

Oracle® Database

同じシステムでの PDB の並列アップグレード 19c

F16177-02(原本部品番号:F10902-02)

2019年5月

タイトルおよび著作権情報

Oracle Database同じシステムでのPDBの並列アップグレード, 19c

F16177-02

Copyright © 2018, 2019, Oracle and/or its affiliates. All rights reserved.

原著者: Sunil Surabhi, Nirmal Kumar

原協力者: Lance Ashdown, Padmaja Potineni, Rajesh Bhatiya, Prakash Jashnani, Douglas Williams, Mark Bauer

原協力者: Roy Swonger, Byron Motta, Hector Vieyra Farfan, Carol Tagliaferri, Mike Dietrich, Marcus Doeringer, Umesh Aswathnarayana Rao, Rae Burns, Subrahmanyam Kodavaluru, Cindy Lim, Amar Mbaye, Akash Pathak, Thomas Zhang, Zhihai Zhang

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアもしくはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアもしくはハードウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。お客様との間に適切な契約が定められている場合を除いて、オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。お客様との間に適切な契約が定められている場合を除いて、オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

- [タイトルおよび著作権情報](#)
- [はじめに](#)
 - [このドキュメントのユースケース・シナリオ](#)
 - [ドキュメントのアクセシビリティについて](#)
- [1 マルチテナント・アーキテクチャの平行・アップグレード](#)
 - [プラグブル・データベース\(PDB\)の平行・アップグレードの概要](#)
 - [マルチテナント・コンテナ・データベースの平行・アップグレード](#)

はじめに

このガイドは、特定のユースケース・シナリオを完了する助けとなるよう収集されたOracle Databaseユーザー支援ドキュメントのトピック集です。

- [このドキュメントのユースケース・シナリオ](#)
- [ドキュメントのアクセシビリティについて](#)

このドキュメントのユースケース・シナリオ

このシナリオ・ドキュメントを使用すると、マルチテナント・アーキテクチャ全体(CDB\$ROOT、PDB\$SEEDおよびCDBに接続されたすべてのプラガブル・データベース(PDB)を含む)を新しいリリースにアップグレードし、パラレル処理を使用して停止時間を制限することが容易になります。

このシナリオの前提条件

- 新しいリリースのOracle Databaseソフトウェアをサーバーにインストールしておきます。
- マルチテナント・アーキテクチャをバックアップしておきます。

CDBのアップグレードを開始する前に、ソース・データベースとターゲット・データベースを最新のリリース更新(Update)またはリリース更新リビジョン(Revision)に更新することをお勧めします。

このシナリオの概要

- **マルチテナント・アーキテクチャのパラレル・アップグレード。** CDBをアップグレードし、CDBのPDBの全部または一部をパラレルにアップグレードする手順を理解してから、適用します。
 - プラガブル・データベースのパラレル・アップグレードの概要を確認します。
 - マルチテナントCDBおよびPDBをパラレルにアップグレードします。

前述のステップはそれぞれ、このドキュメントの各章および各項に対応しています。

親トピック: [はじめに](#)

ドキュメントのアクセシビリティについて

Oracleのアクセシビリティについての詳細情報は、Oracle Accessibility ProgramのWebサイト (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>)を参照してください。

Oracleサポートへのアクセス

サポートを購入したオラクル社のお客様は、My Oracle Supportを介して電子的なサポートにアクセスできます。詳細情報は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>)か、聴覚に障害のあるお客様は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>)を参照してください。

親トピック: [はじめに](#)

1 マルチテナント・アーキテクチャの平行・アップグレード

この方法を使用して、コンテナ・データベース(CDB)をアップグレードしてから複数のプラグブル・データベース(PDB)を平行にアップグレードして、マルチテナント・アーキテクチャのOracle Databaseリリース(Oracle Databaseリリース12c (12.1.0.1)以降)をアップグレードします。

- [プラグブル・データベース\(PDB\)の平行・アップグレードの概要](#)

平行の方法を使用して、CDBをアップグレードしてからすぐに平行SQLプロセッサを使用してPDBをアップグレードできます。

- [マルチテナント・コンテナ・データベースの平行・アップグレード](#)

この方法を使用して、1回のアップグレード操作でCDB\$ROOT、PDB\$SEED、およびCDB内のすべてのPDBをアップグレードします。

プラグブル・データベース(PDB)の平行・アップグレードの概要

平行の方法を使用して、CDBをアップグレードしてからすぐに平行SQLプロセッサを使用してPDBをアップグレードできます。

コンテナ・データベース(CDB)には、0 (ゼロ)、1またはそれ以上のプラグブル・データベース(PDB)を含めることができます。デフォルトでは、平行・アップグレード・ユーティリティ(catctl.pl)は、同じアップグレード・ウィンドウでCDBとそのすべてのPDBを更新します。平行・アップグレード・ユーティリティでは、コンピュータ処理単位(CPU)の数をを使用して、同時にアップグレードされるPDBの最大数を決定します。平行にアップグレードされるPDBの数を求めるには、平行SQLプロセス件数(-nオプション)を平行PDB SQLプロセス件数(-Nオプション)で割ります。



注意:

CDBの各PDBが提供するすべてのデータベース・サービスの一般的な停止時間に対応できるアップグレード・ウィンドウを計画する必要があります。

プラグブル・データベースのアップグレード構文

```
dbupgrade [-M] -n [-N]
```

- -M: CDB\$ROOTをアップグレード・モードで維持するかどうか、またはそれをアップグレードの完了時に使用可能にするかどうかを指定します。
 - -Mパラメータを使用して平行・アップグレード・ユーティリティを実行する場合、アップグレードでは、CDB\$ROOTとそのすべてのPDBがアップグレード・モードになり、合計アップグレード時間が短縮されます。ただし、CDBとそのすべてのPDBがアップグレードされるまで、どのPDBも起動できません。
 - -Mパラメータを使用して平行・アップグレード・ユーティリティを実行しない場合、CDB\$ROOTがアップグレードされて標準モードで再起動され、標準のバックグラウンド・プロセスが起動されます。アップグレードに成功すると、CDB\$ROOTのみが読み取り/書き込みモードでオープンされます。PDBはすべてMOUNTモードのままです。それぞれのPDBのアップグレード中に、他のPDBのアップグレードが進行中でも各PDBをオンラインにできます。
- -n: PDBの平行・アップグレード・プロセッサの数を指定します。
 - nの値を指定しない場合、-nのデフォルトはCPU_COUNTの値になります。
 - nの値を指定しない場合、その値を使用して平行SQLプロセスの数が算出されます。最大値の制限はありません。最小値は4です。
- -Nでは、PDBのアップグレード時に使用するSQLプロセッサの数を指定します。最大値は8です。最小値は1です。-Nの値を指定しない場合、デフォルト値は2になります。
- 同時実行されるPDBアップグレードの最大数は、-Nの値で割られた-nの値になります。

次に、PDBの平行・アップグレードの方法で使用する手順の概要を示します。

1. バックアップ計画が完了していることを確認します。
2. アップグレード前ツールを実行します。レポートされたエラーをすべて修正します。アップグレード前ツール(preupgrade.jar)は、新しいOracle databaseリリースに同梱されています。

3. パラレル・アップグレード・ユーティリティを実行します。次のアップグレードが順番に実行されます。

- a. サイクル1: CDB\$ROOTが新しいOracleリリースにアップグレードされます
- b. サイクル2からサイクルx: PDB\$SEEDおよび複数のPDBがパラレルにアップグレードされます(アップグレードのサイクル数は、-nで指定したパラメータ設定によって決まります)。

4. アップグレード後のステップを完了します。

例1-1 デフォルトを使用したマルチテナント・アーキテクチャのアップグレードの例(パラメータ・セットなし)

このシナリオでは、CPU_COUNTの値は24です。-nオプションを使用してPDBのパラレル・プロセッサの値を指定しない場合、PDBのパラレル・プロセッサ(-n)のデフォルト値は24です。-Nの値を指定しない場合、SQLプロセッサの数(-N)のデフォルト値は2です。

結果:

12のPDBがパラレルにアップグレードされます(CPU_COUNTを2で割る、つまり24を2で割る)。各PDBに対して割り当てられたパラレルSQLプロセスが2つあります。

例1-2 64のPDBのパラレル・アップグレード・プロセッサおよび4つのパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

このシナリオでは、オプション-n 64を指定して、PDBのパラレル・アップグレード・プロセッサの値を64に設定します。オプション-N 4を指定して、パラレルSQLプロセスの値を4に指定します。

結果:

16のPDBがパラレルにアップグレードされます(64を4で割る)。各PDBに対して4つのパラレルSQLプロセスがあります。

例1-3 20のPDBのパラレル・アップグレード・プロセッサおよび2つのパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

このシナリオでは、オプション-n 20を指定して、PDBのパラレル・アップグレード・プロセッサの値を20に設定します。オプション-N 2を指定して、パラレルSQLプロセスの値を2に指定します。

結果:

10のPDBがパラレルにアップグレードされます(20を2で割る)。各PDBに対して2つのパラレルSQLプロセスがあります。

例1-4 10のPDBのパラレル・アップグレード・プロセッサおよび4つのパラレルSQLプロセスを使用したマルチテナント・アーキテクチャのアップグレードの例

このシナリオでは、オプション-n 10を指定して、PDBのパラレル・アップグレード・プロセッサの値を10に設定します。オプション-N 4を指定して、パラレルSQLプロセスの値を4に指定します。

結果:

2つのPDBがパラレルにアップグレードされます(10を4で割る)。各PDBに対して4つのパラレルSQLプロセスがあります。

親トピック: [マルチテナント・アーキテクチャのパラレル・アップグレード](#)

マルチテナント・コンテナ・データベースの平行・アップグレード

この方法を使用して、1回のアップグレード操作でCDB\$ROOT、PDB\$SEED、およびCDB内のすべてのPDBをアップグレードします。

これはアップグレードの直接的な手順であり、メンテナンスが簡単になるため、停止時間をスケジュールできる場合はこのアプローチを使用することをお勧めします。この手順を使用すると、サーバーの使用可能なプロセッサ(CPU)に応じてマルチテナント・アーキテクチャ・コンテナ・データベースのすべてのPDBが平行にアップグレードされます。

注意:



平行・アップグレード方式を使用してコンテナ全体をアップグレードする場合は、すべての PDB を停止する必要があります。すべての PDB を停止できるように、アップグレードはスケジュールされたアップグレード期間に実行します。

注意:



- 構成の変更を開始する前に、既存のデータベースのバックアップを必ず作成してください。
- COMPATIBLE 初期化パラメータを設定した後は、データベースをダウングレードできません。
- アップグレードを開始する前には、ソース・データベースとターゲット・データベースを最新のバンドル・パッチまたはパッチ・セット更新(BP または PSU)にアップグレードし、ダウングレードを開始する前には、ソース・データベースとターゲット・データベースを最新のリリース更新にアップグレードすることをお勧めします。

1. 適切なバックアップ計画が準備されていることを確認します。
2. すべてのPDBをオープンします。

次に例を示します。

```
SQL> alter pluggable database all open;
```

3. 次の構文を使用してアップグレード前情報ツール(preupgrade.jar)を実行します。

```
/java -jar $New_release_Oracle_home/rdbms/admin/preupgrade.jar [TERMINAL|FILE|DIR  
outputdir] [TEXT|XML] [-c InclusionListOfPDBs] [-C ExclusionListOfPDBs]
```

リストはスペースで区切ります。LinuxとUNIXでは、リスト項目を単一引用符 ' で囲んでリストを定義します。

Windowsシステムでは、リスト項目を二重引用符 " で囲んでリストを定義します。

たとえば、次のコマンドを実行してPDBのPDB1からPDB25までにアップグレード前情報ツールを実行します(Oracle DatabaseのOracleホームの環境変数\$ORACLE_HOME_12.2を/u01/app/oracle/product/12.2.0/dbhome_1/に設定し、新しいOracle DatabaseのOracleホームの環境変数\$ORACLE_HOME_19.1を/u01/app/oracle/product/19.1.0/dbhome_1/に設定している場合)。

LinuxおよびUNIXの場合:

```
java -jar $ORACLE_HOME_19.1/rdbms/admin/preupgrade.jar ¥
```

```
-c 'pdb1 pdb2 pdb3 pdb4 pdb5 pdb6 pdb7 pdb8 pdb9 pdb10 pdb11 pdb12 pdb13¥  
pdb14 pdb15 pdb16 pdb17 pdb18 pdb19 pdb20 pdb21 pdb22 pdb23 pdb24 pdb25'
```

Windowsの場合:

```
java -jar %ORACLE_HOME_19.1%/rdbms/admin/preupgrade.jar ¥  
-c "pdb1 pdb2 pdb3 pdb4 pdb5 pdb6 pdb7 pdb8 pdb9 pdb10 pdb11 pdb12 pdb13¥  
pdb14 pdb15 pdb16 pdb17 pdb18 pdb19 pdb20 pdb21 pdb22 pdb23 pdb24 pdb25"
```

注意:



アップグレード前情報ツールを実行するには、Java 1.5 以上を使用する必要があります。デフォルトでは、直接アップグレード可能な Oracle Database リリースの Java リリースでは、このツールがサポートされます。

4. 生成された修正スクリプトおよびログ・ファイルを確認します。

ORACLE_BASEが定義されている場合はデフォルトで、修正ファイルが次のいずれかのパスに配置されます。

- LinuxおよびUNIXの場合:
\$ORACLE_BASE/cfgtoollogs/db_unique_name/preupgrade
- Windowsの場合:
%ORACLE_BASE%\cfgtoollogs¥db_unique_name¥preupgrade

ORACLE_BASEが定義されていない場合は、修正ファイルが次のいずれかのパスに配置されます。

- LinuxおよびUNIXの場合:
\$ORACLE_HOME/cfgtoollogs/db_unique_name/preupgrade
- Windowsの場合:
%ORACLE_HOME%\cfgtoollogs¥db_unique_name¥preupgrade

マルチテナント・アーキテクチャOracle Databaseでは、アップグレード前情報ツールは統合されたpreupgrade_fixups.sqlスクリプトも作成します。統合された修正スクリプトはcatcon.plを使用して実行できます。統合された修正スクリプトは、preupgrade.jarコマンドを実行したときにオープンされた各コンテナで実行されます。

5. preupgrade_fixupsスクリプト、または個別のPDBスクリプトを実行します。preupgrade_fixups SQLスクリプトは、preupgradeスクリプトによってレポートされたいくつかの問題を解決します。

マルチテナント環境のOracle Databaseデプロイメントでは、ソース・データベースでpreupgrade_fixupspdb-name.sqlスクリプトを実行できます(pdb-nameはPDBの名前です)。PDBの修正スクリプトを生成する場合は、PDBの名前が修正ファイル名に付加されます。

個々のPDB修正スクリプトに加え、catcon.plを使用して、統合されたpreupgrade_fixups.sqlスクリプトを実行できます。統合スクリプトは、preupgrade.jarを実行したときにオープンされた各コンテナで実行されます。

注意:



\$はオペレーティング・システムで予約された記号であるため、PDB\$SEEDの修正スクリプトは、

preupgrade_fixups_pdb_seed.sql になります。

アップグレード前情報ツールによって特定された他のアップグレード前の作業をすべて完了します。

6. (条件付き) Oracle RACデータベースの場合、クラスタ・データベース初期化パラメータをfalseに設定します。

次に例を示します。

```
ALTER SYSTEM SET cluster_database=FALSE SCOPE=spfile;
```

7. 古いOracleホームのデータベースを停止します。

たとえば、db_unique_nameはデータベース名です。

```
$ srvctl stop database -d db_unique_name
```

8. PFILEまたはSPFILEを古いOracleホームから新しいOracleホームにコピーします。

9. SQL*Plusと接続:

```
sqlplus / as sysdba
```

10. CDB\$ROOTインスタンスをアップグレード・モードに移行します。

```
STARTUP UPGRADE
```

11. すべてのPDBをアップグレード・モードに移行します。

```
ALTER PLUGGABLE DATABASE ALL OPEN UPGRADE;
```

12. PDBのステータスをチェックし、アップグレードの準備が完了していることを確認します。

```
SHOW PDBS
```

すべてのPDBについて、ステータスがMIGRATEに設定されていることを確認します。

13. SQL*Plusを終了して、ディレクトリを新しいOracleホームの\$ORACLE_HOME/rdbms/adminに変更します。

```
SQL> EXIT  
$ ORACLE_HOME/bin
```

14. パラレル・アップグレード・ユーティリティ(catctl.pl、シェル・コマンドdbupgradeを使用)を実行し、アップグレードを開始します(-dでディレクトリの場所を指定します)。

```
dbupgrade -d $ORACLE_HOME/rdbms/admin
```

パラメータを指定しない場合、パラレル・アップグレード・ユーティリティは、CPUの数を2で割った数に等しい数のPDBに対してパラレル・アップグレードを実行します。64個のCPUを持つサーバーでは、64を2で割った数に等しい32のPDBが、PDBごとに2つのSQLプロセッサを使用してパラレルにアップグレードされます。CDB\$ROOTは、アップグレードの継続中はNORMALモードのままです。

15. upg_summary.logでアップグレードが成功したことを確認します。必要に応じて他のログを確認します。

16. データベースを再コンパイルできるように、すべてのPDBをオープンします。

```
ALTER PLUGGABLE DATABASE ALL OPEN;
```

17. SQL*Plusを終了して、ディレクトリを新しいOracleホーム・パスの\$ORACLE_HOME/rdbms/adminに変更します。

```
SQL> EXIT
```

```
cd $ORACLE_HOME/rdbms/admin
```

18. catcon.pl スクリプトと、新しいリリースの Oracle Database に付属する postupgrade_fixups.sql スクリプトを実行します。

次の例は、catcon.pl を実行するためのコマンド文字列を示しており、PDB ごとに1つのパラレル・プロセッサを指定する -n パラメータ、実行するアップグレード前スクリプトが存在するパスを指定する -d パラメータ、スクリプトでログ・ファイルを配置する場所を指定する -l パラメータ、および postupgrade_fixups.sql スクリプトのログ・ファイル接頭辞を指定する -b フラグを使用しています。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -n 1 -d ¥  
$ORACLE_HOME/cfgtoollogs/cdbupgr/preupgrade -l /home/oracle/upgrdDBA -b ¥  
postupgrade_fixups postupgrade_fixups.sql
```

19. postupgrade_fixups.sql を実行します。

非CDB:

```
SQL> @rdbms/admin/postupgrade_fixups.sql
```

CDB:

```
$ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b postupgradefixups -d ''.'''  
postupgradefixups.sql
```

20. utlusts.sql を実行して、アップグレードの問題がないことを確認します。

非CDB:

```
SQL> @rdbms/admin/utlusts.sql
```

CDB:

```
$ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b utlu122s -d ''.''' utlusts.sql
```

catcon.pl を使用して utlusts.sql を実行する場合、ログ・ファイル utlu122s0.log が生成されます。このログ・ファイルにはアップグレードの結果が含まれます。アップグレード・レポートの upg_summary.log を確認することもできます。

データベースの状態に関する情報を確認するには、アップグレードの完了後、いつでも必要な回数だけ utlusts.sql を実行します。utlusts.sql スクリプトがエラーを返したり、ステータスが VALID ではないコンポーネントを表示した場合や、コンポーネントに対してリストされたバージョンが最新のリリースではない場合、トラブルシューティングを実行してください。

21. (条件付き) Oracle RAC 環境の場合のみ、次のコマンドを入力して **CLUSTER_DATABASE** の初期化パラメータ値を TRUE に設定し、Oracle RAC データベースを起動します (db_name は Oracle RAC データベースの名前)。


```
ALTER SYSTEM SET CLUSTER_DATABASE=TRUE SCOPE=SPFILE;  
srvctl start database -db db_unique_name
```

これで、データベースがアップグレードされました。

注意:



古い Oracle ソフトウェアを保持している場合、古いソフトウェアを使用してアップグレードしたデータベースを起動しないでください。必ず新しい Oracle Database ホームで起動コマンドを使用して Oracle Database を起動します。



古い Oracle 環境を削除する前に、その環境のデータファイルをすべて新しい Oracle Database 環境に再配置してください。

関連項目:

データファイルの再配置の詳細は、[『Oracle Database管理者ガイド』](#)を参照してください

親トピック: [マルチテナント・アーキテクチャの平行アップグレード](#)