス	14
スキーマ・レベルまたは表レベルのエクスポートを実行する	14
自律型データベースへのネットワーク・リンク・モードのインポートを使用する	15
アクセス方法を考慮する	15
DATAPUMP_CLOUD_IMPロールを使用する	15
チェックサムでダンプ・ファイルを検証する	15
パーティション化されていない表へインポートする	16
AL32UTF8データベース・キャラクタ・セットを使用する	16
ADBダンプ・ファイル・セットの作成、送信、ダウンロード	16
まとめ	16



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。一般的な方法は、リリース10g以降のOracle Databaseの機能で、

リリース9i以前のOracleエクスポート/インポート(exp/imp)ユーティリティの後継となるOracle Data Pumpを使用することです。 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、オンプレミスのOracleデータベースや他のOracle CloudデータベースをOracle Cloudに移行する場合、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エクスポートおよびインポート・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、BasicFiles LOBからSecureFiles 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
- impdbを使用して、クラウドPDBにインポート
- または、データベース・リンク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。 以下のパラグラフでは、Oracle Data Pumpを使用して移行を実行するためのベスト・プラクティスについて説明します。

注:"移行"という用語は、Oracle以外のデータベースからOracleデータベースへのデータの移動について説明する場合にも使用されます。この技術概要では、ソースとターゲットの両方がOracleデータベースである場合の移行についてのみ説明します。

サポートとData Pump Bundleパッチの取得

オラクルでは、重要な修正を反映するために最新のData Pump Bundleパッチを使用することを推奨します。このパッチは、データベース・リリース更新には含まれません。これについては、将来のリリースに向けて検討中です。詳しくは、MOS Note『Data Pump Recommended Proactive Patches For 19.10 and Above - 2819284.1』を参照してください。

サポートのリクエストに含めるべき役立つ情報には、以下のようなものがあります。

- 1. データベース・バージョン、リリース更新、Data Pumpクライアント・バージョンなど、ソース/ターゲット環境に関する詳細
- 2. データベースとジョブに関する情報
 - STREAMS_POOL_SIZE設定
 - 量に関連する統計情報が最近実行されたかどうか
 - 発生したエラー・メッセージ(ログ・ファイルにない場合)
 - エラーが表に関連する場合の表定義
 - LOBストレージがSECUREFILEかBASICFILEか
- 3. 関連するファイル
 - ジョブのエクスポートまたはインポートの起動に使用されるparfile.parまたはコマンド
 - 診断パラメータMETRICS=Y LOGTIME=ALLを使用して作成されたlogfile.log
 - 初期化パラメータ・ファイル内のパラメータDIAGNOSTIC_DESTで指定されたADRディレクトリにあるalert_SID.log
 - 代表的な期間(約15分間)のAWRレポート

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。AS SYSDBAには、パラレル DMLの使用にいくつかの制限事項があります。さらに、状況によっては、AS SYSDBAはいくつかの内部の一貫性チェックを回避できる場合があります。これは望ましくありません。

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、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を回避できます。

Autonomous Databaseへの移行に使用するmy_data_pump_parfile.parという名前のparfileの例を次に示します。

PARALLEL=16
DUMPFiLE=adbdump%L.dmp
FILESIZE=10GB METRICS=Y
LOGTIME=ALL
JOB_NAME=ADBMIGRATION
DIRECTORY=DP_DIR
EXCLUDE=STATISTICS
COMPRESSION=ALL
COMPRESSION_ALGORITHM=MEDIUM
FLASHBACK_SCN=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

注:

リリース12.2より前では、DUMPFILE=adbdump_1<mark>%U</mark>.dmp, adbdump_2<mark>%U</mark>.dmp…を使用することで、接頭辞ごとに99のダンプ・ファイルが可能になります。

リリース12.1以降では、LOGTIME=ALLを使用します。

Autonomous Databaseでは、ダンプ・ファイルの最大サイズは10 GBです。より小さなサイズも指定できます。

メンテナンス中にシステムが静的で、エクスポート中にジョブが確実に起動しないようにしたい場合は、flashback_scnの代わりに job_queue_process=0を設定します。UNDO保存期間の延長が必要な場合があります。デフォルトでは、900秒に設定されています。エクスポート・ジョブ中のデータベースへの接続に不安がある場合は、制限されたセッションを使用します。

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=<tirmestamp>を使用すると、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。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。最終的に、
FLASHBACK_TIME=SYSTIMESTAMPに変換されるリリース11.2のレガシー・インタフェースCONSISTENT=Yをまだ使用できます。

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更して、エクスポートを実行することもできます。ただし、Data Pumpでは制御表を作成してエクスポートを調整する必要があります。これには読取り/書込みインスタンスが必要です。

これは、DMLのみをサポートしているDMLリダイレクト操作とActive Data Guardでは実行できません。DDLは、CREATE TABLE コマンドではサポートされません。

GoldenGateのインスタンス化フィルタリングも役立ちます。エクスポート時にFLASHBACK_<SCN|TIME>は必要なくなりました。これは、GoldenGateがおのおのの表のSCNをエクスポートされた時点で認識し、表ごとにそのSCNの時点の変更のレプリケートを開始することを意味します。これで問題がすべて解決されるわけではありません。なぜなら、表AがSCN Xとしてエクスポートされ、表BがXより後のSCN Yの時点でエクスポートされる場合があるためです。これらの表の間に制約のような依存関係がある場合や、表Aのまだレプリケートされていない行に依存する追加の行が表Bにある場合は、これらの制約の検証に失敗する場合があります。

エクスポートとインポートから統計情報を除外する

オラクルでは、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、トランスポータブル表領域に

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。

診断を使用する

常にログ・ファイル診断パラメータを使用する

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ことが役立ちます。

METRICS=YESパラメータを使用して、移動されたオブジェクト、経過時間、およびワーカーの割当てに関する詳細をOracle Data Pumpログ・ファイルに記録します。

例:

```
Processing object type SCHEMA_EXPORT/TABLE/TABLE
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
. imported "METALI"."ALBUMS"
. imported "METALI"."BANDS"
. imported "METALI"."BANDS"
. imported "METALI"."REVIEWS"

All diagnostics

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 17:26:59.082: 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 17:26:59.082: Completed by worker 3 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 17:26:59.082: 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 "METALI"."ALBUMS" 988.8 KB 28069 rows in 2 seconds using external table
16-OCT-20 17:27:03.778: . imported "METALI"."BANDS" 3.444 MB 37723 rows in 2 seconds using external table
16-OCT-20 17:27:12.644: . imported "METALI"."REVIEWS" 66.47 MB 21510 rows in 13 seconds using external table
```

アラート・ログを確認する

Data Pumpジョブ中に、アクティビティのalert.logを確認します。

データベース・ビューでプロセスを監視する

Data Pumpプロセスが遅かったり応答しなかったりする場合は特に、以下のビューを使用してData Pumpプロセスを監視します。

- DBA_DATAPUMP_JOBS。 MOS Note: 1471766.1 『How To Monitor The Progress Of Data Pump Jobs』を参照してください。
- DBA_DATAPUMP_SESSIONS。 MOS Note: 1528301.1 『Finding Out The Current SQL Statement A Data Pump Process Is Executing』を参照してください。

• V\$SESSION_LONGOPS。 MOS Note: 455720.1 - 『How can we monitor a DataPump Job's Progress?』を参照してください。

select sid, serial#, sofar, totalwork from V\$SESSION_LONGOPS where opname = '<your export job name>' and sofar != totalwork;

Data Pumpトレースを使用する

MOS Note: 286496.1 - 『DataPump Parameter TRACE - How to Diagnose Oracle Data Pump』を参照してください。

トレースには、DBAロールまたはEXP_FULL_DATABASE / IMP_FULL_DATABASEロールを持つ特権ユーザーが必要です。 max_dump_file_sizeがトレースを取得するのに十分な大きさであるようにしてください(デフォルトは無制限)。 Data Pumpトレースを開始するには、3つの方法のうちの1つを使用します。 MOS Note: 286496.1を参照してください。

- TRACEパラメータを推奨されるビットマップに設定する: trace=1FF0300
- TRACEをインタラクティブ・モードで設定する: impdp user/password attach=user.imp_job_1 trace=1FF0300
- TRACEイベントをSPFILE/PFILEで設定、またはALTER SYSTEM経由で設定する

SOLトレースを使用する

SQL_TRACE information to diagnoseクエリー・パフォーマンス問題の10046トレースを収集します。 MOS Note: 376442.1 - 『How To Collect 10046 Trace (SQL_TRACE) Diagnostics for Performance Issues』を参照してください。

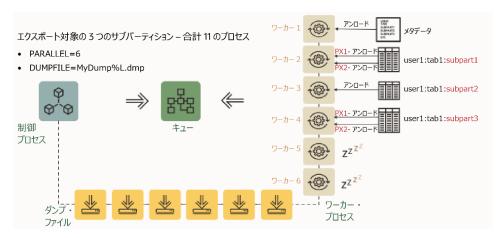
並列処理と統計情報を使用してパフォーマンスを向上させる

並列処理を使用する

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 PARALLELパラメータは、エクスポートまたはインポートに追加のワーカー・プロセスを作成します。

PARALLEL=nパラメータは、エクスポートまたはインポート・ジョブにアクティブになり得るワーカー・プロセスおよびパラレル問合せ(PX)プロセスの最大数を指定します。この数に制御プロセスは含まれていません。通常、nの値はCPUコアの数の2倍にしますが、必要に応じて調整します。

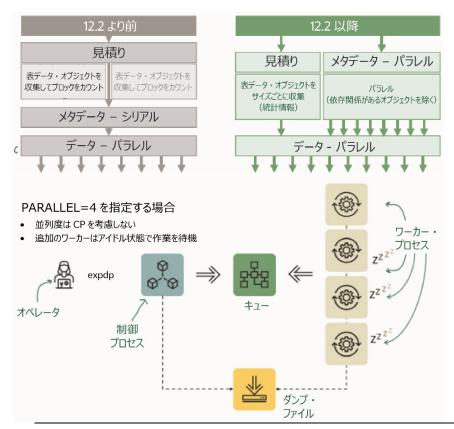
Data Pumpは2種類の並列処理を使用します。



制御プロセスによって1つのワーカーが小さな表、またはパーティションもしくはサブパーティションに割り当てられます。これは、表間の並列処理と呼ばれます。ワーカーはまた、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使用される1つまたは複数のパラレル実行(PX、旧称PQスレーブ)プロセスを調整できます(表内の並列処理)。PXプロセスを調整しているワーカーは、

ジョブの最大並列度に加算されません。

リリース12.2以降、メタデータのエクスポートは表データの見積りフェーズと同時に発生します。

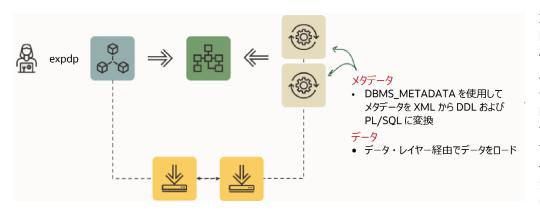


ほとんどのメタデータおよびデータ・オブジェクトは、PARALLEL=2以上の場合にパラレルでエクスポートされます。Oracle Data Pumpは、いくつかのリリースではパッケージ本体をパラレルでインポートしました。リリース12.2より、ダンプ・ファイルによるエクスポートおよびインポートでは、Oracle Data Pumpは、ほとんどのメタデータとほとんどのデータベース・オブジェクトをパラレルでインポートします。タイプ(継承のため)、スキーマ、プロシージャのアクションなど、依存関係のあるデータベース・メタデータ・オブジェクトは引き続き順次インポートされます。また、PARALLELパラメータでは、パラレルで作成される索引の数も決定され、各索引には1つの並列度が含まれます。

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 用の修正を適用すると、制約と索引のパラレル・インポートを有効にできます。

注:トランスポータブル表領域のメタデータのパラレル・エクスポートおよびインポートは、Oracle Database 21cより使用できます。

インポート中、PARALLELはエクスポートで使用した設定とは異なる値に設定できます。



重要なワーカー・プロセスでは、DBMS_METADATA APIを使用して、XMLドキュメントにエクスポートされたメタデータをインポート用にDDLおよびPL/SQLに変換します。1つのXMLドキュメントは一度に1つのワーカーに割り当てられます。データのロード(インポート)またはデータの移動(ネットワーク・モードのインポート)には

データ・レイヤーが使用されます。ネットワーク・モードのインポートでは、DBMS_METADATA APIを使用してソース・データベース・メタデータを実行用に変換し、データをターゲットへ移動します。

Data Pumpインポートでは、ロードされている表、パーティション、またはサブパーティションへのロックを取得する必要があります。



Data Pumpのdata_options=TRUST_EXISTING_TABLE_PARTITIONSパラメータは、既存のターゲット表のパーティション・スキームがソースからインポートされている表のパーティション・スキームと同一であることを確認します。そのため完全並列処理が使用され、それ以外の場合は1つのワーカーが表にアクセスします。

ワイルドカードを指定して複数のダンプ・ファイルを作成する

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%Lまたは%U(レガシー)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。

リリース12.2以降は、%L wildcardを使用すると、%Uで作成される99を超えるファイルを作成できます。ファイル名の数字は3~10桁の可変幅の整数です。

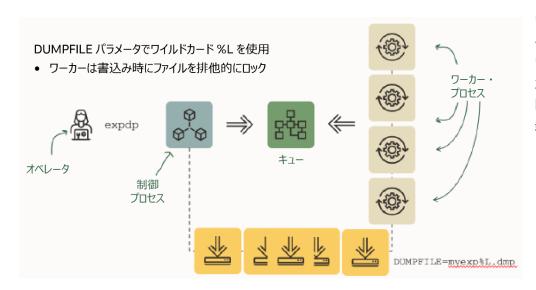
dumpfile=dumpfile%L.dmp filesize=n

レガシーの%Uは、12.2より前のリリースで使用可能な唯一のワイルドカードです。これで作成できるのは、1から99のファイルです。ファイル名の数字は固定幅の2桁で、増分する整数です。

dumpfile=dumpfile%U.dmp filesize=n

リリース12.2より前の移行のために99を超えるダンプ・ファイルが必要な場合は、dump_01_%U.dmp、dump_02_%U.dmpなど、ダンプ・ファイルのファイル名の接頭辞を複数作成することで%Uワイルドカードの使用を継続できます。

ダンプ・ファイル名でワイルドカードを指定しない場合は、ダンプ・ファイルによってパラレル・ワーカーが制限される可能性があります。パラレル・ワーカーは、ログ・ファイルをロックして順番に書込みを行います。



ワイルドカードを指定しない場合、 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 影響が及ぶ場合があります。

Data Pump処理前後で統計情報を収集する

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と、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、Data Pumpがメタデータ・オブジェクトをフィルタ、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、パラレル・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または 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、オラクルでは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ので、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

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ト・キューイング(AQ)機能を使用します。SGA_TARGET初期化パラメータを設定

した場合、次にSTREAMS_POOL_SIZE初期化パラメータをデータベース使用量の適切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。

Data Pumpジョブのリソース使用量を制限する

多数のプラガブル・データベース(PDB)を含むマルチテナント・コンテナ・データベース(CDB)は、1つのPDBのData Pump処理が他のPDBに影響するのを回避しようとします。

リリース19c以降では、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できます。

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。デフォルトでは、このパラメータは SESSIONSの50パーセントに設定されています。この値は、各RACインスタンスの場合と同じです。PDBごとに動的に設定および 変更ができます。

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。デフォルトでは、このパラメータはSESSIONSの25パーセントに設定されています。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。PDBごとに動的に設定および変更ができます。

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、許可しているジョブまたは並列処理がそれぞれ多すぎる可能性があります。

ネットワーク・モードのインポート

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リリース12.2より、パラメータ

ACCESS_METHOD=DIRECT_PATHを使用して、LONGデータやLONG RAWデータなど、Direct Path Load向けのサポートがデータベース・リンク経由で提供されています。

SecureFile LOBを使用する

オラクルでは、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 (BasicFile LOBでの並列処理なし)
- 圧縮
- 暗号化

impdpパラメータLOB_STORAGE=SECUREFILEを使用して自動的にBasicFile LOBをSecureFiles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。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。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。

Oracle Database 18c以降、COMPATIBLEは、リリース更新番号(例:19.3.0)ではなく、メジャー・リリースの3桁を使用してのみ設定する必要があります(例:18.0.0)。

ユニバーサル・クライアントを使用する

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クライアント・バージョン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。ただし、impdpは常に前のバージョンのダンプ・ファイルを読み取ることができます。

Oracle Database 21c以降では、Data Pumpには汎用のexpdpおよびimpdp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。

注: Data Pumpと従来のエクスポートおよびインポート・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『Export/Import DataPump Parameter VERSION - Compatibility of Data Pump Between Different Oracle Versions』(Doc ID 553337.1) を参照してください。

スペース効率とパフォーマンスのために圧縮を使用する

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数、およびネットワーク・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・モードのインポート)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。MEDIUM圧縮アルゴリズムはサイズとパフォーマンスのバランスが最適で、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。

圧縮はOracle Database 10g Release 1 (10.1) で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	4m54s
すべてBASIC	622	8,9	4m58s
すべてLOW	702	7,8	5m24s
すべてMEDIUM	567	9,7	4m55s
すべてHIGH	417	13,2	5m13s

	ファイル・サイズ(MB)	圧縮率	時間
なし	5.800	1,0	2m33s
すべてBASIC	705	8,2	3m03s
すべてLOW	870	6,6	8m11s
すべてMEDIUM	701	8,2	3m01s
すべてHIGH	509	11,3	12m16s

Data Pumpジョブの前にデータベース・パラメータを確認する

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、アドバンスト・キューの処理速度と、アドバンスト・キューイングを使用するData Pumpの処理速度が低下する場合があります。このパラメータはNullのままにするか、O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APPENDヒントを使用した 挿入などのダイレクト・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。 Data Pumpジョブに推奨される 値は、_OPTIMIZER_GATHER_STATS_ON_LOAD=falseです。 ただし、デフォルトの設定は

_OPTIMIZER_GATHER_STATS_ON_LOAD=TRUEです。この設定は、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=パラメータに1より大きい値を設定したimpdp処理中に、'Library Cache Lock'(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、『MyOracle Support note: '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。

IncludeおよびExcludeパラメータを使用する

メタデータのカテゴリは、オブジェクト・パスで表示されます。以下に例を示します。

TABLE
TABLE/INDEX
TABLE/STATISTICS/TABLE_STATISTICS
TABLE/TRIGGER

オブジェクト・パスの完全なリストは、エクスポートのモードに対応する以下のビューから入手できます。

DATABASE_EXPORT_OBJECTS
SCHEMA_EXPORT_OBJECTS
TABLE_EXPORT_OBJECTS

一部のメタデータには依存関係があります。たとえば、表を除外すると、関連する索引、制約、付与、トリガー、および他の表関連の オブジェクトも除外されます。索引を除外すると、その索引上の統計も除外されます。

特定のカテゴリのメタデータを追加または削除するには、INCLUDEパラメータまたはEXCLUDEパラメータを使用します。Oracle Database 21c以降では、これらのパラメータを組み合わせることができます。最初にINCLUDEパラメータが処理され、パラメータに識別されるすべてのオブジェクトが含められます。次に、EXCLUDEパラメータが処理されます。このパラメータにより、含められたオブジェクトのリストにあるすべてのオブジェクトが削除されます。

次の例では、tableオブジェクト・パスが含められ、そのオブジェクト・パスからtable statisticsが除外されます。

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=exp%u.dmp SCHEMAS=hr,oe include=table exclude=statistics

スケジューラ・ジョブをエクスポートから除外する、またはエクスポートに含める

SCHEDULER JOBSメタデータは、SYSスキーマに保存されています。このメタデータを除外するには、MOS Note『How To Exclude Scheduler Job From Data Pump Export (expdp)』 (Doc ID 1414478.1) を参照してください。このメタデータを全データベースのエクスポートに含めるには、反対のアプローチを使用します。

大規模な索引を除外し、インポート後に並列処理によって構築する

各索引は単一のData Pumpワーカーによって作成されます。これにより、小規模の索引の高速処理が可能になりますが、大規模な索引の処理時にインポート・ジョブの長さが増加する場合があります。これを改善するには、以下の手順を使用してインポート・プロセスの後に大規模な索引を作成します。

4. 対象の大規模な索引を見つけます

SQL> select segment_name, round(bytes/1024/1024) as GB

from user_segments

where segment_type='INDEX'

order by GB desc;

5. 大規模な索引をインポートから除外します

\$ cat import.par

... exclude=INDEX:"='BIG1','BIG2','BIG3'"

...

impdp ... parfile=import.par

6. SOLFILEパラメータを使用して大規模な索引のメタデータを生成します

\$ cat import-sqlfile.par

... include=INDEX:"='BIG1','BIG2','BIG3'"

sqlfile=index.sql

•••

impdp ... parfile=import-sqlfile.par

7. インポート・ジョブが完了したら、並列度を変更して索引を作成します

SQL> CREATE INDEX BIG1PARALLEL n; SQL> ALTER INDEX INDEX BIG1PARALLEL 1;

注:sqlfileパラメータを用いてimpdpを使用する代わりに、DBMS_METADATAを使用できます。

DBMS_METADATA.GET_DLL関数は、索引または他のオブジェクトのメタデータを返します。この関数は、エクスポート操作の前に ソース・データベース上で実行できます。

Oracle Autonomous Database向けのその他のプラクティス

ドキュメント<u>Import Data Using Oracle Data Pump on Autonomous Database</u>のAutonomous Databaseサービスのベスト・プラクティスを参照してください。

スキーマ・レベルまたは表レベルのエクスポートを実行する

Oracle DatabaseをOracle Cloudへ移行する際は、SCHEMAまたはTABLESモードのエクスポートを実行します。これは、FULLモードのエクスポートでは、ADMINユーザーがアクセスを許可されていないオブジェクトを取得しようとするためです。

エクスポート。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。スキーマ・モードでは、1つのスキーマから別の

スキーマへの参照、パブリック・シノニム、権限などを手動で処理することに注意してください。これはスクリプトを作成して実行できます。通常は、ADBからのエクスポートではSCHEMAモードまたはTABLEモードを使用する必要があります。これは、FULLモードでは、ADMINユーザーが取得を許可されないオブジェクトを取得しようとするためです。

以下に示す一部のオブジェクトは、全体エクスポートでのみエクスポートされます。

- 監査証跡とポリシー
- Oracle Database Vault
- ディレクトリ
- プロファイルおよびパスワード確認機能、パブリック・シノニム
- ロール
- パブリック・データベース・リンク
- SQL管理オブジェクト(計画履歴、ベースライン、SQLプロファイルなど)
- 表領域
- ユーザー(SCHEMASパラメータで指定されたもの以外)
- Workspace Manager (DBMS_WM.Export_Schemasを使用)

注:Data PumpはSYSオブジェクトまたはAWRの権限付与はエクスポートしません。

自律型データベースへのネットワーク・リンク・モードのインポートを使用する

Oracle Database 19c以降は、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。

アクセス方法を考慮する

Data Pump ExportのACCESS_METHODパラメータは、データをアンロードするための特定の方法を使用するようExportに指示します。デフォルトでは、AUTOMATICに設定されます。リリース12.2より、ソース・データベースのLONGタイプまたはLONG RAW タイプは、ACCESS_METHOD=DIRECT_PATHとNETWORK_LINK=<dblink>を使用してデータベース・リンク経由でエクスポートできます。

DATAPUMP_CLOUD_IMPロールを使用する

DATAPUMP_CLOUD_IMPロールは、完全インポートの実行、または他のユーザーが所有するオブジェクトのインポートに必要です。

チェックサムでダンプ・ファイルを検証する

Oracle Database 21c以降、Data Pumpにはチェックサムをダンプ・ファイルに追加するチェックサム・パラメータが含まれます。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。これにより、ネットワーク経由でオブジェクト・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。

Data Pumpのチェックサムは、通常のmd5sumチェックサムよりも優れています。アルゴリズムのセキュリティが強化されています。 ハッシュが暗号化された形式でダンプ・ファイル・ヘッダーに書き込まれているため、ハッシュ値を用いた改ざんが困難になっています。

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。

インポートの前に、実際のインポートを開始する前に検証を行うことを選択できます。すでにダンプ・ファイルを検証済みの場合は、インポート中の検証をスキップできます。

パーティション化されていない表へインポートする

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パラメータを使用します。これにより、Data Pumpはデータのロード時にパラレル問合せエンジンをより効率的に使用できます。

AL32UTF8データベース・キャラクタ・セットを使用する

オラクルでは、Oracle Databaseキャラクタ・セットにAL32UTF8を使用することを推奨します。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す。

ADBダンプ・ファイル・セットの作成、送信、ダウンロード

FILESIZE=10Gパラメータと10 G以下のファイル・サイズを使用します。これが、Autonomous Databaseの現在の上限です。
Oracle Data Pumpを使用してオブジェクト・ストアに直接エクスポートする場合、オブジェクト・ストア・バケット内のダンプ・ファイルのサイズはゼロと表示されます。ADBでは、各ダンプ・ファイルの部分を小さなチャンクに分割してアップロードを高速化します。たとえば、次のとおりです。

exp01.dmp exp01.dmp_aaaaaa exp02.dmp exp02.dmp_aaaaaa オブジェクト・ストアから完全なダンプ・ファイルをダウンロードするには、CurlなどのSwiftをサポートするツールを使用して、ユーザー・ログインとSwift認証トークンを指定します。たとえば、次のとおりです。

curl -O -v -X GET -u 'user1@example.com:auth_token' \ https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/namespace-string/bucketname/exp01.dmp

curlコマンドは、そのURLでワイルドカードや置換文字をサポートしません。置換文字をサポートするスクリプトを使用して、1つのコマンドですべてのダンプ・ファイルをオブジェクト・ストアからダウンロードできます。スクリプトの例は、このブログの記事を参照してください。

まとめ

Oracle Data Pumpは、Oracle Databaseの論理的移行のための、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 © 2024, Oracle and/or its affiliates.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、Java、MySQLおよびNetSuite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。