

Oracle® Database

同じオペレーティング・システムで新しいリリースのための 非 CDB の新しいハードウェアへの移行

19c

F16173-02(原本部品番号:F10898-03)

2019年5月

タイトルおよび著作権情報

Oracle Database同じオペレーティング・システムで新しいリリースのための非CDBの新しいハードウェアへの移行, 19c F16173-02

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 Oracle Databaseの移行およびアップグレード](#)
 - [アクティブなデータベースの複製の概要](#)
 - [バックアップ・セットを使用したアクティブなデータベースの複製](#)
- [2 データベースを複製するためのサーバーおよびネットワークの準備](#)
 - [アクティブなデータベースの複製に固有の前提条件](#)
 - [アクティブなデータベースの複製を準備するためのチェックリスト](#)
 - [Oracle Databaseの複製で使用するためのRMANチャンネルの構成](#)
 - [ファイル・システムにまたがる自動チャンネルの構成](#)
 - [アクティブなデータベースの複製のためのチャンネル構成](#)
- [3 ソース・ホストでのプライマリ・インスタンスの準備](#)
 - [ソース・ホストにプライマリ・インスタンスを準備するためのSQL*Netの構成](#)
- [4 宛先ホスト上の補助インスタンスの準備](#)
 - [宛先ホストでのOracle Databaseソフトウェアのインストール](#)
 - [補助インスタンス用の初期化パラメータ・ファイルの作成ステップ](#)
 - [ソース・データベースからのサーバー・パラメータ・ファイルのコピー](#)
 - [補助インスタンス用のパスワード・ファイルの作成](#)
 - [複製データベースのディレクトリの作成](#)
 - [ソース・データベースと複製データベースのデータベース・ファイルでの同じ名前の使用](#)
 - [ソース・データベースと補助インスタンス間のOracle Net接続の確立](#)
 - [ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成](#)
 - [リスナーへの静的サービスの追加](#)
 - [宛先ホスト上の補助インスタンスを準備するためのSQL*Netの構成](#)
 - [SQL*Net構成の確認](#)
 - [補助インスタンスの起動](#)
 - [宛先ホストでOracleキーストアを使用可能にする](#)
 - [RMANの起動およびデータベースへの接続](#)
- [5 プライマリOracle Databaseの複製](#)
 - [環境の検証](#)
 - [RMANのDUPLICATEコマンドの実行](#)
 - [複製データベースの移行後の検証](#)
- [6 フィジカル・スタンバイOracle Databaseでのリフレッシュおよびスイッチオーバー](#)
 - [プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュするステップ](#)
 - [フィジカル・スタンバイ・データベースへのスイッチオーバーの実行](#)
- [7 Oracle Databaseをアップグレードする前の互換性の確認](#)
 - [Oracle Databaseの互換性レベルの確認](#)
 - [Oracle DatabaseのCOMPATIBLE初期化パラメータの値](#)
- [8 Oracle Databaseのアップグレードの準備](#)
 - [新しいOracleホームへのOracleソフトウェアのインストール](#)

- [アップグレード時のOracleホームの新しい場所の選択](#)
 - [単一インスタンス用の新しいOracle Databaseソフトウェアのインストール](#)
- [Oracle Databaseのアップグレードのグラフィカル・ユーザー・インタフェースのメソッド](#)
- [手動、コマンドラインでOracle Databaseをアップグレードする方法](#)
- [Oracle Databaseをアップグレードする前のバックアップ計画の準備](#)
- [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)
 - [Oracle Databaseをアップグレードする場合のパッチ・セットの更新および要件](#)
 - [透過的暗号化Oracleウォレットのコピー](#)
 - [Oracle Databaseのアップグレード時のOracle Net Servicesに関する推奨事項](#)
 - [パスワードの大/小文字の区別とアップグレードについて](#)
 - [大/小文字を区別しないパスワード・バージョンを使用しているアカウントがあるかどうかの確認](#)
 - [読取り専用の表領域を使用したアップグレードの実行](#)
- [Oracle Databaseのアップグレード前情報ツールの使用](#)
 - [アップグレード前情報ツールの環境変数の設定](#)
 - [アップグレード前情報ツールの実行](#)
 - [Oracle Databaseのアップグレードでのアップグレード前情報ツールの警告および推奨事項](#)
 - [アクセス制御リストおよびネットワーク・ユーティリティ・パッケージの更新](#)
 - [依存性の評価およびネットワーク・ユーティリティ・パッケージのACLの追加](#)
 - [アップグレード前情報ツールの出力例](#)
- [Oracle Databaseのアップグレード後のOracle Database Vaultの有効化](#)
 - [Oracle Database Vaultの無効化なしでのOracle Databaseのアップグレード](#)
 - [Oracle Database Vaultが関連する一般的なアップグレード・シナリオ](#)
- [9 Oracle Databaseのアップグレード](#)
 - [DBUAを使用したLinux、UNIXおよびWindowsシステムでのデータベースのアップグレード](#)
 - [非CDBアーキテクチャのOracle Databaseの手動アップグレード](#)
- [10 Oracle Databaseのアップグレード後の作業](#)
 - [アップグレード後の状態ツールを使用したアップグレードの確認](#)
 - [Oracleデータ・ディクショナリの現在の状態の表示方法](#)
 - [Oracle Databaseのアップグレード後に必要な作業](#)
 - [手動アップグレード後のLinuxおよびUNIXシステム上での環境変数の設定](#)
 - [すべての無効なオブジェクトの再コンパイル](#)
 - [無効なオブジェクトの再コンパイルの進行状況の追跡](#)
 - [Oracle Databaseのアップグレード後のOPatchコマンドの実行](#)
 - [Oracle Databaseのアップグレード後にoratabおよびスクリプトが新しいOracleの場所を指すようにするための設定](#)
 - [PL/SQLパッケージおよび依存プロシージャの確認](#)
 - [Oracle管理タイプに依存する表のアップグレード](#)
 - [新しい拡張データ型機能の有効化](#)
 - [パラレル実行サーバーの最大値および最小値の調整](#)
 - [Oracle Databaseのアップグレード後のリカバリ・カタログ・アップグレードの概要](#)
 - [Oracle Databaseのアップグレード後のタイムゾーン・ファイルのバージョンのアップグレード](#)
 - [Oracle Databaseのアップグレード後のDBMS_STATSパッケージで作成された統計表のアップグレード](#)
 - [Oracle Databaseのアップグレード後の外部認証されたSSLユーザーのアップグレード](#)
 - [Oracle XML DBに対するFTPとHTTPのポートおよびHTTP認証の構成](#)

- [Oracle Databaseのアップグレード後のOracle Textが提供するナレッジ・ベースのインストール](#)
- [Oracle Databaseのアップグレード後のOracle Application Express構成の更新](#)
- [外部ネットワーク・サービスへのアクセス制御リスト\(ACL\)の構成](#)
- [SQLNET.ALLOWED_LOGON_VERSIONパラメータの動作の確認](#)
- [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)
 - [データベースのバックアップ](#)
 - [postupgrade_fixups.sqlスクリプトを実行する非CDBのシナリオ](#)
 - [アップグレード後のディクショナリ統計の収集](#)
 - [DBMS_STATSを使用した固定オブジェクトの統計の再収集](#)
 - [パスワードのリセットによる大/小文字区別の強制](#)
 - [10Gパスワード・バージョンを使用するユーザーのパスワードの確認と再設定](#)
 - [新機能の適宜追加](#)
 - [必要な新しい管理手順の作成](#)
 - [表領域アラートのしきい値の設定](#)
 - [ロールバック・セグメントから自動UNDOモードへの移行](#)
 - [Oracle Data Guard Brokerの構成](#)
 - [LONGデータ型からLOBデータ型への表の移行](#)
 - [再構築対象のOracle Text索引の識別](#)
 - [アップグレードした本番Oracle Databaseのテストの概要](#)

はじめに

このガイドは、特定のユースケース・シナリオの完了を支援するために、Oracle Databaseユーザー支援ドキュメントからのトピックを編成したものです。

- [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、RMANを使用してリモート・サーバーにアクティブなデータベースを複製し、Oracle Databaseを新しいハードウェアに移行する場合に役立ちます。アクティブなデータベースを複製した後、データベースをアップグレードできます。

フィジカル・スタンバイOracle Databaseインスタンスでデータベースを複製している間、プライマリOracle Databaseインスタンスを停止する必要はありません。複製を実行する間、ユーザーは、引き続きデータベースに問題なくアクセスできます。

このシナリオの前提条件

- オペレーティング・システムとファイル・システムが、ソース・サーバーと宛先サーバーの両方で同一であることを確認します。
- 宛先サーバーに、新しいリリースのOracle Databaseソフトウェアをインストールしていることを確認します。
- ソース・データベース・ファイルと複製データベース・ファイルで、同じディレクトリ構造を使用していることを確認します。

このシナリオのアウトライン

- **Oracle Databaseの移行およびアップグレード。**バックアップ・セットを含むアクティブなデータベースの複製を使用する移行方法を確認します。
- **データベースを複製するためのサーバーとネットワークの準備。**RMANを使用して、複製用のサーバーおよびネットワークを準備します。
- **ソース・ホストでのプライマリ・インスタンスの準備。**RMANを使用して、プライマリ・データベースを準備します。
- **宛先ホスト上の補助インスタンスの準備。**補助インスタンスをスイッチオーバー用に準備するには、次の手順を実行します。
- **プライマリOracle Databaseインスタンスの複製。**RMAN DUPLICATEを使用します。
- **フィジカル・スタンバイOracle Databaseでのリフレッシュおよびスイッチオーバー。**補助インスタンスにスイッチオーバーするには、次の手順を実行します。
- **Oracle Databaseのアップグレード。**プライマリ・インスタンスでアップグレードを実行します。
- **Oracle Databaseのアップグレード後の作業**アップグレードしたOracle Databaseインスタンスでアップグレード後のタスクを完了します(スイッチバックの準備でアップグレードしたインスタンスのテストなど)。

これらのステップは、このドキュメントの各章に対応しています。

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 Oracle Databaseの移行およびアップグレード

アクティブなOracle Databaseを複製することによって、リモート・ホストにスタンバイOracle Databaseを作成します。

- [アクティブなデータベースの複製の概要](#)

アクティブなデータベースの複製では、ソース・データベースのバックアップは必要ありません。ネットワークを介してデータベース・ファイルを補助インスタンスにコピーして、アクティブなソース・データベースを宛先ホストに複製します。RMANは、必要なファイルをイメージ・コピーまたはバックアップ・セットとしてコピーできます。

- [バックアップ・セットを使用したアクティブなデータベースの複製](#)

アクティブなデータベースの複製の実行にバックアップ・セットを使用することは、アクティブなデータベースの複製のプルベースの方法とも呼ばれます。

アクティブなデータベースの複製の概要

アクティブなデータベースの複製では、ソース・データベースのバックアップは必要ありません。ネットワークを介してデータベース・ファイルを補助インスタンスにコピーして、アクティブなソース・データベースを宛先ホストに複製します。RMANは、必要なファイルをイメージ・コピーまたはバックアップ・セットとしてコピーできます。

アクティブなデータベースの複製の場合、使用される複製方法によって、主な作業をどのチャンネルが実行するかが決まります。バックアップ・セットを使用してアクティブなデータベース複製が実行される場合、複製に関する主な作業は補助チャンネルによって実行されます。イメージ・コピーが使用される場合、主な作業はターゲット・チャンネルによって実行されます。

アクティブなデータベースの複製を実行するには、ターゲット・データベースへの接続が必要です。ソース・ホストと宛先ホスト間のネットワーク帯域幅が制約にならないかぎり、通常は、アクティブなデータベースの複製を使用することをお勧めします。アクティブなデータベースの複製は、必要なセットアップが少なく、実行方法がよりシンプルです。

注意:



アクティブなデータベースの複製では、ソース・データベースでサーバー・パラメータ・ファイルを使用する必要があります。

いくつかのシナリオでは、イメージ・コピーではなく、バックアップ・セットを使用したアクティブなデータベースの複製が適している場合があります。

- データベースの複製中に、マルチセクション・バックアップ、圧縮または暗号化を使用する場合。
- 必要なデータベース・ファイルを複製データベースに転送するために、十分なネットワーク・リソースがソース・データベースにない場合。
- 複製プロセスで使用されるリソースを最小限にする場合。

バックアップ・セットを使用したアクティブなデータベースの複製では、ソース・データベースで使用されるリソースが最小限に抑えられます。

親トピック: [Oracle Databaseの移行およびアップグレード](#)

バックアップ・セットを使用したアクティブなデータベースの複製

アクティブなデータベースの複製の実行にバックアップ・セットを使用することは、アクティブなデータベースの複製のプルベースの方法とも呼ばれます。

この方法では、RMANはTARGETとしてソース・データベースに接続し、AUXILIARYとして補助インスタンスに接続します。その後、補助インスタンスはOracle Net Servicesを介してソース・データベースに接続し、ソース・データベースから必要なデータベース・ファイルをネットワークを介して取得します。

アクティブなデータベースの複製にバックアップ・セットを使用すると、次の利点があります。

- RMANは、未使用ブロック圧縮を使用できるため、ネットワーク経由でトランスポートする必要があるバックアップのサイズが削減されます。
- マルチセクション・バックアップを使用することで、バックアップ・セットをソース・データベース上でパラレルに作成できます。
- ソース・データベース上で作成されたバックアップ・セットを暗号化できます。

親トピック: [Oracle Databaseの移行およびアップグレード](#)

2 データベースを複製するためのサーバーおよびネットワークの準備

アクティブなデータベースを複製してスタンバイ・データベースを作成します。RMANは、プライマリ・データベースからスタンバイ・データベースへ直接データファイルをコピーします。

RMANのDUPLICATE FROM ACTIVE DATABASEコマンドを実行する前に、プライマリ・データベースをマウントまたはオープンする必要があります。

- [アクティブなデータベースの複製に固有の前提条件](#)
FROM ACTIVE DATABASEを指定してDUPLICATEを実行する場合、1つ以上の通常のターゲット・チャンネルおよび1つ以上のAUXILIARYチャンネルが必要です。
- [アクティブなデータベースの複製を準備するためのチェックリスト](#)
RMANを使用してアクティブなデータベースの複製を実行する前に、ソース・データベースとターゲット・データベースを必ず準備してください。
- [Oracle Databaseの複製で使用するためのRMANチャンネルの構成](#)
データベース複製の主な作業は、RMANチャンネルによって実行されます
- [ファイル・システムにまたがる自動チャンネルの構成](#)
すべてのRMANセッションで使用するために、一連の永続的な自動チャンネルを構成します。
- [アクティブなデータベースの複製のためのチャンネル構成](#)
アクティブなデータベースの複製では、ソース・データベースのチャンネル構成を変更したり、補助チャンネルを構成する必要はありません。ただし、RMANがネットワークを介して平行にファイルをコピーできるよう、ソース・データベースのディスク・チャンネルの平行化設定を増加させる必要がある場合があります。

アクティブなデータベースの複製に固有の前提条件

FROM ACTIVE DATABASEを指定してDUPLICATEを実行する場合、1つ以上の通常のターゲット・チャンネルおよび1つ以上のAUXILIARYチャンネルが必要です。

チャンネルを構成しない、または事前に割り当てない場合、RMANは必要なチャンネルをデフォルトで割り当てます。バックアップ・セットによるアクティブな複製のために、チャンネルを構成するか、または手動で割り当てる場合、補助チャンネルの数がターゲット・チャンネルの数以上であることを確認します。

RMANからTARGETとしてソース・データベースに接続するときは、RMANでオペレーティング・システム認証を使用している場合でも、ユーザー名とパスワードを指定する必要があります。補助インスタンスへの接続では、ソース・データベース接続と同じユーザー名とパスワードを使用する必要があります。ソース・データベースはマウントまたはオープン状態である必要があります。ソース・データベースがオープン状態の場合は、アーカイブ処理を有効にする必要があります。ソース・データベースがオープンされていない場合は、一貫性を保った状態で停止されている必要があります。

RMANから補助インスタンスに接続する場合は、次の規則が適用されます。

- 補助インスタンスと同じホストでRMANを実行している場合、ユーザー名とパスワードを使用して接続し、DUPLICATEコマンドにPASSWORD FILE句が含まれていないときは、ネット・サービス名なしでローカルに接続できます。接続するユーザーは、SYSDBA権限またはSYSBACKUP権限を持っている必要があります。
- リモートで接続する場合、またはDUPLICATEコマンドでPASSWORD FILE句を使用する場合、ネット・サービス名を使用して接続する必要があります。最初に補助インスタンス用のパスワード・ファイルを作成する必要があります。

ソース・データベースと補助インスタンスは、同じSYSおよびSYSBACKUPパスワードを使用する必要があります。両方のインスタンスにはパスワード・ファイルが存在している必要があります。パスワード・ファイルには、SYSユーザーとSYSBACKUPユーザーのために、2つ以上のパスワードを含める必要があります。補助インスタンスを開始し、ソース・データベースから接続できるようにすることができます。

パスワード・ファイルに対するDUPLICATEの動作は、複製データベースがスタンバイ・データベースとして動作するかどうかで異なります。作成する複製データベースがスタンバイ・データベースでない場合、デフォルトでは、RMANはパスワード・ファイルをコピーしません。PASSWORD FILEオプションを使用すると、補助インスタンス上の既存のパスワード・ファイルを上書きするようにRMANに指定できます。スタンバイ・データベースを作成すると、デフォルトでは、RMANがパスワード・ファイルをスタンバイ・ホストにコピーするため、既存のパスワード・ファイルは上書きされます。その場合は、PASSWORD FILE句は必要ありません。

アクティブなデータベースの複製を実行する場合は、UNTIL句を使用できません。RMANは、データファイルを一貫性のある時点にリカバリできるように、オンライン・データファイルが完全にコピーされた時間に基づいて時間を選択します。

親トピック: [データベースを複製するためのサーバーおよびネットワークの準備](#)

アクティブなデータベースの複製を準備するためのチェックリスト

RMANを使用してアクティブなデータベースの複製を実行する前に、ソース・データベースとターゲット・データベースを必ず準備してください。

ソースOracle Database:

- ソース・データベースを移行するには、データベース名、一意のデータベース名、リスナー・ポート、サービス名、データベース・ホームのパッチ・レベルおよびSYSのパスワードが必要です。
- 透過的データ暗号化(TDE)を使用してソース・データベースを構成した場合、暗号化されたデータを含むデータベースを複製できるようにするには、ウォレットのバックアップおよびウォレット・パスワードが必要です。
- ソース・データベースは、オープン状態またはマウント状態のどちらかにできます。
 - ソース・データベースがオープン状態になっている場合は、ARCHIVELOGモードである必要があります。
データベースの複製を実行する間、ユーザーは、引き続きソース・データベースに問題なくアクセスできます。
データファイルの複製中、CPUの使用率とネットワーク帯域幅の消費に多少影響があることを考慮してください。
- ソース・データベースがマウント状態の場合は、完全に停止してからマウント状態にします。



注意:

ソース・データベースをマウント状態で保持することを選択した場合、ユーザーはデータベースにアクセスできません。

ターゲットOracle Database:

- ソース・データベースのエディションと同じデータベース・エディションをサポートする、ターゲット・データベース・システム。
- ターゲット・データベース名、一意のデータベース名、補助サービス名および適用されている現在のデータベース・ホームのパッチ・レベルを把握しておいてください。
- 補助インスタンスを設定するためのターゲット・データベースの空きTCPポート。

親トピック: [データベースを複製するためのサーバーおよびネットワークの準備](#)

Oracle Databaseの複製で使用するためのRMANチャンネルの構成

データベース複製の主な作業は、RMANチャンネルによって実行されます

各チャンネルが、複製タスクを実行するOracle Databaseのサーバー・セッションに対応します。複製方法に応じて、RMANは補助チャンネルかターゲット・チャンネルを使用します。

次の方法のいずれかを使用して、チャンネルを構成します。

- CONFIGUREコマンドを使用したチャンネルの自動割当て
- ALLOCATEコマンドを使用したチャンネルの手動割当て

自動チャンネルを構成していない場合は、複製を開始する前に1つ以上のチャンネルを手動で割り当てることができます。チャンネルを割り当てるALLOCATEコマンドは、DUPLICATEコマンドと同じRUNブロック内にある必要があります。

ソース・データベース・チャンネルでAUXILIARYオプションが指定されていなくても、RMANは、宛先ホスト上の複製に、ソース・データベースと同じチャンネル構成を使用できます。

親トピック: [データベースを複製するためのサーバーおよびネットワークの準備](#)

ファイル・システムにまたがる自動チャネルの構成

すべてのRMANセッションで使用するために、一連の永続的な自動チャネルを構成します。

この例では、2つのファイル・システムにまたがる自動ディスク・チャネルを構成します。

```
CONFIGURE DEVICE TYPE DISK PARALLELISM 2;  
CONFIGURE CHANNEL 1 DEVICE TYPE DISK FORMAT '/disk1/%U';  
CONFIGURE CHANNEL 2 DEVICE TYPE DISK FORMAT '/disk2/%U';
```

PARALLELISMが2に設定されているため、次のコマンドはバックアップ・ピースを2つのファイル・システム間で分割します。

```
BACKUP DEVICE TYPE DISK  
DATABASE PLUS ARCHIVELOG;
```

次のLISTコマンドは、データファイルのバックアップがどのようにパラレル化されたかを示します。

```
RMAN> LIST BACKUPSET 2031, 2032;
```

List of Backup Sets

=====

BS Key	Type	LV Size	Device Type	Elapsed Time	Completion Time
--------	------	---------	-------------	--------------	-----------------

2031	Full	401.99M	DISK	00:00:57	19-JAN-07
BP Key: 2038 Status: AVAILABLE Compressed: NO Tag: TAG20070119T100532					
Piece Name: /disk1/24i7ssnc_1_1					

List of Datafiles in backup set 2031

File	LV	Type	Ckp SCN	Ckp Time	Name
------	----	------	---------	----------	------

1	Full	973497	19-JAN-07	/disk3/oracle/dbs/t_db1.f
5	Full	973497	19-JAN-07	/disk3/oracle/dbs/tbs_112.f

BS Key	Type	LV Size	Device Type	Elapsed Time	Completion Time
--------	------	---------	-------------	--------------	-----------------

2032	Full	133.29M	DISK	00:00:57	19-JAN-07
BP Key: 2039 Status: AVAILABLE Compressed: NO Tag: TAG20070119T100532					
Piece Name: /disk2/25i7ssnc_1_1					

List of Datafiles in backup set 2032

File	LV	Type	Ckp SCN	Ckp Time	Name
------	----	------	---------	----------	------

2	Full	973501	19-JAN-07	/disk3/oracle/dbs/t_ax1.f
3	Full	973501	19-JAN-07	/disk3/oracle/dbs/t_undo1.f
4	Full	973501	19-JAN-07	/disk3/oracle/dbs/tbs_111.f

親トピック: [データベースを複製するためのサーバーおよびネットワークの準備](#)

アクティブなデータベースの複製のためのチャンネル構成

アクティブなデータベースの複製では、ソース・データベースのチャンネル構成を変更したり、補助チャンネルを構成する必要はありません。ただし、RMANがネットワークを介して平行にファイルをコピーできるよう、ソース・データベースのディスク・チャンネルの平行化設定を増加させる必要がある場合があります。

アクティブなデータベースの複製に使用される複製方法のタイプによって、複製の主な作業をどのチャンネルが実行するかが決まります。アクティブなデータベースの複製の実行にイメージ・コピーが使用される場合、主な作業はターゲット・チャンネルによって実行されます。複製のパフォーマンスを向上させるには、ソース・データベースに複数のターゲット・チャンネルを構成します。バックアップ・セットを使用してアクティブなデータベース複製が実行される場合、複製に関する主な作業は補助チャンネルによって実行されます。このため、追加の補助チャンネルを割り当てることをお勧めします。補助チャンネルの数は、ターゲット・チャンネルの数以上である必要があります。アクティブな複製にバックアップ・セットを使用することによって平行可も可能になり、複製プロセスの速度を向上させることができます。

親トピック: [データベースを複製するためのサーバーおよびネットワークの準備](#)

3 ソース・ホストでのプライマリ・インスタンスの準備

ネット・サービスを構成して、プライマリOracle Databaseインスタンスを準備します。

- [ソース・ホストにプライマリ・インスタンスを準備するためのSQL*Netの構成](#)
tnsnames.oraファイルにプライマリ・データベースとスタンバイ・データベースのエントリを追加し、ファイルを保存します。

ソース・ホストにプライマリ・インスタンスを準備するための SQL*Netの構成

tnsnames.oraファイルにプライマリ・データベースとスタンバイ・データベースのエントリを追加し、ファイルを保存します。

例3-1 ネット・サービスの追加

```
[oracle @ ora12c-prm ~] $ cd $ ORACLE_HOME / network / admin
[oracle @ ora12c-prm admin] $ cat tnsnames.ora
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/12.2.0/dbhome/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.
DUPDB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = ora12c-dup) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = dupdb )
    )
  )
PRMDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ora12c-prm.localdomain) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = prmdb )
    )
  )
```

親トピック: [ソース・ホストでのプライマリ・インスタンスの準備](#)

4 宛先ホスト上の補助インスタンスの準備

RMANは、補助インスタンスを使用して複製データベースを作成します。複製を開始する前に、補助インスタンスを準備する必要があります。

- [宛先ホストでのOracle Databaseソフトウェアのインストール](#)
ソース・ホストと宛先ホストが異なる場合、補助インスタンスを作成できるように、Oracle Databaseソフトウェアを宛先ホストにインストールする必要があります。
- [補助インスタンス用の初期化パラメータ・ファイルの作成ステップ](#)
補助インスタンスの初期化パラメータ・ファイルには、少なくともDB_NAMEとDB_DOMAINの初期化パラメータが含まれている必要があります。必要に応じて、追加パラメータを指定できます。初期化パラメータ・ファイルが、複製を実行するRMANクライアントと同じホストに存在することを確認します。
- [ソース・データベースからのサーバー・パラメータ・ファイルのコピー](#)
ソース・データベースでサーバー・パラメータ・ファイルが使用されている場合、DUPLICATEコマンドにSPFILEオプションを含めて、ソース・データベースのサーバー・パラメータ・ファイルを補助インスタンスに対して使用するようにRMANに指定します。
- [補助インスタンス用のパスワード・ファイルの作成](#)
補助インスタンスへの接続は、オペレーティング・システム認証またはパスワード・ファイル認証を使用して確立できます。バックアップベースの複製の場合、パスワード・ファイルを作成するか、オペレーティング・システム認証を使用して補助インスタンスに接続できます。アクティブなデータベースの複製の場合、パスワード・ファイル認証を使用する必要があります。
- [複製データベースのディレクトリの作成](#)
宛先ホスト上で、複製データベース・ファイルの格納にRMANによって使用されるディレクトリを作成する必要があります。
- [ソース・データベースと複製データベースのデータベース・ファイルでの同じ名前の使用](#)
ソース・データベースと複製データベースでファイルに同じ名前を使用するには、特定の条件を満たす必要があります。
- [ソース・データベースと補助インスタンス間のOracle Net接続の確立](#)
特定の形式の複製では、ソース・データベースと補助インスタンス間の接続を確立する必要があります。
- [ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成](#)
Oracle Databaseの複製に備えて、ソースOracle DatabaseインスタンスとターゲットOracle Databaseインスタンスの間のネットワークを構成します。
- [補助インスタンスの起動](#)
補助インスタンスを起動するには、作成した初期化パラメータ・ファイルを使用します。
- [宛先ホストでOracleキーストアを使用可能にする](#)
ソース・データベースで透過的暗号化が構成されている場合、補助インスタンスでソース・データベースのOracleソフトウェア・キーストアを使用できることを確認する必要があります。ソース・データベースから宛先ホストに、キーストアを手動でコピーします。
- [RMANの起動およびデータベースへの接続](#)
選択した複製方法の必要に応じて、RMANクライアントを起動してデータベース・インスタンスに接続する必要があります。RMANクライアントは、ネットワーク経由で必要なデータベースに接続可能であるかぎり、任意のホスト上に配置できます。

宛先ホストでのOracle Databaseソフトウェアのインストール

ソース・ホストと宛先ホストが異なる場合、補助インスタンスを作成できるように、Oracle Databaseソフトウェアを宛先ホストにインストールする必要があります。

注意:



ソース・ホストと宛先ホストの両方で、必ず、同じレベルのパッチが適用された同じリリースの Oracle Database ソフトウェアをインストールしてください。

次のいずれかの方法を使用して、ソフトウェアをインストールします。

- Oracle Universal Installer (OUI)で、通常のインストールを実行します。

ソース・データベースと同じリリース番号のOracle Databaseをインストールします。データベースは作成されず、ソフトウェアのみがインストールされます。必要なパッチを適用します。

- ソースのOracleホームをクローニングします。

OUIを使用して、ソースのOracleホームをクローニングします。これによって、ソース・データベースに適用されるすべてのパッチが、複製データベースに存在することになります。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

補助インスタンス用の初期化パラメータ・ファイルの作成ステップ

補助インスタンスの初期化パラメータ・ファイルには、少なくともDB_NAMEとDB_DOMAINの初期化パラメータが含まれる必要があります。必要に応じて、追加パラメータを指定できます。初期化パラメータ・ファイルが、複製を実行するRMANクライアントと同じホストに存在することを確認します。

補助インスタンス用の初期化パラメータ・ファイルを作成する手順

1. 次のいずれかの操作を行います。

- 初期化パラメータ・ファイルをソース・ホストから宛先ホストにコピーし、オペレーティング・システムに固有のデフォルトの場所に配置して、DB_NAMEとDB_DOMAINの初期化パラメータを変更します。

CDBを複製する場合、ENABLE_PLUGGABLE_DATABASEパラメータが存在し、TRUEに設定されていることを確認します。

[ソース・データベースからのサーバー・パラメータ・ファイルのコピー](#)を参照してください。

- 次のステップを実行してください。

- a. テキスト・エディタを使用して、テキストベースの初期化パラメータ・ファイルに使用するための空のファイルを作成し、オペレーティング・システム固有のデフォルトの場所に保存します。
- b. パラメータ・ファイルに、DB_NAMEとDB_DOMAINの初期化パラメータを設定します。これらは、唯一の必須パラメータです。

DB_DOMAINパラメータを設定することで、ネット・サービス名で接続する際に、デフォルトのデータベース・サービスに接続できるようになります。

- c. 補助インスタンスをCDBにする場合は、次のパラメータを設定します。

```
ENABLE_PLUGGABLE_DATABASE=TRUE
```

2. CONTROL_FILESやDB_RECOVERY_FILE_DESTなどの様々な場所に関連するパラメータを設定します。

3. Oracle Real Application Clustersのために必要な初期化パラメータなど、必要に応じてその他の初期化パラメータを設定します。

4. 必須の環境変数(たとえばORACLE_HOMEとORACLE_SID)を設定します。

5. (オプション)次の条件のいずれかが満たされている場合は、複製データベース・ファイルの場所を指定する初期化パラメータを設定します。

- ソース・ホストと宛先ホストが同じである(ローカル・ホストへの複製)。
- 複製データベースが、ソース・ホストとは異なるディレクトリ構造を使用して、データベース・ファイルを保存している。

複製データベース・ファイルに別の名前を指定するために使用した方法に応じて、初期化パラメータ・ファイルに次のパラメータを1つ以上含めます: CONTROL_FILES、DB_FILE_NAME_CONVERT、LOG_FILE_NAME_CONVERT、DB_CREATE_FILE_DEST、DB_CREATE_ONLINE_FILE_DEST、およびRECOVERY_FILE_DEST。



注意:

宛先ホストと補助インスタンスのサーバー・セッションから、指定したすべてのパスにアクセスできることを確

認することをお薦めします。

Oracle Databaseバックアップおよびリカバリ・ユーザズ・ガイドの複製データベースのデータベース・ファイル名の生成方法を参照してください。

6. SQL*Plusを起動し、SYSDBA権限またはSYSBACKUP権限で補助インスタンスに接続します。補助インスタンスをNOMOUNTモードで起動します。ファイルがデフォルトの場所にある場合は、STARTUPコマンドにPF FILEパラメータは必要ありません。

```
SQL> STARTUP NOMOUNT;
```

例4-1 補助インスタンス用のサンプル初期化パラメータ・ファイル

```
DB_NAME=dupdb
CONTROL_FILES=(/dup/oracle/oradata/prod/control01.ctl,
               dup/oracle/oradata/prod/control02.ctl)
DB_FILE_NAME_CONVERT=(/oracle/oradata/prod/, /dup/oracle/oradata/prod/)
LOG_FILE_NAME_CONVERT=(/oracle/oradata/prod/redo, /dup/oracle/oradata/prod/redo)
```

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

ソース・データベースからのサーバー・パラメータ・ファイルのコピー

ソース・データベースでサーバー・パラメータ・ファイルが使用されている場合、DUPLICATEコマンドにSPFILEオプションを含めて、ソース・データベースのサーバー・パラメータ・ファイルを補助インスタンスに対して使用するようRMANに指定します。

バックアップベースの複製の場合は、サーバー・パラメータ・ファイルはバックアップからリストアされます。アクティブなデータベースの複製の場合、サーバー・パラメータ・ファイルはソース・データベースから補助インスタンスにコピーされます。

ソース・データベースでテキストベースの初期化パラメータ・ファイルが使用されている場合は、DUPLICATEコマンドのPFILE句を使用して、ソース・データベースの初期化パラメータ・ファイルを補助インスタンスにコピーします。

ソース・データベースのサーバー・パラメータ・ファイルからコピーまたはリストアされた値は、DUPLICATEのSPFILEまたはSET句のPARAMETER_VALUE_CONVERTオプションを使用して変更できます。たとえば、SET句を使用して、補助インスタンスのサーバー・パラメータ・ファイルのDB_FILE_NAME_CONVERTパラメータの値を変更できます。

ソース・データベースでサーバー・パラメータ・ファイルを使用していない場合や、RMANがサーバー・パラメータ・ファイルのバックアップをリストアできない場合は、[補助インスタンス用の初期化パラメータ・ファイルの作成ステップ](#)で説明する方法で、テキストベースの初期化パラメータ・ファイルを手動で作成する必要があります。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

補助インスタンス用のパスワード・ファイルの作成

補助インスタンスへの接続は、オペレーティング・システム認証またはパスワード・ファイル認証を使用して確立できます。バックアップベースの複製の場合、パスワード・ファイルを作成するか、オペレーティング・システム認証を使用して補助インスタンスに接続できます。アクティブなデータベースの複製の場合は、パスワード・ファイル認証を使用する必要があります。

パスワード・ファイル認証を使用してデータベースに接続するには、そのデータベースのパスワード・ファイルを作成する必要があります。リモート・ホストに複製する場合、パスワード・ファイルの設定は必須です。パスワード・ファイルのデフォルトの場所は、Windowsの場合には\$ORACLE_BASE¥database、LinuxおよびUNIXの場合には\$ORACLE_BASE/db\$です。

注意:



RMAN 複製を使用してスタンバイ・データベースを作成する場合、パスワード・ファイルは常にコピーされます。その他すべてのケースにおいて、DUPLICATE コマンドに PASSWORD FILE オプションを指定した場合にのみ、パスワード・ファイルがコピーされます。

次のいずれかのオプションを使用して、宛先ホストの補助インスタンス用にパスワード・ファイルを作成します。

- オペレーティング・システムに固有のユーティリティを使用して、ソース・データベースのパスワード・ファイルを宛先ホストにコピーし、名前を補助インスタンス名と一致するように変更します。これは、ソース・ホストと宛先ホストが同じプラットフォームにある場合にのみ該当します。
- パスワード・ファイルを手動で作成します。SYSDBAユーザーおよびSYSBACKUPユーザーのパスワードが、ソース・データベースと補助インスタンスで同じになっていることを確認します。
- orapwdユーティリティを使用して、パスワード・ファイルを作成します。SYSBACKUPオプションは、新しいパスワード・ファイルでSYSBACKUPエントリを作成します。

次の例では、12.2形式のorapwrdという名前のパスワード・ファイルが作成され、オペレーティング・システム・ファイルのデフォルトの場所に配置されます。

```
orapwd FILE='/u01/oracle/dbs/orapwrd' FORMAT=12.2
```

- DUPLICATE... FROM ACTIVE DATABASEコマンドにPASSWORD FILEオプションを指定します。

RMANは、ソース・データベース・パスワード・ファイルを宛先ホストにコピーし、既存のすべての補助インスタンスのパスワード・ファイルを上書きします。この方法は、複製データベースで使用できるようにするパスワードが、ソース・データベースのパスワード・ファイルに複数含まれている場合に有効です。

アクティブなデータベースの複製を使用する際は、パスワード・ファイルにSYSユーザーとSYSBACKUPユーザーのために、2つ以上のパスワードを含める必要があります。これらのパスワードは、ソース・データベースのパスワードと一致する必要があります。

注意:



FROM ACTIVE DATABASE オプションを使用してスタンバイ・データベースを作成する場合、RMAN は、パスワード・ファイルを常にスタンバイ・ホストにコピーします。

関連項目:

[Oracle Database管理者ガイド](#)

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

複製データベースのディレクトリの作成

宛先ホスト上で、複製データベース・ファイルの格納にRMANによって使用されるディレクトリを作成する必要があります。

これには、データファイル、制御ファイル、オンラインREDOログ・ファイルおよび一時ファイルを格納するディレクトリが含まれます。

NOFILENAMECHECK句を使用して、データベース・ファイルの名前がソース・データベースと複製データベースで同じ場合に、RMANでエラーを表示する必要がないことを示します。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

ソース・データベースと複製データベースのデータベース・ファイルでの同じ名前の使用

ソース・データベースと複製データベースでファイルに同じ名前を使用するには、特定の条件を満たす必要があります。

最もシンプルな複製方法は、ソース・データベースと同じディレクトリ構造とファイル名を使用するように複製データベースを構成することです。同じディレクトリ構造と名前を使用できるのは、リモート・ホストに複製する場合のみです。

同じディレクトリ構造とファイル名を使用することは、使用する環境が次の要件を満たしていることを意味します。

- ソース・データベースがASMディスク・グループを使用している場合、複製データベースは同じ名前のASMディスク・グループを使用する必要があります。
- ソース・データベース・ファイルがOracle Managed Filesの場合、補助インスタンスは、ソース・データベースと同じディレクトリの場所に、DB_CREATE_FILE_DESTパラメータを設定する必要があります。ソース・ホスト上と宛先ホスト上でディレクトリが同じであっても、Oracle Databaseは複製ファイルに対して相対ファイル名を選択します。
- ソース・データベースのデータベース・ファイルの名前にパスが含まれている場合、そのパス名は複製データベースにあるパス名と同じである必要があります。
- Oracle Real Application Clusters (RAC)環境の場合は、ソース・データベースと宛先データベースのORACLE_SIDパラメータに、同じ値を使用します。

前述の推奨事項に従って環境を構成する場合、複製ファイルの名前の指定で必要となる追加の構成はありません。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

ソース・データベースと補助インスタンス間のOracle Net接続の確立

特定の形式の複製では、ソース・データベースと補助インスタンス間の接続を確立できる必要があります。

次の条件のいずれかが該当する場合、補助インスタンスはOracle Net Servicesを介して使用可能である必要があります。

- RMANクライアントが宛先ホスト以外のホストから実行されている
- 複製方法としてアクティブなデータベースの複製が選択されている
- 宛先ホストがソース・ホストとは異なる

アクティブなデータベースの複製を実行するには、SYSDBAまたはSYSBACKUP権限とネット・サービス名を使用して補助インスタンスに接続する必要があります。RMANがTARGETとして接続しているソース・データベースは、このネット・サービス名を使用して、補助データベース・インスタンスに直接接続します。

Oracle Netの接続を確立し静的リスナーを設定する手順

- データベースに接続するようにクライアントを構成し、リスナーの静的サービス情報を追加するには、[Oracle Database管理者ガイド](#)の手順を実行します。

例4-2 例: ソース・データベースと補助インスタンス間のOracle Net接続の確立

ソース・データベースのDB_NAMEがsrcで、ソース・ホストがsrc.example.comと仮定します。補助インスタンスのDB_NAMEはdupで、補助インスタンスはホストdup.example.com上に作成されます。

次のステップを使用して、ソース・データベースと補助インスタンス間のOracle Net接続を確立します。

1. ソース・データベースのtnames.oraファイルに、複製データベースに対応する次のエントリを追加します。

```
dupdb =
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=dup.example.com) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=dup)))
```

2. 宛先ホストの\$ORACLE_HOME/admin/networkフォルダに、tnames.oraファイルを作成します。ソース・データベースに対応する次のエントリを追加します。

```
srcdb =
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=src.example.com) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=src)))
```

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成

Oracle Databaseの複製に備えて、ソースOracle DatabaseインスタンスとターゲットOracle Databaseインスタンスの間のネットワークを構成します。

- [リスナーへの静的サービスの追加](#)
listener.oraファイルにスタンバイ・データベースの静的エントリを挿入し、ファイルを保存します。
- [宛先ホスト上の補助インスタンスを準備するためのSQL*Netの構成](#)
tnsnames.oraにプライマリ・データベースとスタンバイ・データベースのエントリを追加し、ファイルを保存します。
- [SQL*Net構成の確認](#)

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

リスナーへの静的サービスの追加

listener.oraファイルにスタンバイ・データベースの静的エントリを挿入し、ファイルを保存します。

例4-3 リスナーへの静的サービスの追加

```
[oracle @ ora12c-dup admin] $ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/oracle/product/12.2.0/dbhome/network/admin/listener.ora
# Generated by Oracle configuration tools.
  SID_LIST_LISTENER =
  (SID_LIST =
  (SID_DESC =
  (GLOBAL_DBNAME = dupdb )
  (ORACLE_HOME = /u01/app/oracle/product/12.2.0/dbhome)
  (SID_NAME = dupdb )
  )
  )
  LISTENER =
  (DESCRIPTION_LIST =
  (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP) (HOST = ora12c-dup.localdomain) (PORT = 1521))
  )
  (DESCRIPTION =
  (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
  )
  )
  ADR_BASE_LISTENER = / u01 / app / oracle
[oracle @ oel62-ora12c-dup admin] $ lsnrctl status
```

親トピック: [ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成](#)

宛先ホスト上の補助インスタンスを準備するためのSQL*Net の構成

tnsnames.oraにプライマリ・データベースとスタンバイ・データベースのエントリを追加し、ファイルを保存します。

例4-4 ネット・サービスの追加

```
[oracle @ ora12c-dup ~] $ cd $ ORACLE_HOME / network / admin
[oracle @ ora12c-dup admin] $ cat tnsnames.ora
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/12.2.0/dbhome/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.
DUPDB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = ora12c-dup) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = dupdb )
    )
  )
PRMDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ora12c-prm.localdomain) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = prmdb )
    )
  )
```

親トピック: [ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成](#)

SQL*Net構成の確認

ソース・サーバーおよび宛先サーバーからtnspingコマンドを実行します。

```
[oracle @ ora12c-prm admin] $ tnsping standby-db-unique-name  
[oracle @ ora12c-dup admin] $ tnsping primary-db-unique-name
```

通信が確実に適切に確立されるように、sysdbaとしてリモートで接続することをお勧めします。

次の形式でコマンドを使用し、SQL*Plusを起動します。

```
sqlplus {username | /} [as sysdba]
```

次に例を示します。

```
$ sqlplus / AS SYSDBA  
Enter password: password
```

または、デフォルト以外のデータベースに接続されているSQL*Plusを起動するには、次の形式でSQL*Plusコマンドを入力します。

```
$> sqlplus username/password@connect_identifier
```

親トピック: [ソースOracle DatabaseとターゲットOracle Databaseの間のネットワークの構成](#)

補助インスタンスの起動

補助インスタンスを起動するには、作成した初期化パラメータ・ファイルを使用します。

RMANは、複製の一環として補助インスタンスを停止して再起動します。このため、補助インスタンス用にサーバー側の初期化パラメータ・ファイルをデフォルトの場所に作成することをお勧めします。サーバー側の初期化パラメータ・ファイルがデフォルトの場所がない場合は、DUPLICATEコマンドのPFILEパラメータに、クライアント側の初期化パラメータ・ファイルを指定する必要があります。

注意:



補助インスタンスに制御ファイルがないため、このインスタンスは NOMOUNT モードでのみ起動できます。制御ファイルの作成、補助インスタンスのマウントまたはオープンは実行しないでください。

補助インスタンスを起動する手順

1. RMANを起動します。

```
% rman
```

2. SYSDBA権限またはSYSBACKUP権限を持つユーザーとして、補助インスタンスに接続します。次の例では、パスワード・ファイル認証を使用して、補助インスタンスに接続します。

```
RMAN> CONNECT SYS@dupdb AS SYSDBA;
```

次の例では、オペレーティング・システム認証を使用して、SYSBACKUP権限で補助インスタンスに接続します。

```
RMAN> CONNECT / AS SYSBACKUP;
```

3. 補助インスタンスをNOMOUNTモードで起動します。

```
RMAN > STARTUP FORCE NOMOUNT;
```

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

宛先ホストでOracleキーストアを使用可能にする

ソース・データベースで透過的暗号化が構成されている場合、補助インスタンスでソース・データベースのOracleソフトウェア・キーストアを使用できることを確認する必要があります。ソース・データベースから宛先ホストに、キーストアを手動でコピーします。Oracleソフトウェア・キーストアには、次の操作に使用されるTDEマスター・キーが含まれています。

- バックアップベースの複製を実行するときに、暗号化されたバックアップを復号化します。
- TDEで暗号化されたデータベースまたは表領域のアクティブなデータベースの複製を実行するときに、データベースまたは表領域データを復号化します。

複製データベースでのキーストアの要件は次のとおりです。

- キーストアは、デフォルトの場所、またはsqlnet.oraファイルに指定された場所にある必要がある。
- データベースがファイルにアクセスできるように、Oracleキーストア・ファイルに権限を設定する必要がある。
- 複製中に補助インスタンスは再起動されるため、Oracleソフトウェア・キーストアは使用できなくなります。補助インスタンスからキーストアへのアクセスを確保するため、sqlnet.oraファイルのENCRYPTION_WALLET_LOCATIONパラメータをキーストアの場所をポイントするように設定します。

ENCRYPTION_WALLET_LOCATION sqlnet.oraパラメータは、Oracle Databaseリリース19cでは非推奨です。ソフトウェア・キーストアの場所を構成するには、WALLET_ROOT初期化パラメータとTDE_CONFIGURATION初期化パラメータを使用します。

- Oracle Real Application Clusters (Oracle RAC)では、補助インスタンスをOracle Grid Infrastructureリスナーに静的に登録し、Oracle Gridホームのsqlnet.oraファイル内のENVSパラメータを使用して、キーストアの場所とデータベースの一意の名前を設定する環境変数を指定します。

次の例では、sqlnet.oraのENVSパラメータを設定して、キーストアの場所と一意のデータベース名を指定します。

```
(ENVS="ORACLE_UNQNAME=cdbbrpt1,  
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE) (METHOD_DATA=(DIRECTORY=/etc/ORACLE/WALLETS/cd  
brpt1)))")
```

- ソース・データベースでパスワードベースのソフトウェア・キーストア(自動ログイン・ソフトウェア・キーストアではなく)が使用される場合、複製を開始する前に、キーストア・パスワードを指定する必要があります。

DECRYPTION WALLET OPEN IDENTIFIED BY句を指定してSETコマンドを使用し、キーストアをオープンするために使用するパスワードを指定します。

次のコマンドは、キーストアを開くために使用するパスワードを指定します(passwordは、入力する実際のパスワードのプレースホルダ)。

```
SET DECRYPTION WALLET OPEN IDENTIFIED BY password;
```

関連項目:

- sqlnet.oraでのOracleキーストアの場所の指定の詳細は、[『Oracle Database Advanced Securityガイド』](#)を参照してください。
- Oracleキーストアのデフォルトの場所の詳細は、[『Oracle Database Advanced Securityガイド』](#)を参照してください。

- 標準的なOracleキーストアから自動ログイン・キーストアへの変換については、[『Oracle Database Advanced Securityガイド』](#)を参照してください。
- SETコマンドの詳細は、[『Oracle Databaseバックアップおよびリカバリ・リファレンス』](#)を参照してください。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

RMANの起動およびデータベースへの接続

選択した複製方法の必要に応じて、RMANクライアントを起動してデータベース・インスタンスに接続する必要があります。RMANクライアントは、ネットワーク経由で必要なデータベースに接続可能であるかぎり、任意のホスト上に配置できます。

RMANを起動し、ターゲット・インスタンスおよび補助インスタンスに接続する手順

1. 必要なデータベース・インスタンスに接続可能なホスト上で、RMANを起動します。

たとえば、宛先ホストのオペレーティング・システム・プロンプトで、次のコマンドを入力します。

```
% rman
```

2. RMANプロンプトで、選択した複製方法に必要なデータベース・インスタンスに対して、CONNECTコマンドを実行します。

注意:



CDB 全体、または 1 つ以上の PDB を複製する場合は、両方のインスタンスの root に接続します。

- イメージ・コピーを使用したアクティブなデータベースの複製の場合、ソース・データベースにTARGETとして、補助インスタンスにAUXILIARYとして接続する必要があります。AUXILIARYインスタンスに接続するには、ネット・サービス名を指定する必要があります。リカバリ・カタログ接続は任意です。両方のインスタンスで、複製を実行しているユーザーのパスワードを同じにする必要があります。SYSDBA権限またはSYSBACKUP権限を持つユーザーは、誰でも複製を実行できます。
- バックアップ・セットを使用したアクティブなデータベースの複製の場合、ネット・サービス名を使用してソース・データベースにTARGETとして接続する必要があります。補助インスタンスは、このネット・サービス名を使用してソース・データベースに接続し、複製に必要なバックアップ・セットを取得します。補助インスタンスにAUXILIARYとして接続します。リモートから補助インスタンスに接続するか、またはDUPLICATEコマンドのPASSWORD FILEオプションを使用する場合は、ネット・サービス名で補助インスタンスに接続します。両方のインスタンスで、複製を実行しているユーザーのパスワードを同じにする必要があります。SYSDBA権限またはSYSBACKUP権限を持つユーザーは、誰でも複製を実行できます。リカバリ・カタログ接続は任意です。
- ターゲット接続を行わないバックアップベースの複製の場合、AUXILIARYとして補助インスタンスに接続し、CATALOGとしてリカバリ・カタログに接続する必要があります。
- ターゲット接続を行うバックアップベースの複製の場合、TARGETとしてソース・データベースに接続し、AUXILIARYとして補助インスタンスに接続する必要があります。リカバリ・カタログは任意です。
- ターゲットおよびリカバリ・カタログへの接続を行わないバックアップベースの複製の場合、AUXILIARYとして補助インスタンスに接続する必要があります。

注意:



TARGETとしてスタンバイ・データベースに接続することはできません。

例: アクティブなデータベースの複製実行中の必要なデータベースへの接続

この例では、ネット・サービス名を使用して、ソース・データベースと補助インスタンスへの接続が確立されます。ソース・データベースのネット・サービス名はsrcdbで、補助インスタンスではdupdbです。

宛先ホストから必要なデータベースに接続する手順

1. 宛先ホストでRMANクライアントを起動します。

```
% rman
```

2. ソース・データベースにTARGETとして接続します。

```
RMAN> CONNECT TARGET sys@srcdb;
```

要求されたらソース・データベースのSYSユーザーのパスワードを入力します。

3. 補助インスタンスにAUXILIARYとして接続します。

```
RMAN> CONNECT AUXILIARY sys@dupdb;
```

要求されたら補助インスタンスのSYSユーザーのパスワードを入力します。

親トピック: [宛先ホスト上の補助インスタンスの準備](#)

5 プライマリOracle Databaseの複製

プライマリOracle Database全体を宛先サーバーの補助Oracle Databaseインスタンスに複製するステップを完了します。

- [環境の検証](#)
複製を開始する前に、ステップを完了します。
- [RMANのDUPLICATEコマンドの実行](#)
宛先サーバーでDUPLICATEコマンドを実行します。
- [複製データベースの移行後の検証](#)
データベースが宛先サーバーでリストアおよびリカバリされているかどうかを検証します。

環境の検証

複製を開始する前に、ステップを完了します。

1. ソース・データベース・ホストが接続先ホストからアクセス可能であることを確認します。2つのホスト間でSSHを使用できる必要があります。
2. 宛先ホストでTNSPINGユーティリティを使用して、ソース・ホストのリスナー・ポートが正常に機能していることを確認します。

次に例を示します。

```
tnsping source host:1521
```

3. 宛先ホストで簡易接続を使用して、ソース・データベースへの接続を検証します。

```
host:port/servicename
```

次に例を示します。

```
sqlplus system@ip-address:1521/proddb
```

接続文字列が64文字以内であることを確認します。

4. ソース・データベース・ホームからターゲット・データベースに、必要なsqlpatchファイル(ロールバック用)をコピーします。
5. ソース・データベースで少なくとも1つのアーカイブ・ログが作成されていることを確認してください。そうしないと、RMAN複製がエラーで失敗します。
6. ソース・データベースでウォレットが使用されている場合は、パスワード・ベースのウォレットをバックアップし、宛先ホスト上の標準の場所にコピーします。

次に例を示します。

```
/opt/oracle/dcs/commonstore/wallets/tde/db_unique_name/
```

7. ソース・データベースの互換性パラメータが、11.2.0.4データベースについては少なくとも11.2.0.4.0に、12.1.0.2データベースについては少なくとも12.1.0.2.0に設定されていることを確認してください。

親トピック: [プライマリOracleデータベースの複製](#)

RMANのDUPLICATEコマンドの実行

宛先サーバーでDUPLICATEコマンドを実行します。

次に例を示します。

```
DUPLICATE DATABASE TO dupdb
FOR STANDBY
FROM ACTIVE DATABASE
PASSWORD FILE
SPFILE PARAMETER_VALUE_CONVERT=' /app/dbhome1', ' /app/db_home2'
SET db_file_name_convert=' /app/dbhome1/dbs', ' /app/db_home2/database/dbs'
SET log_file_name_convert=' /app/dbhome1/log', ' /app/db_home2/logfiles'
NOFILENAMECHECK;
```

親トピック: [プライマリOracleデータベースの複製](#)

複製データベースの移行後の検証

データベースが宛先サーバーでリストアおよびリカバリされているかどうかを検証します。

次に例を示します。

V\$DATABASEは、制御ファイルからのデータベースに関する情報を示します。

```
[oracle @ oraDB-dup admin] $ sqlplus
SQL> select name, open_mode, dbid, created from v$database;
```

V\$INSTANCEは、現行インスタンスの状態を示します。

```
SQL> select instance_name, host_name from v$instance;
```

V\$DATAFILEは、制御ファイルからのデータベースに関する情報を示します。

V\$CONTROLFILEは、制御ファイルの名前を示します。

V\$LOGFILEは、REDOログ・ファイルの情報を示します。

V\$TEMPFILEは一時ファイルの情報を示します。

```
SQL> select name from v$datafile
union
select name from v$controlfile
union
select member from v$logfile
union
select name from v$tempfile;
```

V\$VERSIONは、Oracle Databaseのバージョン番号を示します。

```
SQL> select banner from v$version;
```

親トピック: [プライマリOracleデータベースの複製](#)

6 フィジカル・スタンバイOracle Databaseでのリフレッシュおよびスイッチオーバー

プライマリOracle Databaseに加えられた変更でフィジカル・スタンバイのOracle Databaseをリフレッシュした後、ロールを切り替えます。

- [プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュするステップ](#)
FROM SERVICE句を指定してRECOVER STANDBY DATABASEコマンドを使用し、プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュします。
- [フィジカル・スタンバイ・データベースへのスイッチオーバーの実行](#)
このステップでは、フィジカル・スタンバイ・データベースへのスイッチオーバーを実行する方法について説明します。

プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュするステップ

FROM SERVICE句を指定してRECOVER STANDBY DATABASEコマンドを使用し、プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュします。

この例では、プライマリ・データベースのDB_UNIQUE_NAMEはMAINで、そのネット・サービス名はprimary_dbであるとします。スタンバイ・データベースのDB_UNIQUE_NAMEは、STANDBYで、そのネット・サービス名はstandby_dbだとします。

プライマリ・データベースで行われた変更を使用してフィジカル・スタンバイ・データベースをリフレッシュする手順

1. 次の前提条件が満たされていることを確認してください。

- Oracle Net接続性が、フィジカル・スタンバイ・データベースとプライマリ・データベースとの間で確立されている。
これは、フィジカル・スタンバイ・データベースのtnsnames.oraファイルに、プライマリ・データベースに対応するエントリを加えることによって行うことができます。
- プライマリ・データベースとフィジカル・スタンバイ・データベースのパスワード・ファイルが同じである。
- プライマリ・データベースおよびフィジカル・スタンバイ・データベースの初期化パラメータファイルの、COMPATIBLEパラメータが12.0に設定されている。

2. RMANを起動して、ターゲットとしてフィジカル・スタンバイ・データベースに接続します。リカバリ・カタログにも接続することをお勧めします。

次のコマンドでTARGETとしてフィジカル・スタンバイ・データベースに接続し、CATALOGとしてリカバリ・カタログに接続します。フィジカル・スタンバイ・データベースへの接続は、sbuユーザーを使用して確立されます。このユーザーにはSYSBACKUP権限が与えられています。フィジカル・スタンバイ・データベースのネット・サービス名はstandby_dbで、リカバリ・カタログのネット・サービス名はcatdbです。

```
CONNECT TARGET "sbu@standby_db AS SYSBACKUP";  
CONNECT CATALOG rman@catdb;
```

3. FROM SERVICE句を指定してRECOVER STANDBY DATABASEコマンドを使用し、フィジカル・スタンバイ・データベースをロールフォワードします。

FROM SERVICE句は、フィジカル・スタンバイをロールフォワードする際に使用する必要があるプライマリ・データベースのサービス名を指定します。スタンバイ・データベースは、ロール・フォワード操作後に再起動されます。

次の例では、サービス名がprimary_dbであるプライマリ・データベースを使用して、フィジカル・スタンバイ・データベースをロールフォワードします。

```
RECOVER STANDBY DATABASE FROM SERVICE primary_db;
```

4. (Active Data Guardの場合のみ)次のステップを実行して、REDOデータをリカバリし、フィジカル・スタンバイ・データベースを読取り専用モードでオープンします。

```
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE UNTIL CONSISTENT;  
ALTER DATABASE OPEN READ ONLY;
```

5. フィジカル・スタンバイ・データベース上の管理リカバリ・プロセスを開始します。

次のコマンドでは、管理リカバリ・プロセスを開始します。

```
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
```

Data Guard Brokerを使用する場合、次のコマンドを使用して、管理リカバリ・プロセスを開始します。

```
DGMGRL> edit database standby_db set state='APPLY-ON';
```

関連項目:

Oracle Netの接続確立の詳細は、[『Oracle Database Net Services管理者ガイド』](#)を参照してください。

親トピック: [フィジカル・スタンバイOracle Databaseでのリフレッシュおよびスイッチオーバー](#)

フィジカル・スタンバイ・データベースへのスイッチオーバーの実行

これらのステップでは、フィジカル・スタンバイ・データベースへのスイッチオーバーを実行する方法について説明します。

注意:



プライマリ・データベースおよびスタンバイ・データベースに接続する遠隔同期インスタンス(または優先および代替の遠隔同期インスタンスの組合せ)が存在する場合、スタンバイにスイッチオーバーする手順はこのトピックで説明しているものと同じです。遠隔同期インスタンスが使用可能か否かは、スイッチオーバーに影響しません。スイッチオーバー中、プライマリとスタンバイは直接相互に通信し、遠隔同期インスタンスを意識せずにスイッチオーバーのロールの推移ステップを実行できる必要があります。スイッチオーバー後、遠隔同期インスタンスが新しいロールの2つのデータベースにサービスを提供できるように構成を正しく設定する方法の例は、*Oracle Data Guard 概要および管理の遠隔同期インスタンスの使用方法を参照してください。*

1. ターゲット・スタンバイ・データベースのスイッチオーバーの準備ができていることを確認します。

新しいスイッチオーバーの文にはVERIFYオプションがあり、スイッチオーバーに必要な多くの条件のチェックが実行されることとなります。チェックされる項目の一部は次のとおりです。Redo Applyがスイッチオーバー・ターゲットで実行しているか、スイッチオーバー・ターゲットのリリース・バージョンが12.1以降か、スイッチオーバー・ターゲットが同期しているか、および実行中のMRPがあるかどうか。

プライマリ・データベースのDB_UNIQUE_NAMEがBOSTONで、スイッチオーバー・ターゲット・スタンバイ・データベースのDB_UNIQUE_NAMEがCHICAGOだとします。プライマリ・データベースBOSTONで、次のSQL文を発行し、スイッチオーバー・ターゲットCHICAGOのスイッチオーバーの準備ができていることを確認します。

```
SQL> ALTER DATABASE SWITCHOVER TO CHICAGO VERIFY;  
ERROR at line 1:  
ORA-16470: Redo Apply is not running on switchover target
```

この操作が正常に実行されると、「データベースが変更されました。」というメッセージが返されますが、この例ではORA-16470エラーが返されます。このエラーは、スイッチオーバー・ターゲットCHICAGOのスイッチオーバーの準備ができていないことを意味します。スイッチオーバー操作の前にREDO Applyを開始する必要があります。

REDO Applyを開始した後で、次の文を再度発行します。

```
SQL> ALTER DATABASE SWITCHOVER TO CHICAGO VERIFY;  
ERROR at line 1:  
ORA-16475: succeeded with warnings, check alert log for more details
```

スイッチオーバー・ターゲットCHICAGOはスイッチオーバーの準備ができています。しかし、ORA-16475エラーで示された警告は、スイッチオーバーのパフォーマンスに影響を与える可能性があります。アラート・ログには次のようなメッセージが含まれます。

```
SWITCHOVER VERIFY WARNING: switchover target has dirty online redo logfiles that require clearing. It takes time to clear online redo logfiles. This may slow down switchover process.
```

問題を解決できる場合、またはスイッチオーバーのパフォーマンスが重要ではない場合は、これらの警告を無視できます。必要だと判断した解決を行った後で、次のSQL文を再度発行します。

```
SQL> ALTER DATABASE SWITCHOVER TO CHICAGO VERIFY;
```

```
Database altered.
```

スイッチオーバー・ターゲットCHICAGOのスイッチオーバーの準備ができました。

2. 次のSQL文を発行して、プライマリ・データベースBOSTONでスイッチオーバーを開始します。

```
SQL> ALTER DATABASE SWITCHOVER TO CHICAGO;  
Database altered.
```

この文がエラーなしに終了した場合は、ステップ3に進みます。

エラーが発生したら、古いプライマリ・データベース(BOSTON)および古いスタンバイ・データベース(CHICAGO)をマウントします。両方のデータベースで、V\$DATABASEのDATABASE_ROLEを問い合わせます。BOSTONとCHICAGOのデータベース・ロールで考えられる組合せは3つあります。次の表で、その組合せを説明し、推定される原因と、それぞれの状況に対する高レベルの修正処置を示します。特定のエラー状況の詳細は、*Oracle Data Guard*概要および管理のOracle Data Guardのトラブルシューティングを参照してください。

V\$DATABASEの

DATABASE_ROLE列の値

原因と修正処置

BOSTON データベースがプライマリ、
CHICAGO データベースがスタンバイ

原因: BOSTON データベースがスタンバイ・データベース・ロールへの変換に失敗しました。

処置: BOSTON のスタンバイ・ロールへの切替えを妨げているエラーの詳細についてアラート・ログを参照し、エラーを解決するために必要な処置をとり、必要であれば BOSTON のいずれかのノードを再度オープンし、スイッチオーバー・プロセスをステップ 1 から繰り返します。

BOSTON データベースがスタンバイ、
CHICAGO データベースもスタンバイ

原因: CHICAGO データベースがプライマリ・データベース・ロールへの変換に失敗しました。

処置: 次の SQL 文を発行して、BOSTON または CHICAGO のいずれかをプライマリ・データベースに変換します。

```
SQL> ALTER DATABASE SWITCHOVER TO target_db_name FORCE;
```

次に例を示します。

- CHICAGO データベースで、次の SQL 文を発行し、プライマリ・データベースに変換します。

```
ALTER DATABASE SWITCHOVER TO CHICAGO FORCE;
```

- BOSTON データベースで、次の SQL 文を発行し、プライマリ・データベースに変換します。

```
ALTER DATABASE SWITCHOVER TO BOSTON FORCE;
```

SQL 文が ORA-16473 エラーで失敗した場合、コマンドを再度発行する前に REDO Apply を開始する必要があります。

V\$DATABASEの

DATABASE_ROLE列の値

原因と修正処置

次のように REDO Apply を再開します。

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE  
DISCONNECT;
```

次のように、スイッチオーバー・コマンドを再度発行します。

```
SQL> ALTER DATABASE SWITCHOVER TO BOSTON FORCE; Database  
altered.
```

BOSTON データベースがスタンバイ、
CHICAGO データベースがプライマリ

原因: BOSTON および CHICAGO データベースは新しいロールに正常に切り
替わりましたが、最終的な成功のステータスの BOSTON への送信時にエ
ラーが発生しました。

処置: ステップ 3 を続行してスイッチオーバー操作を終了します。

3. 次のSQL文を新しいプライマリ・データベースCHICAGOで発行して、開きます。

```
SQL> ALTER DATABASE OPEN;
```

4. 次のSQL文を発行して、新しいフィジカル・スタンバイ・データベースBOSTONをマウントします:

```
SQL> STARTUP MOUNT;
```

または、BOSTONがOracle Active Data Guardのフィジカル・スタンバイ・データベースである場合、次のSQL文を発行して読取り専用で開きます。

```
SQL> STARTUP;
```

5. 新しいフィジカル・スタンバイ・データベースでREDO Applyを開始します。次に例を示します。

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
```

親トピック: [フィジカル・スタンバイOracle Databaseでのリフレッシュおよびスイッチオーバー](#)

7 Oracle Databaseをアップグレードする前の互換性の確認

Oracle Databaseをアップグレードする前に、Oracle Databaseサーバーのアップグレードの互換性マトリクスを確認します。

- [Oracle Databaseの互換性レベルの確認](#)
このSQL問合せを使用して、データベースの互換性レベルが**COMPATIBLE**初期化パラメータの値に対応しているかどうかを確認します。
- [Oracle DatabaseのCOMPATIBLE初期化パラメータの値](#)
参照して、**COMPATIBLE**初期化パラメータのデフォルト値、最小値および最大値を見つけてください。

Oracle Databaseの互換性レベルの確認

このSQL問合せを使用して、データベースの互換性レベルが**COMPATIBLE**初期化パラメータの値に対応しているかどうかを確認します。

```
SQL> SELECT name, value FROM v$parameter  
        WHERE name = 'compatible';
```

親トピック: [Oracle Databaseをアップグレードする前の互換性の確認](#)

Oracle DatabaseのCOMPATIBLE初期化パラメータの値

参照して、**COMPATIBLE**初期化パラメータのデフォルト値、最小値および最大値を見つけてください。

次の表に、Oracle Database 19cおよびOracle Database 19cへのアップグレードがサポートされている各リリースでの**COMPATIBLE**のデフォルト値、最小値および最大値を示します。

表7-1 COMPATIBLE初期化パラメータ

Oracle Databaseのリリース	デフォルト値	最小値	最大値
Oracle Database 19c	19.0.0	11.2.0	CDB または非 CDB インスタンスの場合、RU または RUR の COMPATIBLE パラメータを変更しないでください。
Oracle Database 12c リリース 2 (12.2)	12.2.0	11.2.0	12.2.0
Oracle Database 12c リリース 1(12.1)	12.0.0	11.0.0	12.1.0
Oracle Database 11g リリース 2(11.2)	11.2.0	10.0.0	11.2.0

親トピック: [Oracle Databaseをアップグレードする前の互換性の確認](#)

8 Oracle Databaseのアップグレードの準備

データベースをアップグレードする前に、新機能を確認し、使用する最適なアップグレード・パスおよび方法を決定して、データベースのアップグレードを準備する手順を実行することをお勧めします。特に、アップグレード処理をテストし、バックアップ計画を準備することをお勧めします。

- [新しいOracleホームへのOracleソフトウェアのインストール](#)
Oracleホームの新しい場所を選択してから、単一インスタンス用の新しいOracle Databaseソフトウェアをインストールします。
- [Oracle Databaseのアップグレードのグラフィカル・ユーザー・インターフェースのメソッド](#)
Database Upgrade Assistant (DBUA)は、対話形式でアップグレード処理のステップを実行し、新しいOracle Databaseリリースのデータベースを構成します。
- [手動、コマンドラインでOracle Databaseをアップグレードする方法](#)
手動アップグレードでは、アップグレード・プロセスに対してよりきめ細やかな制御が可能です。
- [Oracle Databaseをアップグレードする前のバックアップ計画の準備](#)
アップグレードが確実に成功するように、適切なバックアップ計画を設計して実行する必要があります。
- [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)
Oracle Databaseのアップグレードを開始する前に必ずこれらのデータベースの準備作業を完了しておきます。
- [Oracle Databaseのアップグレード前情報ツールの使用](#)
アップグレード前情報ツール(preupgrade.jar)を理解および使用するには、これらのトピックを参照してください。
- [Oracle Databaseのアップグレード後のOracle Database Vaultの有効化](#)
ターゲットのデータベース・リリースによっては、Oracle Database Vaultを無効にしないとOracle Databaseのアップグレードを完了できない場合があります。

新しいOracleホームへのOracleソフトウェアのインストール

Oracleホームの新しい場所を選択した後、単一インスタンス用の新しいOracle Databaseソフトウェアをインストールします。

- [アップグレード時のOracleホームの新しい場所の選択](#)
新しいリリースのOracle Databaseに、現行のリリースのOracleホームとは別のOracleホームの場所を選択する必要があります。
- [単一インスタンス用の新しいOracle Databaseソフトウェアのインストール](#)
この手順の概要を使用すると、単一インスタンス・デプロイメント用の新しいOracle Databaseリリースのソフトウェアをインストールする際に役立ちます。

親トピック: [Oracle Databaseのアップグレードの準備](#)

アップグレード時のOracleホームの新しい場所の選択

新しいリリースのOracle Databaseに、現行のリリースのOracleホームとは別のOracleホームの場所を選択する必要があります。

別のインストール場所を使用すると、新しいOracleソフトウェアとともに、既存のOracleソフトウェアをインストールしたままにできます。個別のインストール場所を使用することで、本番環境全体を置き換える前にテスト・データベース上でアップグレード処理をテストできます。

データベースをアップグレードする際、データベースが非CDBかCDBかにかかわらず、新しいOracleホームをインストールするための新しい場所が必要です。

切断/接続アップグレードを使用してPDBをアップグレードする場合、PDBを接続するターゲットCDBがPDBの場所になります。ターゲットCDBにはすでにOracleホームが存在するため、ターゲットOracleホームをインストールするための新しい場所を選択する必要はありません。

親トピック: [新しいOracleホームへのOracleソフトウェアのインストール](#)

単一インスタンス用の新しいOracle Databaseソフトウェアのインストール

この手順の概要を使用すると、単一インスタンス・デプロイメント用の新しいOracle Databaseリリースのソフトウェアをインストールする際に役立ちます。

注意:



ソースおよびターゲットの Oracle ホームを異なるユーザーが所有している場合は、Database Upgrade Assistant (DBUA)を使用してデータベースをアップグレードできません。これを行うと、エラーPRKH-1014 が返されます。ソース・データベースとターゲット・データベースの所有者が同じであることを確認するか、手動アップグレードを実行してください。

このリリースの新しいOracle Databaseソフトウェアをインストールするには、次の手順を実行します。

1. Oracle Databaseソフトウェアのインストールを準備するには、Oracleオペレーティング・システム固有のドキュメントの指示に従ってください。
2. Oracle Universal Installerを起動して、ソフトウェアのみのインストールを選択します。

Oracle Databaseソフトウェアのインストールが正常に完了したら、「終了」ボタンをクリックしてOracle Universal Installerを閉じます。

3. アップグレード前情報ツールを実行します。このツールを使用すると、DBUAによってチェックされるアイテムのタイプを確認できます。ツールによって問題が識別され、検出された問題の修正に役立てることができます。

デフォルトで、このツールはNew_release_Oracle_home/rdbms/admin/preupgrade.jarの場所にあります。

Oracle Label Security、Oracle Database Vault、またはこれらの両方を使用する場合、データベース・エディションの選択ページで「Enterprise Edition」を選択し、「オプションの選択」をクリックして、コンポーネント・リストから一方または両方のコンポーネントを有効にします。

親トピック: [新しいOracleホームへのOracleソフトウェアのインストール](#)

Oracle Databaseのアップグレードのグラフィカル・ユーザー・インタフェースのメソッド

Database Upgrade Assistant (DBUA)は、対話形式でアップグレード処理のステップを実行し、新しいOracle Databaseリリースのデータベースを構成します。

DBUAは、一部の構成設定をアップグレードに必要な値に修正するアップグレード前情報ツールを起動します。たとえば、このツールは、初期化パラメータをアップグレードに必要な値に変更できます。このツールにより、アップグレードを続行する前に手動で修正可能な項目のリストも提供されます。

親トピック: [Oracle Databaseのアップグレードの準備](#)

手動、コマンドラインでOracle Databaseをアップグレードする方法

手動アップグレードでは、アップグレード・プロセスに対してよりきめ細やかな制御が可能です。

手動アップグレードでは、コマンドラインからSQLスクリプトおよびユーティリティを実行して、データベースを新しいOracle Databaseリリースへアップグレードします。

アップグレードの前に

- アップグレード前情報ツールを使用して、データベースを分析します。

アップグレード前情報ツールは、Oracle Databaseに付属するJava JARファイルです。ツールを起動すると自己解凍し、SQLスクリプトが実行されます。

アップグレード前情報ツールでは、データベースで発生する可能性のあるアップグレードの問題に関する警告が表示され、一部の問題に対処するために使用する修正スクリプトも生成されます。また、新しいリリースのOracle Databaseで必要な初期化パラメータの情報も表示されます。

- 新しいOracleホームを準備します。
- データベースのバックアップを実行します。

アップグレードするOracle Databaseのリリースによっては、さらにアップグレード前のステップを実行する必要がある場合もあります。こうしたステップにはアップグレードのパラメータ・ファイルの調整、サポート対象外となった初期化パラメータの削除、アップグレードで問題の原因となる可能性のある初期化パラメータの調整などがあります。

親トピック: [Oracle Databaseのアップグレードの準備](#)

Oracle Databaseをアップグレードする前のバックアップ計画の準備

アップグレードを確実に成功させるため、適切なバックアップ計画を設計して実行する必要があります。

バックアップ戦略を展開するには、次のような問題について考慮する必要があります。

- 業務上、本番データベースの実行不可能状態の許容範囲がどの程度の期間か。
- 可用性要件を満たすには、どのバックアップ戦略が必要か。
- サイトから離れた安全な場所にバックアップをアーカイブする必要があるか。
- 正しくバックアップされたことを確認するためのテストを実行するか。
- どのくらいの時間でバックアップをリストアできるか(オフサイト記憶域でのバックアップを含む)。
- 障害リカバリ手順は正常にテストされているか。

バックアップ計画は、これらの問題のすべてに答え、データベースを正常にバックアップおよびリカバリするための手順を備えている必要があります。RMANを使用したバックアップ計画の実装の詳細は、*Oracle Database*バックアップおよびリカバリ・ユーザーズ・ガイドを参照してください。

関連トピック

- [『Oracle Databaseバックアップおよびリカバリ・アドバンスド・ユーザーズ・ガイド』](#)

親トピック: [Oracle Databaseのアップグレードの準備](#)

Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業

Oracle Databaseのアップグレードを開始する前に、これらのデータベースの準備作業を完了していることを確認してください。

- [Oracle Databaseをアップグレードする場合のパッチ・セットの更新および要件](#)
アップグレードを開始する前に、新しいリリースのOracle Databaseを最新のOracleバンドル・パッチ、パッチ・セット更新(BPまたはPSU)、リリース更新(Update)またはリリース更新のリビジョン(Revision)に更新します。
- [透過的暗号化Oracleウォレットのコピー](#)
透過的データ暗号化(TDE)でOracleウォレットを使用する場合は、sqlnet. oraおよびウォレット・ファイルを新しいOracleホームにコピーします。
- [Oracle Databaseのアップグレード時のOracle Net Servicesに関する推奨事項](#)
アップグレード前にOracle Net Servicesに関するこれらの手順およびパラメータ変更を参照してください。
- [パスワードの大/小文字の区別とアップグレードについて](#)
デフォルトで、Oracle Database 12cリリース2 (12.2)以降のリリースは、排他モードにアップグレードされます。排他モードは大/小文字を区別しないパスワードを基にした認証をサポートしません。
- [大/小文字を区別しないパスワード・バージョンを使用しているアカウントがあるかどうかの確認](#)
次の手順を使用して、アップグレードするOracle Databaseに大/小文字を区別しないパスワード・バージョンを使用しているアカウントや構成パラメータがないか、確認します。
- [読取り専用の表領域を使用したアップグレードの実行](#)
アップグレード中にスキーマベースの表領域をオフラインに設定するには、-Tオプションを指定してパラレル・アップグレード・ユーティリティを使用します。

親トピック: [Oracle Databaseのアップグレードの準備](#)

Oracle Databaseをアップグレードする場合のパッチ・セットの更新および要件

アップグレードを開始する前に、新しいリリースのOracle Databaseを最新のOracleバンドル・パッチ、パッチ・セット更新(BPまたはPSU)、リリース更新(Update)またはリリース更新のレビジョン(Revision)に更新します。

新しいOracle Databaseリリースのソフトウェアには、リリースの時点で最新のOracle Databaseのすべてのパッチおよび更新を含む完全なリリースが含まれます。

アップグレードまたはダウングレード・プロセスを開始する前に、以前のOracle Databaseリリースと新しいOracle Databaseリリースの両方を更新することをお勧めします。Oracle Database 12c以前のリリースの場合は、最新のOracleバンドル・パッチまたはパッチ・セット更新(BPまたはPSU)に更新します。Oracle Database 12cリリース2 (12.2)、Oracle Database 18cまたはそれ以降のリリースの場合、最新の四半期のリリース更新(Update)またはリリース更新のレビジョン(Revision)に更新します。

My Oracle Supportで、最新パッチの入手方法に加え、ライフサイクル管理と自動パッチ適用に使用するツールについて詳細に説明したノートが提供されています。次に例を示します。

- My Oracle Supportノート854428.1には、パッチ・セットと更新に関する情報が記載されています。
- My Oracle Supportノート730365.1には、使用可能なOracle Databaseのほとんどのリリースのアップグレードの参照リスト(ダウンロード情報、パッチ番号、他のノートへのリンクなど)が記載されています。
- My Oracle Supportノート2180188.1にはアップグレード用、ダウングレード用、および以前のリリースとの共存用の単発のパッチが含まれています。
- My Oracle Supportノート1227443.1には、Oracle DatabaseのPSU/BP/Update/Revisionに関する既知の問題のリストが含まれています。このノートでは、Oracle Database、Oracle Grid InfrastructureおよびOracle JavaVMコンポーネント(OJVM)のすべての既知の問題ノートに関する情報を提供します。
- My Oracle Supportノート2118136.2には、環境に必要な更新、レビジョン、パッチ・セット更新(PSU)、SPU (CPU)、バンドル・パッチ、パッチ・セットおよびベース・リリースの選択に役立つダウンロード・アシスタントが含まれています。ここで開始することをお勧めします。

関連トピック

- [My Oracle Supportノート854428.1](#)
- [My Oracle Supportノート730365.1](#)
- [My Oracle Supportノート2180188.1](#)
- [My Oracle Supportノート1227443.1](#)
- [My Oracle Supportノート2118136.2](#)

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

透過的暗号化Oracleウォレットのコピー

透過的データ暗号化(TDE)でOracleウォレットを使用する場合は、sqlnet.oraおよびウォレット・ファイルを新しいOracleホームにコピーします。

アップグレードを開始する前に、sqlnet.oraおよびウォレット・ファイルをコピーする必要があります。

1. 許可ユーザーとしてログインします。
2. sqlnet.oraファイル、そしてウォレット・ファイルwallet.p12を新しいリリースのOracleホームに手動でコピーします。
3. マウントされているOracleウォレットをオープンします。

次に例を示します。

```
SQL> STARTUP MOUNT;  
SQL> ALTER SYSTEM SET ENCRYPTION WALLET OPEN
```

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

Oracle Databaseのアップグレード時のOracle Net Servicesに関する推奨事項

アップグレード前にOracle Net Servicesに関するこれらの手順およびパラメータ変更を参照してください。

Oracle Database 12cでは、新しい基礎となるネット・サービス・パラメータによってデータ圧縮が可能で、これにより、SQL TCP接続を介して転送されるセッション・データ・ユニットのサイズが縮小されます。

次のsqlnet.oraファイルの新しいパラメータで、圧縮、および優先圧縮方式を指定します。

- SQLNET.COMPRESSION
- SQLNET.COMPRESSION_LEVELS
- SQLNET.COMPRESSION_THRESHOLD

Oracle Database 12cで導入されたこれらのパラメータは、以前のリリースではサポートされていません。これらはOracle Database 12c以降のリリースでのみ使用できます。これらのsqlnet.ora圧縮パラメータの詳細は、*Oracle Net Services* リファレンスを参照してください。

アップグレードするOracle Databaseでリスナーが構成されていない場合、DBUAを実行する前に、Oracle Net Configuration Assistant (NETCA)を実行して、listener.oraファイルを含むOracle Databaseの新しいリリースのリスニング・プロトコルのアドレスおよびサービス情報を構成する必要があります。リリース11.2より前のリリースのOracle Databaseには、新しいバージョンのリスナーを作成する必要があります。現在のリスナーには、以前のOracle Databaseリリースとの下位互換性があります。

DBUAを使用してOracle RACデータベースをアップグレードする場合、リスナーが古いOracleホームから新しいOracle Grid Infrastructureホームに自動的に移行されます。Oracle Grid Infrastructureホームでlsnrctlコマンドを使用してリスナーを管理する必要があります。以前のリリースのOracleホームの場所からlsnrctlコマンドを使用しないでください。

関連トピック

- [『Oracle Database Net Servicesリファレンス・ガイド』](#)

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

パスワードの大/小文字の区別とアップグレードについて

デフォルトで、Oracle Database 12cリリース2 (12.2)以降のリリースは、排他モードにアップグレードされます。排他モードは大/小文字を区別しないパスワードを基にした認証をサポートしません。

サーバーが排他モードで実行されている場合、10Gパスワード・バージョンだけを持つアカウントにはアクセスできなくなります。

以前のリリースのOracle Databaseでは、SEC_CASE_SENSITIVE_LOGON=FALSEを設定して、大/小文字を区別しないパスワード・ベースの認証を許可するように認証プロトコルを構成できます。Oracle Database 12cリリース2 (12.2)以降、デフォルトのパスワード・ベースの認証プロトコル構成では、大/小文字を区別しない10Gパスワード・バージョンの使用は除外されます。デフォルトでは、SQLNET.ORAパラメータSQLNET.ALLOWED_LOGON_VERSION_SERVERは、排他モードである12に設定されています。データベースが排他モードで設定されている場合、パスワード・ベースの認証プロトコルでは大/小文字を区別するいずれかのパスワード・バージョン(11Gまたは12G)が、認証されるアカウントに存在する必要があります。このモードは、以前のリリースで使用されていた10Gパスワード・バージョンの使用を排除します。Oracle Database 12cリリース2以降にアップグレードすると、大/小文字を区別しない10Gパスワード・バージョンのみを含むアカウントにはアクセスできなくなります。この変更が発生するのは、サーバーがデフォルトで排他モードで実行されるためです。Oracle Databaseが排他モードで構成されていると、古い10Gパスワード・バージョンを使用してクライアントを認証できません。サーバーにはそのクライアントの認証に使用するパスワード・バージョンがありません。

セキュリティを高めるためには、大/小文字を区別するパスワード・ベースの認証を有効にしておくことをお勧めします。この設定がデフォルトです。ただし、新しいOracle Databaseリリースへのアップグレード中に、大/小文字を区別する認証を一時的に無効にすることはできます。アップグレード後に、パスワード・バージョンを管理するための実装計画の一環として、大/小文字を区別するパスワード・ベースの認証を有効にするかどうかを決定できます。

アップグレード前に、デフォルトのパスワード・ベースの認証プロトコル構成に対するこの変更により影響を受けるかどうかを判定することをお勧めします。次のことを確認してください。

- 10Gの大/小文字を区別しないパスワード認証バージョンだけを持つアカウントがあるかどうかを確認します。
- 重要なパッチ更新CPUOct2012またはそれ以降のパッチ更新を適用していないOracle Database 11gリリース2 (11.2.0.3)データベースがあるかどうか、かつ、大/小文字を区別しない10Gパスワード・バージョンを持たないアカウントがあるかどうかを確認します。
- FALSEに設定した非推奨のパラメータSEC_CASE_SENSITIVE_LOGONがないことを確認します。このパラメータをFALSEに設定すると、大/小文字を区別するパスワード・バージョン(11Gおよび12Gパスワード・バージョン)を認証に使用できません。

大/小文字を区別しないバージョンを使用するアカウントのオプション

大/小文字を区別しない10Gパスワード・バージョンのみを持つユーザー・アカウントがある場合は、次の代替方式のいずれかを選択する必要があります。

- アップグレード前に、10Gパスワード・バージョンだけを持つそれぞれのアカウントのパスワード・バージョンを更新します。パスワード・バージョンを更新するには、10Gパスワードを使用するユーザー・パスワードを期限切れにして、これらのユーザーにアカウントにログインするよう要求します。ログインが試行されると、サーバーはパスワード・バージョンのリストを自動的に更新しますが、これには大/小文字を区別するパスワード・バージョンが含まれます。
- SQLNET.ORAパラメータSQLNET.ALLOWED_LOGON_VERSION_SERVERの設定を、排他モード以外の設定に変更します。例: SQLNET.ALLOWED_LOGON_VERSION_SERVER=11

関連トピック

- [Oracle Database 2日でデータベース管理者](#)
- 『[Oracle Database Net Servicesリファレンス・ガイド](#)』
- 『[Oracle Databaseセキュリティ・ガイド](#)』

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

大/小文字を区別しないパスワード・バージョンを使用しているアカウントがあるかどうかの確認

次の手順を使用して、アップグレードするOracle Databaseに大/小文字を区別しないパスワード・バージョンを使用しているアカウントや構成パラメータがないか、確認します。

デフォルトで、Oracle Database 12cリリース2 (12.2)以降のリリースでは、10Gパスワード・バージョンは生成されることも許可されることもありません。

SQLNET.ALLOWED_LOGON_VERSION_SERVERを、大/小文字を区別しないバージョンを使用できる許容度の高い認証プロトコルに設定しておらず、かつ、大/小文字を区別しないパスワード・バージョンで認証されるユーザー・アカウントがデータベースでロックされないようにする場合は、該当するアカウントを確認して、それらが大/小文字を区別するパスワード・バージョンを使用していることを確認します。

例8-1 大/小文字を区別しない(10G)バージョンを使用するユーザー・アカウントの検索

管理ユーザーとしてSQL*Plusにログインし、次のSQL問合せを入力します。

```
SELECT USERNAME, PASSWORD_VERSIONS FROM DBA_USERS;
```

次の結果は、アカウントのパスワード・バージョンを示しています。

USERNAME	PASSWORD_VERSIONS
JONES	10G 11G 12C
ADAMS	10G 11G
CLARK	10G 11G
PRESTON	11G
BLAKE	10G

この例では、各ユーザー・アカウント・パスワードの使用中のバックグラウンド検証バージョンは異なります。

- JONESはOracle Database 10Gで作成されましたが、JONESのパスワードは、SQLNET.ALLOWED_LOGON_VERSION_SERVERパラメータの設定が8に設定されたときにOracle Database 12Cにリセットされました。その結果、このパスワードのリセットによって3つのバージョンがすべて作成されました。11Gおよび12Cは大/小文字を区別するパスワードを使用します。
- ADAMSおよびCLARKは最初に10Gバージョンで作成され、次に、以前のリリースからインポートされた後に11Gバージョンで作成されました。これらのアカウント・パスワードはその後、非推奨のパラメータSEC_CASE_SENSITIVE_LOGONをTRUEに設定した場合、11Gでリセットされました。
- BLAKEのパスワードは10Gバージョンで作成され、パスワードはリセットされませんでした。その結果、ユーザーBLAKEは引き続き10Gパスワード・バージョンを使用し、これは大/小文字を区別しないパスワードを使用します。

ユーザーBLAKEはアップグレード前には10Gパスワード・バージョンだけを持っています。

```
SQL> SELECT USERNAME, PASSWORD_VERSIONS FROM DBA_USERS;
```

```
USERNAME PASSWORD_VERSIONS
-----
BLAKE 10G
```

追加の実行を実行せずに新しいOracle Databaseリリースにアップグレードした場合、そのアカウントにはアクセスできなくなります。アップグレード前に、(SQLNET.ORAパラメータSQLNET.ALLOWED_LOGON_VERSION_SERVERをより許容度の高い認証モード

に設定して、)システムが排他モードで構成されていないことを確認してください。

例8-2 大/小文字を区別しないパスワードを持つアカウントの修正

次の手順を実行します。

1. 次のSQL問合せを使用して、10Gパスワード・バージョンだけを持つアカウントを検索します。

```
select USERNAME
  from DBA_USERS
 where ( PASSWORD_VERSIONS = '10G '
        or PASSWORD_VERSIONS = '10G HTTP ' )
    and USERNAME <> 'ANONYMOUS' ;
```

2. SQLNET.ORAパラメータSQLNET.ALLOWED_LOGON_VERSION_SERVERの設定を、該当するアカウントに適切なレベルに編集することにより、システムが排他モードで実行されるようにシステムを構成します。次に例を示します。

```
SQLNET.ALLOWED_LOGON_VERSION_SERVER=11
```

この変更を加えた後、アップグレードに進みます。

3. アップグレードが終了したら、次のコマンド構文を使用してステップ1で検出したアカウントを期限切れにします。usernameは、ステップ1の問合せで返されたユーザーの名前です。

```
ALTER USER username PASSWORD EXPIRE;
```

4. パスワードを期限切れにしたユーザーに、ログインするよう依頼します。
5. これらのユーザーがログインするときに、パスワードをリセットするよう、プロンプトが表示されます。システムは10Gパスワード・バージョンに加えて、欠落している11Gおよび12Cパスワード・バージョンを内部的に生成します。システムは許可モードで実行されているため、10Gパスワード・バージョンはそのまま存在します。
6. ユーザーが接続しているクライアント・ソフトウェアに05L_NP機能フラグがあることを確認します。

注意:



Oracle Database リリース 11.2.0.4 以降のすべてのクライアント、および Oracle Database リリース 12.1 以降のすべてのクライアントには 05L_NP 機能があります。他のクライアントでは CPU0ct2012 パッチを適用して 05L_NP 機能を取得する必要があります。

05L_NP 機能フラグは、*Oracle Database Net Services* リファレンスのパラメータ SQLNET.ALLOWED_LOGON_VERSION_SERVER の項に記載されています。

7. すべてのクライアントに05L_NP機能が導入されたら、次のプロシージャを使用してサーバー・セキュリティを元の排他モードに上げます。
 - a. インスタンス初期化ファイルからSEC_CASE_SENSITIVE_LOGON設定を削除するか、SEC_CASE_SENSITIVE_LOGONインスタンス初期化パラメータをTRUEに設定します。次に例を示します。

```
SEC_CASE_SENSITIVE_LOGON = TRUE
```
 - b. サーバーのSQLNET.ORAファイルからSQLNET.ALLOWED_LOGON_VERSION_SERVERパラメータを削除するか、サーバーのSQLNET.ORAファイルのSQLNET.ALLOWED_LOGON_VERSION_SERVERの値を12に戻す変更を加えることによって、排他モードに設定します。次に例を示します。

SQLNET.ALLOWED_LOGON_VERSION_SERVER = 12

8. 次のSQL問合せを使用して、まだ10Gパスワード・バージョンを持っているアカウントを検索します。

```
select USERNAME
  from DBA_USERS
 where PASSWORD_VERSIONS like '%10G%'
    and USERNAME <> 'ANONYMOUS' ;
```

9. ステップ8の問合せから戻されたアカウントのリストを使用して、まだ10Gパスワード・バージョンを持っているすべてのアカウントを期限切れにします。次の構文を使用して、アカウントを期限切れにします。usernameは問合せで返されたリストにある名前です。

```
ALTER USER username PASSWORD EXPIRE;
```

10. 期限切れにしたアカウントのユーザーに、アカウントにログインするよう依頼します。

ユーザーはログインするときに、パスワードを再設定するようプロンプトが表示されます。システムは、アカウントの11Gおよび12Cパスワード・バージョンだけを内部的に生成します。システムは排他モードで実行されているため、10Gパスワード・バージョンはもう生成されません。

11. ステップ1の問合せを実行して、システムがセキュア・モードで実行されていることを確認します。どのユーザーも検出されないことを確認します。この問合せでどのユーザーも検出されない場合、この結果は、10Gパスワード・バージョンがシステムにもう残っていないことを意味します。

例8-3 SEC_CASE_SENSITIVE_LOGON SetがFALSEになっているものがないことの確認

SQLNET.ALLOWED_LOGON_VERSION_SERVERパラメータが12または12aに設定されている場合、Oracle DatabaseはSEC_CASE_SENSITIVE_LOGONのFALSE設定の使用を妨げません。この設定により、アップグレードされたデータベースのすべてのアカウントがアクセス不能になる可能性があります。

```
SQL> SHOW PARAMETER SEC_CASE_SENSITIVE_LOGON
```

NAME	TYPE	VALUE
sec_case_sensitive_logon	boolean	FALSE

次のコマンドを使用して、このパラメータを変更できます。

```
SQL> ALTER SYSTEM SET SEC_CASE_SENSITIVE_LOGON = TRUE;
```

System altered.

注意:



パラメータ SQLNET.ALLOWED_LOGON_VERSION_SERVER の値が 12 よりも許容度の高いバージョン(11 など)に変更されないかぎり、SEC_CASE_SENSITIVE_LOGON パラメータを FALSE に設定しないでください。

関連トピック

- [『Oracle Database Net Servicesリファレンス・ガイド』](#)
- [『Oracle Databaseセキュリティ・ガイド』](#)

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

読取り専用の表領域を使用したアップグレードの実行

アップグレード中にスキーマベースの表領域をオフラインに設定するには、`-T`オプションを指定して`parallel_upgrade_utility`を使用します。

Oracle Databaseでは以前のリリースで作成されたファイル・ヘッダーを読み取ることができるため、アップグレード時にそれらに対して処理を行う必要はありません。READ ONLY表領域のファイル・ヘッダーは、それらがREAD WRITEに変更されたときに更新されます。

アップグレードで致命的なエラーが発生し、その結果アップグレードで表領域をオンラインに戻すことができなくなった場合、アップグレード・ログ・ファイルを参照してください。ログ・ファイルには、表領域を使用可能にするために必要な実際のSQL文が含まれています。表領域をオンラインに戻すには、データベースにログ・ファイルのSQL文を実行するか、PDBごとにログ・ファイルを実行する必要があります。

アップグレード・ログ・ファイルの表領域コマンドの表示

致命的なアップグレードの失敗が発生した場合は、ログ・ディレクトリ(`Oracle_base/cfgtoollogs/dbua`)に移動し、ログ・ファイルのコマンドを手動で実行して表領域を表示させることができます。表領域コマンドは次のログ・ファイルで表示できます。

- 非CDBのアップグレード: `catupgrd0.log`
- PDBデータベース: `catupgrdpdbname0.log` (`dbname`はアップグレードするPDBの名前です)。

各ログ・ファイルの先頭で、次のようなSQL文が見つかります。これは、表をREAD ONLYに設定する文です。

```
SQL> ALTER TABLESPACE ARGROTBLSPA6 READ ONLY;
```

```
Tablespace altered.
```

```
SQL> ALTER TABLESPACE ARGROTBLSPB6 READ ONLY;
```

```
Tablespace altered.
```

各ログ・ファイルの末尾近くには、表をREAD WRITEに再設定するSQL文が見つかります。

```
SQL> ALTER TABLESPACE ARGROTBLSPA6 READ WRITE;
```

```
Tablespace altered.
```

```
SQL> ALTER TABLESPACE ARGROTBLSPB6 READ WRITE;
```

```
Tablespace altered.
```

関連項目:

データベース間での表領域の転送の詳細は、[Oracle Database管理者ガイド](#)を参照してください。

親トピック: [Oracle Databaseのアップグレードを開始する前に完了するデータベースの準備作業](#)

Oracle Databaseのアップグレード前情報ツールの使用

アップグレード前情報ツール(preupgrade.jar)を理解および使用するには、これらのトピックを参照してください。

- [アップグレード前情報ツールの環境変数の設定](#)
アップグレード前情報ツールを実行する前に、非CDBのアップグレードの準備で、ツールを実行するOracleユーザーのユーザー環境変数を設定します。
- [アップグレード前情報ツールの実行](#)
システムおよびデータベースをアップグレードできるかどうかを確認するには、アップグレード前情報ツール(preupgrade.jar)を使用します
- [Oracle Databaseのアップグレードでのアップグレード前情報ツールの警告および推奨事項](#)
新しいリリースのOracle Databaseにアップグレードする前に、アップグレード前情報ツールの警告を分析します。ツールによってレポートされる項目ごとに、問題または警告の修正方法に関する情報が提供されます。
- [アップグレード前情報ツールの出力例](#)
この例では、アップグレード前情報ツールによって推奨される修正の表示方法がわかります(ただし、修正は自動的に実行されません)。

親トピック: [Oracle Databaseのアップグレードの準備](#)

アップグレード前情報ツールの環境変数の設定

アップグレード前情報ツールを実行する前に、非CDBのアップグレードの準備で、ツールを実行するOracleユーザーのユーザー環境変数を設定します。

アップグレード前情報ツールのユーザー環境変数を設定する必要があります。この例は、シェル・コマンドを使用してユーザー環境変数を設定し、以前のリリースのOracleホームをポイントする方法を示しています。

この例ではオペレーティング・システムがLinuxまたはUNIX、システムIDがsales01、以前のリリースのOracleホームのパスは/u01/app/oracle/product/12.1.0/dbhome_1です。

1. Oracleインストール所有者(oracle)としてログインします。
2. アップグレードする以前のリリースのOracleホームをポイントするように、ユーザー環境変数を設定します。

次に例を示します。

```
$ export ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
$ export ORACLE_BASE=/u01/app/oracle
$ export ORACLE_SID=sales01
$ export PATH=.:$ORACLE_HOME/bin:$PATH
```

親トピック: [Oracle Databaseのアップグレード前情報ツールの使用](#)

アップグレード前情報ツールの実行

システムおよびデータベースをアップグレードできるかどうかを確認するには、アップグレード前情報ツール(preupgrade.jar)を使用します

アップグレード前情報ツールは新しいリリースのOracleホームのファイル・パスである

ORACLE_HOME/rdbms/admin/preupgrade.jarにあります。Oracleは新しいOracle Databaseリリースに必要なシステム・チェックを付けてこれを構成してきました。ただし、ツールが実行するチェックは以前のリリースのOracle Databaseホームで実行されます。以前のリリースのOracleホームをポイントするように、Oracleユーザー環境関数を設定します。

以前のリリースのOracleホームではJavaバージョンを使用して、アップグレード前情報ツールを実行します。マルチテナント・アーキテクチャ(CDBおよびPDB)のアップグレードでは、ツールを実行する前に、ツールで分析するすべてのPDBをオープンします。

ユーザー・アカウントの環境変数を設定して、以前のリリースのORACLE_HOME、ORACLE_BASEおよびORACLE_SIDをポイントするようにします。

アップグレード前情報ツールの場所

preupgrade.jarファイルは新しいOracleホームにあります。

```
New_release_Oracle_home/rdbms/admin/preupgrade.jar
```

また、選択したパスにpreupgrade.jarバイナリをコピーできます。次に例を示します。

```
/tmp/preupgrade.jar
```

アップグレード前情報ツールの構文

```
$Earlier_release_Oracle_home/jdk/bin/java -jar $New_release_Oracle_home/rdbms/admin/preupgrade.jar [FILE|TERMINAL] [TEXT|XML] [DIR output_dir]
```

例8-4 ソースOracleホームの非CDBの例

1. 以前のリリースのOracleホームをポイントするようにユーザー環境変数を設定します。

```
$ export ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
$ export ORACLE_BASE=/u01/app/oracle
$ export ORACLE_SID=sales01
$ export PATH=.:$ORACLE_HOME/bin:$PATH
```

2. 以前のリリースのOracleホームに設定した環境設定を使用して、以前のリリースのOracle Databaseサーバーに新しいリリースのOracle Databaseのアップグレード前情報ツールを実行します。

```
$ORACLE_HOME/jdk/bin/java -jar
/u01/app/oracle/product/18.0.0/dbhome_1/rdbms/admin/preupgrade.jar TERMINAL TEXT
```

関連トピック

- [Oracle Databaseアップグレード・ガイド](#)

親トピック: [Oracle Databaseのアップグレード前情報ツールの使用](#)

Oracle Databaseのアップグレードでのアップグレード前情報ツールの警告および推奨事項

新しいリリースのOracle Databaseにアップグレードする前に、アップグレード前情報ツールの警告を分析します。ツールによってレポートされる項目ごとに、問題または警告の修正方法に関する情報が提供されます。

インストールされているデータベースのコンポーネントとスキーマの詳細は、My Oracle Supportノート472937.1を参照してください。NON VALIDステータスのコンポーネント診断の詳細は、My Oracle Supportノート753041.1を参照してください。

- [アクセス制御リストおよびネットワーク・ユーティリティ・パッケージの更新](#)
この手順を使用して、アクセス制御リスト(ACL)およびネットワーク・ユーティリティ・パッケージを更新します。
- [依存性の評価およびネットワーク・ユーティリティ・パッケージのACLの追加](#)
ネットワーク・ユーティリティ・パッケージの依存性についての警告を受信できます。依存性の評価および適切なアクセス制御リスト(ACL)の追加によるアクセスの提供を行うには、この手順を実行します。

関連トピック

- <https://support.oracle.com/rs?type=doc&id=472937.1>
- <https://support.oracle.com/rs?type=doc&id=753041.1>

親トピック: [Oracle Databaseのアップグレード前情報ツールの使用](#)

アクセス制御リストおよびネットワーク・ユーティリティ・パッケージの更新

この手順を使用して、アクセス制御リスト(ACL)およびネットワーク・ユーティリティ・パッケージを更新します。

Oracle Database 12c以降、UTLパッケージのアクセス制御はOracle Database Real Application Securityを使用して実装されています。UTLパッケージにはUTL_TCP、UTL_SMTP、UTL_MAIL、UTL_HTTPおよびUTL_INADDRが含まれています。アクセス制御にOracle XML DBは必要ありません。

1. ログイン中のユーザーに、DBMS_LDAP. initで指定されたホストおよびポートのconnect権限があることを確認します。DBMS_LDAP PL/SQLパッケージおよびHttpUriTypeタイプには新しい動作があります。この新しい動作のため、新しいOracle Databaseリリースにアップグレードした後にアクセス制御リスト(ACL)を作成、または更新する必要があります。たとえば、アプリケーションがDBMS_LDAPパッケージに依存している場合は、エラー「ORA-24247: アクセス制御リスト(ACL)によりネットワーク・アクセスが拒否されました」が発生する可能性があります。このエラーを防ぐには、ログイン中のユーザーは、DBMS_LDAP. initで指定されたホストおよびポートのconnect権限を持っている必要があります。
2. 次のパッケージのいずれかをインストールした場合、アップグレード後にこれらのパッケージを再インストールするように求められる可能性があります。
 - UTL_TCP
 - UTL_SMTP
 - UTL_MAIL
 - UTL_HTTP
 - UTL_INADDR

新しいOracle Databaseリリース用にこれらのパッケージが最新バージョンであることを確認します。

関連項目:

アクセス制御リストの構成の詳細は、『[Oracle Database Real Application Security管理者および開発者ガイド](#)』を参照してください

親トピック: [Oracle Databaseのアップグレードでのアップグレード前情報ツールの警告および推奨事項](#)

依存性の評価およびネットワーク・ユーティリティ・パッケージのACLの追加

ネットワーク・ユーティリティ・パッケージの依存性についての警告を受信できます。依存性の評価および適切なアクセス制御リスト(ACL)の追加によるアクセスの提供を行うには、この手順を実行します。

1. アップグレード前情報ツールを実行します。
2. 次の例のような警告メッセージがないか、アップグレード前情報ツール(preupgrade.jar)の出力を調べます。

```
WARNING: --> Database contains schemas with objects dependent on network packages.
.... Refer to the Database Upgrade Guide for instructions to configure Network ACLs.
.... USER WKSYS has dependent objects.
.... USER SYSMAN has dependent objects.
.... USER FLOWS_010600 has dependent objects.
```

3. ビューDBA_DEPENDENCIESを問い合せて、依存性の詳細を取得します。次に例を示します。

```
SELECT * FROM DBA_DEPENDENCIES
WHERE referenced_name IN (' UTL_TCP', ' UTL_SMTTP', ' UTL_MAIL', ' UTL_HTTP', ' UTL_INADDR', ' DBMS_LDAP' )
AND owner NOT IN (' SYS', ' PUBLIC', ' ORDPLUGINS' );
```

4. 新しいアクセス制御がアップグレード・テストの一部であることを確認するには、アップグレード後スクリプトを用意して、そのスクリプトをデータベース環境の変数にします。

パッケージのDBMS_NETWORK_ACL_ADMINを使用して、アクセス制御リスト(ACL)を更新します。更新したアクセス制御パッケージが前のリリース同様、機能できるように、このパッケージを使用して権限を作成し、割り当て、新しいアクセス制御に追加します。DBMS_NETWORK_ACL_ADMINを使用してアクセス制御リストを更新する方法については、『*Oracle Database Real Application Security*管理者および開発者ガイド』で提供されているスクリプト例を参照してください。

5. 必要な個別の権限は、アップグレード後に付与します。アクセス権は、元のデータベースでの使用方法に基づいて決定します。

関連トピック

- [『Oracle Database Real Application Security管理者および開発者ガイド』](#)

親トピック: [Oracle Databaseのアップグレードでのアップグレード前情報ツールの警告および推奨事項](#)

アップグレード前情報ツールの出力例

この例では、アップグレード前情報ツールによって推奨される修正の表示方法がわかります(ただし、修正は自動的に実行されません)。

修正スクリプトを実行する方法とタイミングは、ユーザーが制御できます。

次の例は、リリース12.2.0.1 CDBでOracle Database 18cのアップグレード前情報ツールを実行することで生成され、preupgrade.logに書き込まれる出力を示しています。

```
$ java -jar preupgrade.jar TEXT TERMINAL
Report generated by Oracle Database Pre-Upgrade Information Tool Version
18.0.0.0.0 on 2018-06-13T15:08:45
```

```
Upgrade-To version: 18.0.0.0.0
```

```
=====  
Status of the database prior to upgrade  
=====
```

```
Database Name: CDB2
Container Name: CDB$ROOT
Container ID: 1
Version: 12.2.0.1.0
Compatible: 12.2.0
Blocksize: 8192
Platform: Linux x86 64-bit
Timezone File: 26
Database log mode: NOARCHIVELOG
Readonly: FALSE
Edition: EE
```

Oracle Component	Upgrade Action	Current Status
Oracle Server	[to be upgraded]	VALID
JServer JAVA Virtual Machine	[to be upgraded]	VALID
Oracle XDK for Java	[to be upgraded]	VALID
Real Application Clusters	[to be upgraded]	OPTION OFF
Oracle Workspace Manager	[to be upgraded]	VALID
Oracle Label Security	[to be upgraded]	VALID
Oracle XML Database	[to be upgraded]	VALID
Oracle Java Packages	[to be upgraded]	VALID

```
=====  
BEFORE UPGRADE  
=====
```

```
REQUIRED ACTIONS  
=====
```

```
None
```

```
RECOMMENDED ACTIONS  
=====
```

1. Run 12.2.0.1.0 \$ORACLE_HOME/rdbms/admin/utlrp.sql to recompile invalid objects. You can view the individual invalid objects with

```
SET SERVEROUTPUT ON;  
EXECUTE DBMS_PREUP.INVALID_OBJECTS;
```

3 objects are INVALID.

There should be no INVALID objects in SYS/SYSTEM or user schemas before database upgrade.

2. Review and remove any unnecessary HIDDEN/UNDERSCORE parameters.

The database contains the following initialization parameters whose name begins with an underscore:

`_exclude_seed_cdb_view`

Remove hidden parameters before database upgrade unless your application vendors and/or Oracle Support state differently. Changes will need to be made in the pfile/spfile.

3. (AUTOFIXUP) Gather stale data dictionary statistics prior to database upgrade in off-peak time using:

```
EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

Dictionary statistics do not exist or are stale (not up-to-date).

Dictionary statistics help the Oracle optimizer find efficient SQL execution plans and are essential for proper upgrade timing. Oracle recommends gathering dictionary statistics in the last 24 hours before database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

INFORMATION ONLY

=====

4. To help you keep track of your tablespace allocations, the following AUTOEXTEND tablespaces are expected to successfully EXTEND during the upgrade process.

Tablespace	Size	Min Size For Upgrade
SYSAUX	550 MB	616 MB
SYSTEM	700 MB	1117 MB
TEMP	22 MB	150 MB
UNDOTBS1	315 MB	433 MB

Minimum tablespace sizes for upgrade are estimates.

5. No action needed.

Using default parallel upgrade options, this CDB with 1 PDBs will first upgrade the CDB\$ROOT, and then upgrade at most 1 PDBs at a time using 2 parallel processes per PDB.

The number of PDBs upgraded in parallel and the number of parallel processes per PDB can be adjusted as described in Database Upgrade Guide.

ORACLE GENERATED FIXUP SCRIPT

=====

All of the issues in database CDB2 container CDB\$ROOT which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by

executing the following from within the container

```
SQL>@/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/preupgrade_fixups.sql
```

```
=====
AFTER UPGRADE
=====
```

REQUIRED ACTIONS

```
=====
None
```

RECOMMENDED ACTIONS

- ```
=====
6. Upgrade the database time zone file using the DBMS_DST package.
```

The database is using time zone file version 26 and the target 18.0.0.0.0 release ships with time zone file version 31.

Oracle recommends upgrading to the desired (latest) version of the time zone file. For more information, refer to "Upgrading the Time Zone File and Timestamp with Time Zone Data" in the 18.0.0.0.0 Oracle Database Globalization Support Guide.

- ```
7. (AUTOFIXUP) Gather dictionary statistics after the upgrade using the
command:
```

```
EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

Oracle recommends gathering dictionary statistics after upgrade.

Dictionary statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. After a database upgrade, statistics need to be re-gathered as there can now be tables that have significantly changed during the upgrade or new tables that do not have statistics gathered yet.

- ```
8. Gather statistics on fixed objects after the upgrade and when there is a
representative workload on the system using the command:
```

```
EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
```

This recommendation is given for all preupgrade runs.

Fixed object statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. Those statistics are specific to the Oracle Database release that generates them, and can be stale upon database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

#### ORACLE GENERATED FIXUP SCRIPT

```
=====
All of the issues in database CDB2 container CDB$ROOT
which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
executing the following from within the container
```

```
SQL>@/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/postupgrade_fixups.sql
```

Upgrade-To version: 18.0.0.0.0

=====  
Status of the database prior to upgrade  
=====

Database Name: CDB2  
Container Name: PDB\$SEED  
Container ID: 2  
Version: 12.2.0.1.0  
Compatible: 12.2.0  
Blocksize: 8192  
Platform: Linux x86 64-bit  
Timezone File: 26  
Database log mode: NOARCHIVELOG  
Readonly: TRUE  
Edition: EE

| Oracle Component          | Upgrade Action   | Current Status |
|---------------------------|------------------|----------------|
| Oracle Server             | [to be upgraded] | VALID          |
| Real Application Clusters | [to be upgraded] | OPTION OFF     |
| Oracle Workspace Manager  | [to be upgraded] | VALID          |
| Oracle XML Database       | [to be upgraded] | VALID          |

=====  
BEFORE UPGRADE  
=====

REQUIRED ACTIONS  
=====

None

RECOMMENDED ACTIONS  
=====

1. Run 12.2.0.1.0 \$ORACLE\_HOME/rdbms/admin/utlrp.sql to recompile invalid objects. You can view the individual invalid objects with

```
SET SERVEROUTPUT ON;
EXECUTE DBMS_PREUP.INVALID_OBJECTS;
```

6 objects are INVALID.

There should be no INVALID objects in SYS/SYSTEM or user schemas before database upgrade.

2. Review and remove any unnecessary HIDDEN/UNDERSCORE parameters.

The database contains the following initialization parameters whose name begins with an underscore:

`_exclude_seed_cdb_view`

Remove hidden parameters before database upgrade unless your application vendors and/or Oracle Support state differently. Changes will need to be made in the pfile/spfile.

3. (AUTOFIXUP) Gather stale data dictionary statistics prior to database upgrade in off-peak time using:

```
EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

Dictionary statistics do not exist or are stale (not up-to-date).

Dictionary statistics help the Oracle optimizer find efficient SQL execution plans and are essential for proper upgrade timing. Oracle recommends gathering dictionary statistics in the last 24 hours before database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

4. (AUTOFIXUP) Gather statistics on fixed objects prior the upgrade.

None of the fixed object tables have had stats collected.

Gathering statistics on fixed objects, if none have been gathered yet, is recommended prior to upgrading.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

#### INFORMATION ONLY

5. To help you keep track of your tablespace allocations, the following AUTOEXTEND tablespaces are expected to successfully EXTEND during the upgrade process.

| Tablespace | Size   | Min Size<br>For Upgrade |
|------------|--------|-------------------------|
| SYSAUX     | 235 MB | 500 MB                  |
| SYSTEM     | 210 MB | 584 MB                  |
| TEMP       | 20 MB  | 150 MB                  |
| UNDOTBS1   | 210 MB | 412 MB                  |

Minimum tablespace sizes for upgrade are estimates.

6. No action needed.

Using default parallel upgrade options, this CDB with 1 PDBs will first upgrade the CDB\$ROOT, and then upgrade at most 1 PDBs at a time using 2 parallel processes per PDB.

The number of PDBs upgraded in parallel and the number of parallel processes per PDB can be adjusted as described in Database Upgrade Guide.

#### ORACLE GENERATED FIXUP SCRIPT

All of the issues in database CDB2 container PDB\$SEED which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by executing the following from within the container

```
SQL>@/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/preupgrade_fixups.sql
```

=====

AFTER UPGRADE

=====

## REQUIRED ACTIONS

=====

None

## RECOMMENDED ACTIONS

=====

7. Upgrade the database time zone file using the DBMS\_DST package.

The database is using time zone file version 26 and the target 18.0.0.0.0 release ships with time zone file version 31.

Oracle recommends upgrading to the desired (latest) version of the time zone file. For more information, refer to "Upgrading the Time Zone File and Timestamp with Time Zone Data" in the 18.0.0.0.0 Oracle Database Globalization Support Guide.

8. (AUTOFIXUP) Gather dictionary statistics after the upgrade using the command:

```
EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

Oracle recommends gathering dictionary statistics after upgrade.

Dictionary statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. After a database upgrade, statistics need to be re-gathered as there can now be tables that have significantly changed during the upgrade or new tables that do not have statistics gathered yet.

9. Gather statistics on fixed objects after the upgrade and when there is a representative workload on the system using the command:

```
EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
```

This recommendation is given for all preupgrade runs.

Fixed object statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. Those statistics are specific to the Oracle Database release that generates them, and can be stale upon database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

## ORACLE GENERATED FIXUP SCRIPT

=====

All of the issues in database CDB2 container PDB\$SEED which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by executing the following from within the container

```
SQL>@/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/postupgrade_fixups.sql
```

=====

## PREUPGRADE SUMMARY

=====

```
/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/preupgrade.log
/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/preupgrade_fixups.sql
```

```
/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/postupgrade_fixups.sql
```

Execute fixup scripts across the entire CDB:

Before upgrade:

1. Execute preupgrade fixups with the below command

```
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB2/preupgrade/ -b preup_CDB2
/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/preupgrade_fixups.sql
```

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB2/preupgrade/

After the upgrade:

1. Execute postupgrade fixups with the below command

```
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB2/preupgrade/ -b postup_CDB2
/u01/app/oracle/cfgtoollogs/CDB2/preupgrade/postupgrade_fixups.sql
```

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB2/preupgrade/

Preupgrade complete: 2018-06-13T15:09:00

**親トピック:** [Oracle Databaseのアップグレード前情報ツールの使用](#)

# Oracle Databaseのアップグレード後のOracle Database Vaultの有効化

ターゲットのデータベース・リリースによっては、Oracle Database Vaultを無効にしないとOracle Databaseのアップグレードを完了できない場合があります。

- [Oracle Database Vaultの無効化なしでのOracle Databaseのアップグレード](#)  
ターゲットのOracle Databaseリリースが12.2以上である場合は、Oracle Database Vaultを無効にせずにアップグレードできます。
- [Oracle Database Vaultが関連する一般的なアップグレード・シナリオ](#)  
アップグレード後にOracle Database Vaultを有効にする必要があるかどうかは、ソースのOracle Databaseリリースによって異なります。

**親トピック:** [Oracle Databaseのアップグレードの準備](#)

# Oracle Database Vaultの無効化なしでのOracle Databaseのアップグレード

ターゲットのOracle Databaseリリースが12.2以上である場合は、Oracle Database Vaultを無効にせずにアップグレードできます。

ソースのOracle DatabaseリリースでOracle Database Vaultを有効にしていた場合は、Oracle Database Vaultを事前に無効にしなくても、Oracle DatabaseをOracle Database 18c以降のリリースにアップグレードできます。ソースのOracle DatabaseリリースがOracle Database 12cリリース1 (12.1)以上である場合は、アップグレード後に、アップグレード前に用意したものと同一強化設定を使用してOracle Database Vaultが有効になります。たとえば、ソース・データベースがOracle Databaseリリース12.1であり、そのリリースでOracle Database Vaultを無効にしていた場合は、アップグレード後も無効になります。アップグレード前にソースのOracle Databaseリリース12.1データベースでOracle Database Vaultを有効にしていた場合は、アップグレード後もOracle Database Vaultは有効になります。

アップグレード前に手動でOracle Database Vaultを無効にした場合は、アップグレード後に手動でOracle Database Vaultを有効にする必要があります。

アップグレード前にOracle Database Vaultを有効にしていなかった場合は、アップグレード後に手動で有効にできます。

プロシージャ `dvsys.dbms_macadm.enable_dv()` を使用して、アップグレードしたデータベースでOracle Database Vaultを有効にします。このプロシージャは、DV\_OWNERを付与されたユーザー・アカウントを使用して実行します。プロシージャを実行したら、そのプロシージャを有効にするためにデータベース・インスタンスを再起動します。

## 関連トピック

- [Oracle Database Vaultの管理者ガイド](#)

**親トピック:** [Oracle Databaseのアップグレード後のOracle Database Vaultの有効化](#)

# Oracle Database Vaultが関連する一般的なアップグレード・シナリオ

アップグレード後にOracle Database Vaultを有効にする必要があるかどうかは、ソースのOracle Databaseリリースによって異なります。

- Oracle Database 11gリリース2 (11.2)以下からのアップグレード: アップグレード後、Oracle Database Vaultはデフォルトで無効になります。
- Oracle Database 12cリリース1 (12.1)以上からのアップグレード: アップグレード後、Oracle Database Vaultはアップグレード前に用意したものと同一強化ステータスになります。

表8-1 Oracle Database Vaultの一般的なアップグレード・シナリオおよびアップグレード準備作業

| ソース・データベース・リリース   | ターゲット・データベース・リリース | アップグレード前に Database Vaultを無効にする必要があるかどうか | アップグレード後のDatabase Vaultのステータス                       |
|-------------------|-------------------|-----------------------------------------|-----------------------------------------------------|
| 11.2 以前           | 12.1              | はい                                      | 無効になっています。アップグレード後に Database Vaultを手動で有効にする必要があります。 |
| 11.2 以前           | 12.2、18.1 以降      | いいえ                                     | 無効になっています。アップグレード後に Database Vaultを手動で有効にする必要があります。 |
| 12.1、12.2、18.1 以降 | 12.2、18.1 以降      | いいえ                                     | Database Vaultはアップグレード前に用意したものと同一強化ステータスになります。      |

**親トピック:** [Oracle Databaseのアップグレード後のOracle Database Vaultの有効化](#)

# 9 Oracle Databaseのアップグレード

エンタープライズの複雑さに対応するため、データベースをアップグレードするための様々な方法が提供されています。

- [DBUAを使用したLinux、UNIXおよびWindowsシステムでのデータベースのアップグレード](#)  
DBUAのグラフィカル・ユーザー・インターフェースを使用してデータベースをアップグレードするには、新しいOracleホーム内からこれらのステップを実行します。
- [非CDBアーキテクチャのOracle Databaseの手動アップグレード](#)  
この手順では、非CDBアーキテクチャのOracle Databaseをアップグレードするためのステップを示します。

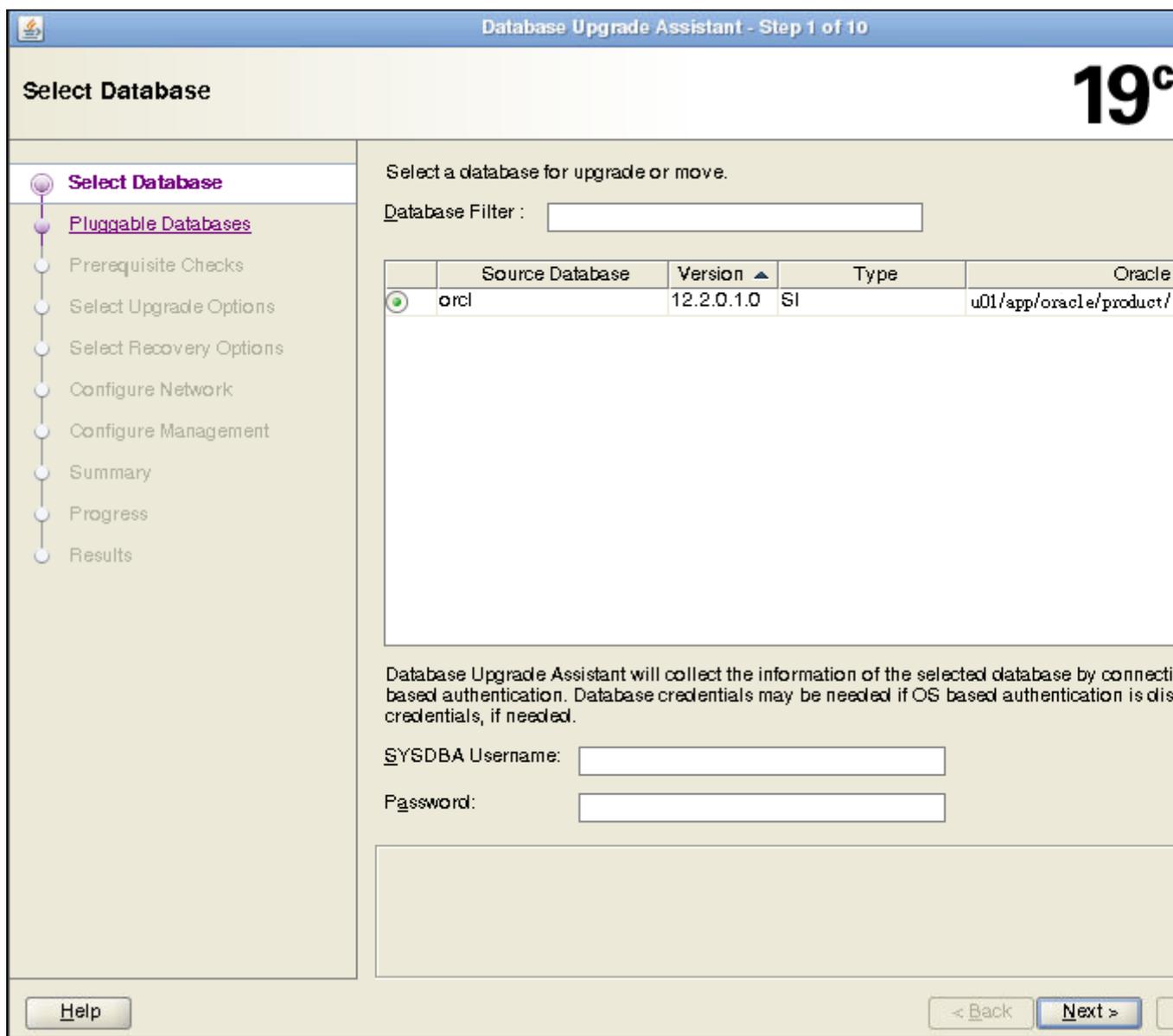
# DBUAを使用したLinux、UNIXおよびWindowsシステムでのデータベースのアップグレード

DBUAのグラフィカル・ユーザー・インターフェースを使用してデータベースをアップグレードするには、新しいOracleホーム内からこれらのステップを実行します。

Microsoft Windowsシステム(Windows)で、Oracle Database管理ユーザー(オペレーティング・システム割当てのORA\_DBAロールを持つユーザー)またはOracleインストール所有者アカウントとしてDBUAを実行します。

1. 新しいデータベース・ソフトウェアがインストールされたOracleホームからOracle Database Upgrade Assistant (DBUA)を起動します。dbua実行可能ファイルはディレクトリ・パスORACLE\_HOME/binにあります。
  - LinuxまたはUNIXプラットフォームの場合、SYSDBA権限を持つユーザーとしてログインし、Oracle Database 19cの新しいホームでシステム・プロンプトに次のコマンドを入力します。

```
./dbua
```
  - Windowsオペレーティング・システムの場合、「スタート」→「プログラム」→「Oracle HOME\_NAME」→「Configuration and Migration Tools」→「Database Upgrade Assistant」を選択します。
2. 「データベースの選択」ウィンドウが表示されます。以前のリリースのOracle Databaseがインストールされている場合は、これらがアップグレード可能としてリストされます。



いずれかのDBUAウィンドウのヘルプが必要な場合、またはDBUAの詳細を参照する場合は、「ヘルプ」をクリックしてオンライン・ヘルプを開きます。

必要に応じて、選択したデータベースのSYSDBAユーザー名とパスワードを入力します。

SYSDBA権限を所有していないユーザー・アカウントからDBUAを実行する場合、またはソース・データベースにオペレーティング・システム認証がない場合、ユーザー名およびパスワード資格証明を入力して、選択されたデータベースに対するSYSDBA権限を有効にする必要があります。使用するユーザー・アカウントにSYSDBA権限がある場合、またはオペレーティング・システム認証を使用した場合、ユーザー名とパスワードを入力する必要はありません。

選択を行った後に「次へ」をクリックします。

### 注意:

- 一度に1つのデータベースのみを選択できます。
- 単一インスタンスのアップグレードで、データベースがリストに表示されない場合、`/etc/oratab` にそのデータベースがそこにリストされていない場合は、DBUA に特定のデータベースをアップグレードするように指示
- 使用している単一インスタンス・データベースが`/etc/oratab` にリストされておらず、かつ、DBUA が

sid Oracle\_SID、-oracleHome Oracle\_home および sysDBAPassword password を使用し、  
ドするよう DBUA に明示的に指示できます。次に例を示します。

```
dbua -sid Oracle_SID -oracleHome /u01/app/oracle/18.1.0/dbhome1 -sysDBAUserName
```

- アカウントに SYSDBA 権限がない場合、またはオペレーティング・システム認証が設定されていない場合は、Database SID、username は SYSDBA 権限を持つユーザー名、password はユーザー名のパスワードで

```
dbua -sid mydb -oracleHome /u01/app/oracle/18.1.0/dbhome1 -sysDBAUserName - username -
```

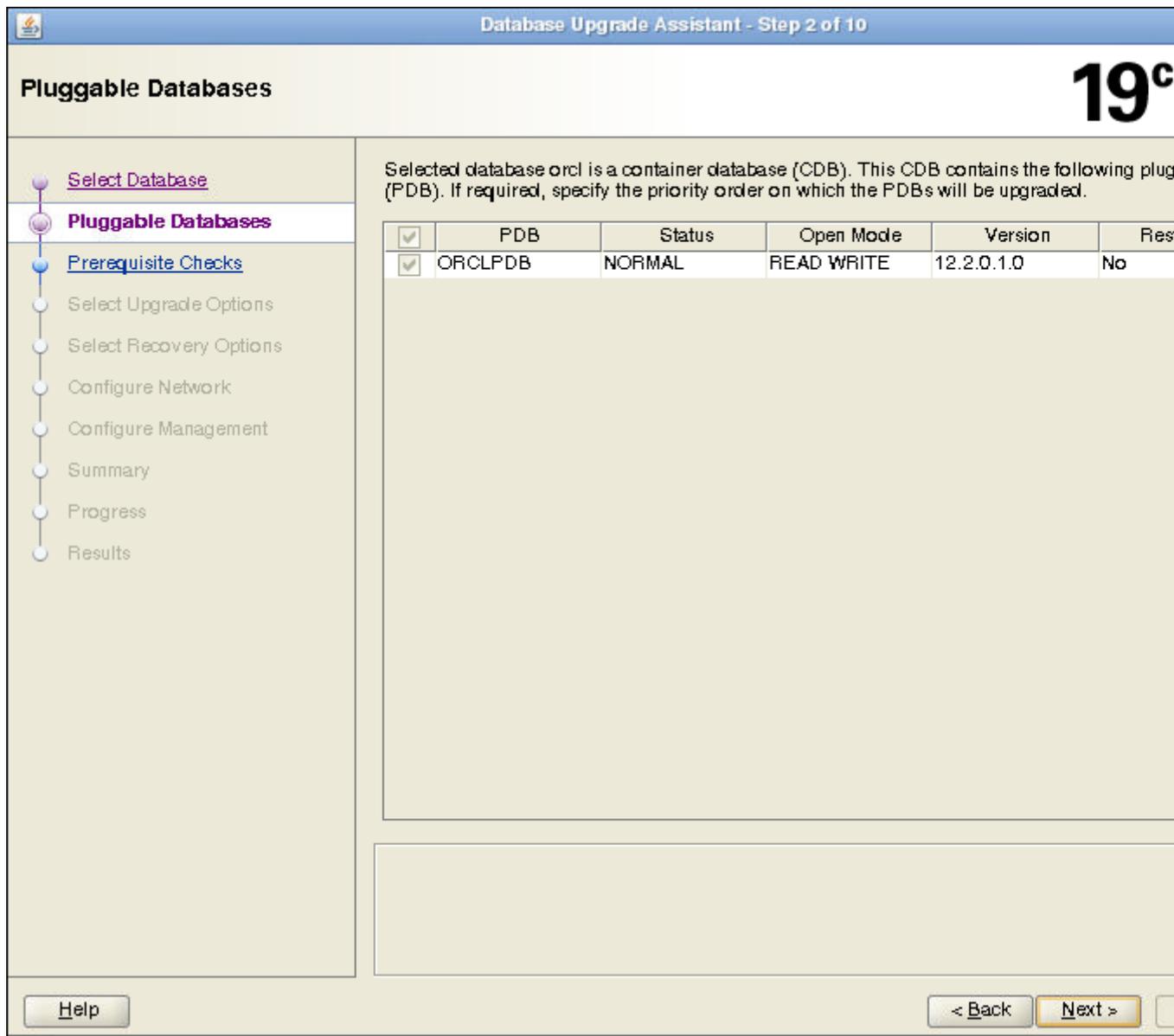
3. Oracle Real Application Clusters (Oracle RAC)のアップグレードで、データベースがリストに表示されない場合、次のcrsctlコマンドを入力してOracle RACインスタンスを確認してください。

```
crsctl status resource -t
```

次のコマンドを入力して特定のOracle RACデータベースを確認することもできます(db\_nameはOracle RACデータベースの名前)。

```
crsctl status resource ora.db_name.db
```

4. Microsoft Windowsでは、次のセキュリティ変更が認証とユーザー・アカウントに影響します。
  - セキュリティ上の理由から、NTLMプロトコルを使用したWindows NTS認証はサポートされなくなりました。Kerberos認証のみがサポートされている認証です。このリリースでは、NTSは、Windows NTドメインでも、Windows NTコントローラを使用するドメインでも動作しません。
  - Oracleは、Windows Local Systemアカウントではなく標準のMicrosoft Windowsユーザー・アカウントを使用してOracle Databaseサービスを実行します。Oracleインストール所有者のアカウント・アクセス権を制限することで、Microsoft Windowsのセキュリティが向上します。
5. 選択したデータベースがマルチテナント・コンテナ・データベース(CDB)の場合、DBUAによって「プラグブル・データベース」ウィンドウが表示されます。「プラグブル・データベース」ウィンドウにはCDBに含まれているプラグブル・データベースがリストされます。リストされたPDBは、選択されたCDBのアップグレードの一部としてアップグレードされます。

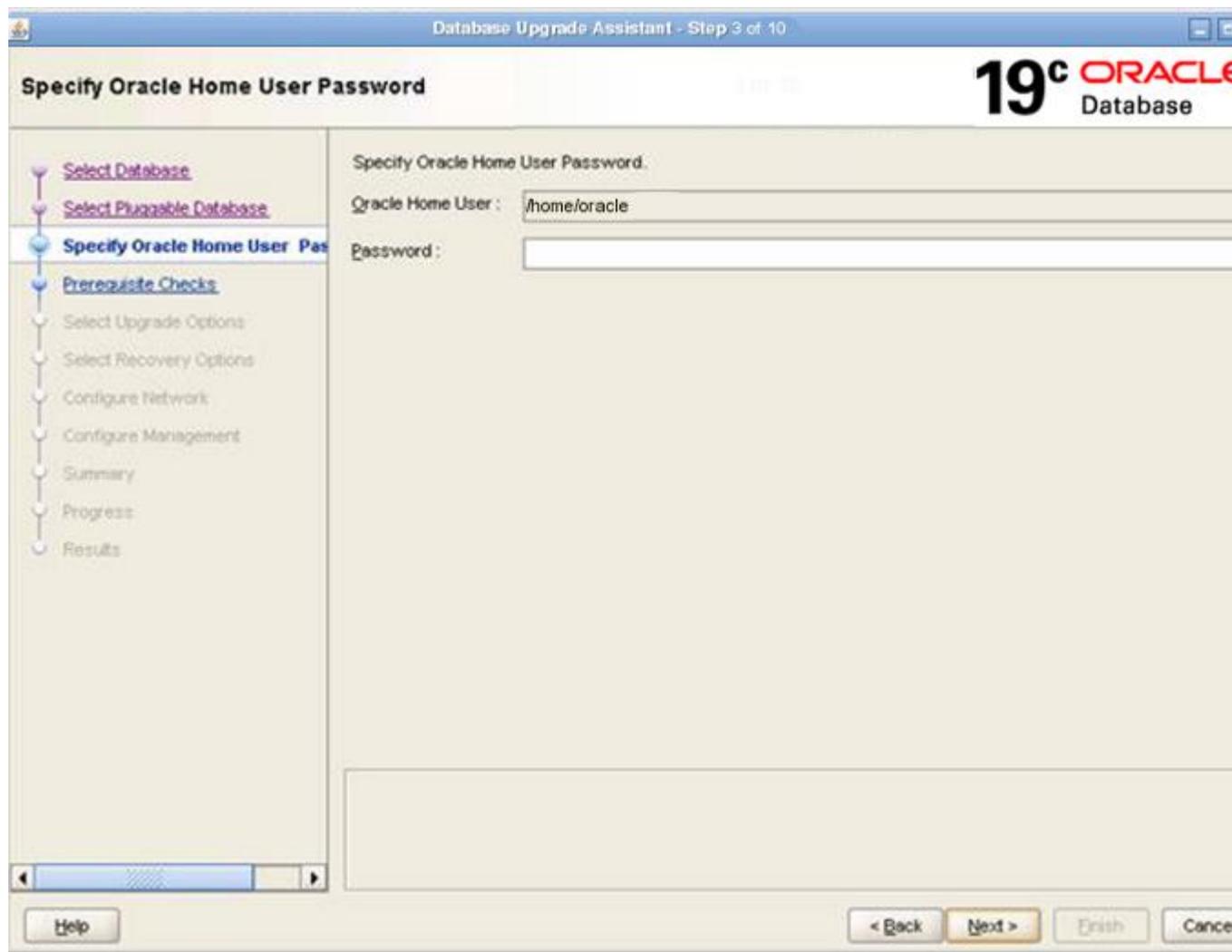


PDBのアップグレード優先度を選択できます。各PDBの優先度列をクリックし、アップグレード優先度の数値を入力します(1が最初のアップグレード、2が2番目のアップグレードというように続きます)。

デフォルトで、CDB\$ROOT、PDB\$SEEDおよびCDBに接続されているすべてのPDBがアップグレードされます。このときに一部のPDBをアップグレードしない場合は、それらのPDBを切断します。

PDBおよびアップグレード優先度の選択が完了したら、「次へ」をクリックします。

- Windowsプラットフォームでのみ、アップグレード・ターゲット・ホームがOracleホーム・ユーザーに関連付けられたセキュアなホームの場合は、「Oracleホームのユーザー・パスワードの指定」ウィンドウが開きます。他のプラットフォームでは、次のステップに進みます。



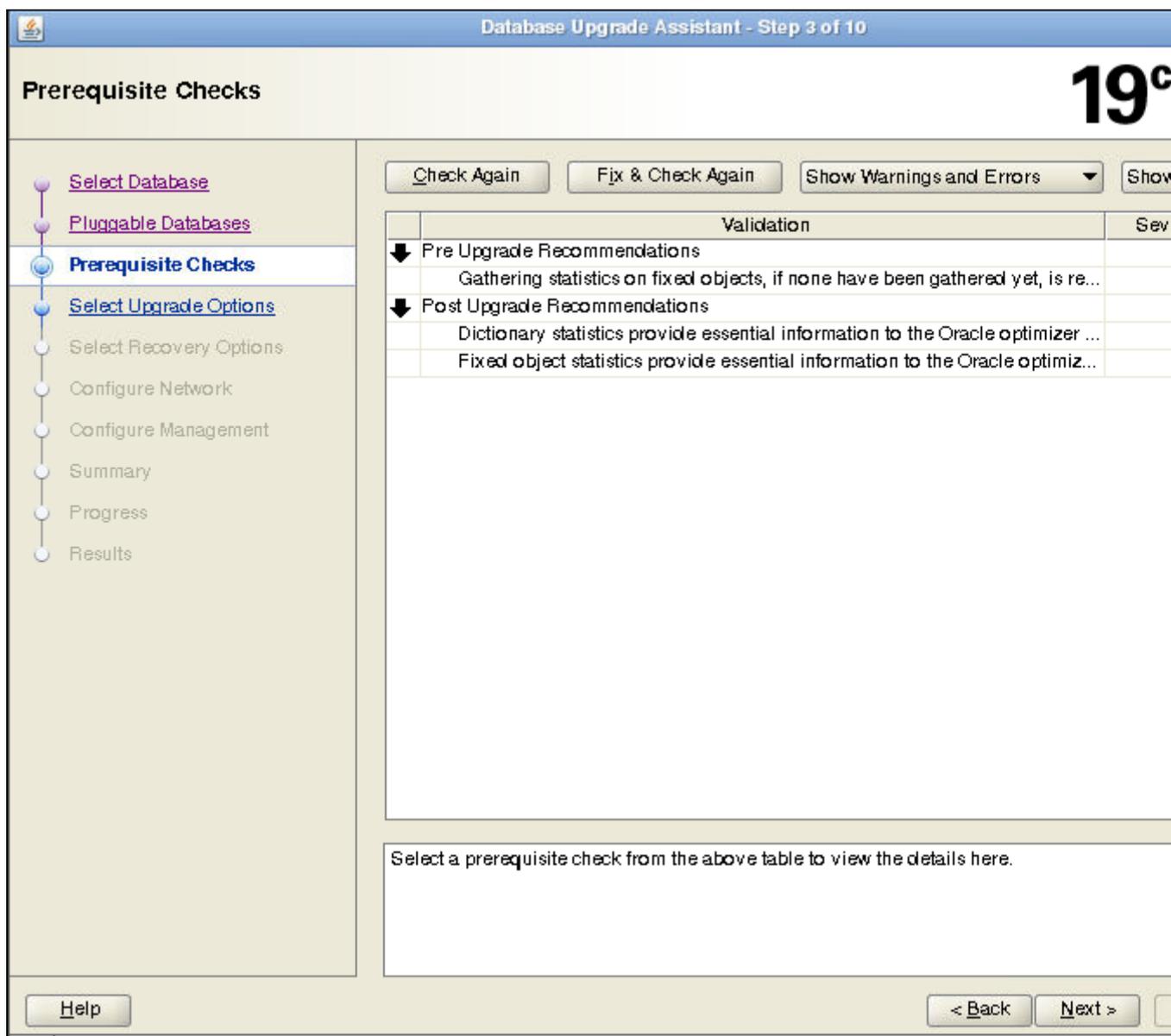
Oracleホームのユーザー名と、そのユーザー・アカウントのパスワードを指定して、「次へ」をクリックします。

7. 「前提条件チェック」ウィンドウが開きます。DBUAによって、データベースの分析およびアップグレード前のチェックが実行され、必要に応じて警告が表示されます。次のリストは、データベースに対するDBUAのチェックおよびDBUAが行う処置の例です。

- 空のデータベースのごみ箱。
- 無効なオブジェクトの識別。
- 非推奨となったかサポートが終了した初期化パラメータの識別。
- タイムゾーン・データファイルの識別。

分析完了までには数分かかります。

DBUAによる分析が終了すると、「前提条件チェック」ウィンドウが再び表示され、チェック結果が示されます。



「前提条件チェック」ウィンドウには、DBUAが完了したチェックと、検出されたエラーの重大度が表示されます。DBUAによってエラーが検出された場合、エラーが修正可能かどうかと、エラーを修正するために実行できる処置の内容が表示されます。

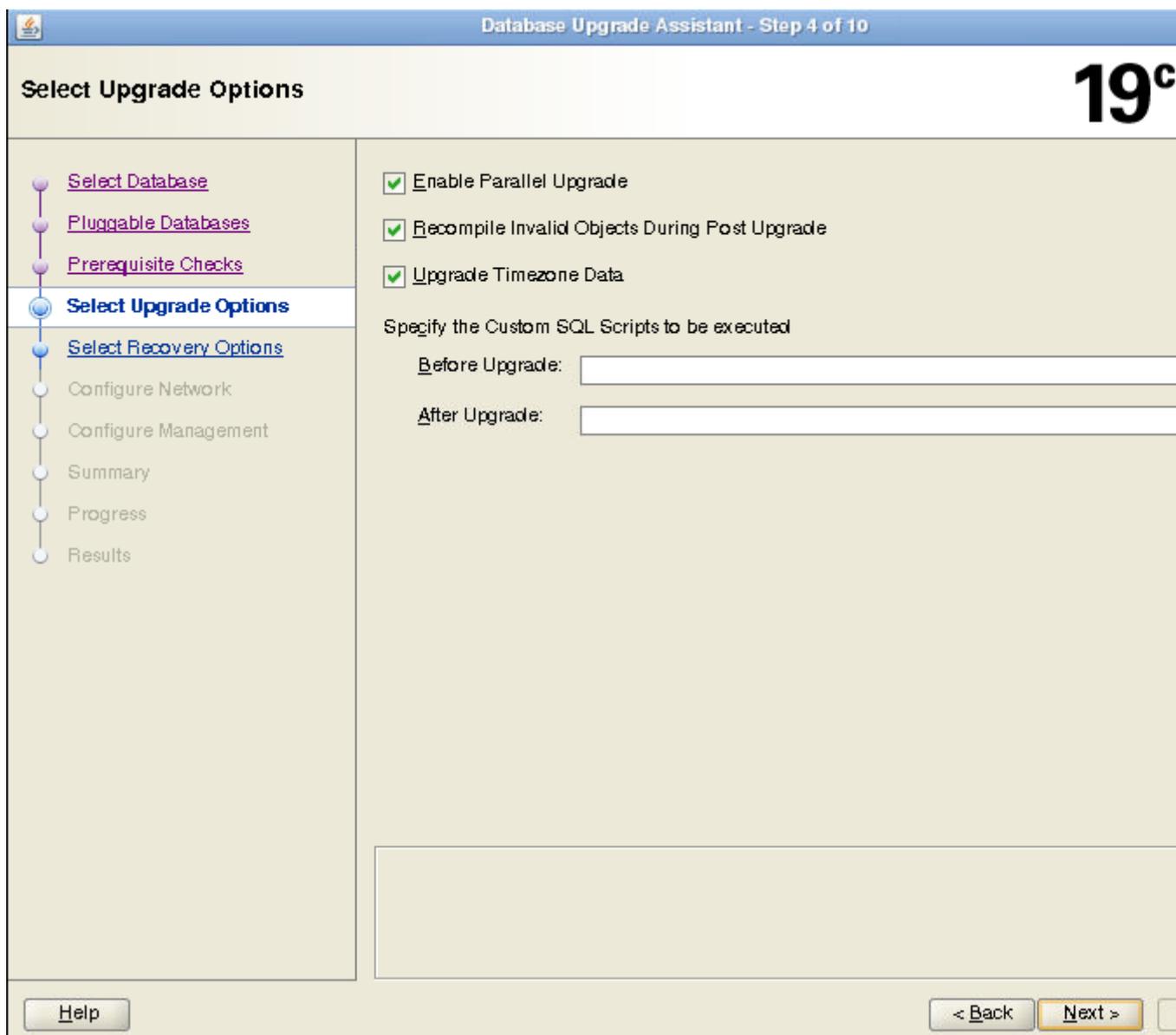
DBUAで修正できるエラーが表示されている場合、「修正および再チェック」を選択します。

DBUAで修正できないエラーが検出された場合、エラーの原因を手動で修正し、「再チェック」を選択します。

DBUAでエラーまたは警告が検出されない場合、このウィンドウは自動的に省略され、次のウィンドウに進みます。

検出されたエラーを修正したら、「次へ」をクリックします。

8. 「アップグレード・オプションの選択」ウィンドウが表示されます。



このウィンドウには次のオプションがあります。

#### 並列アップグレードの有効化

アップグレード処理中に平行実行を有効にするには、このオプションを選択します。アップグレード処理の平行実行は、スクリプトおよび処理の実行を同時に処理するために使用可能なCPUの数に基づいて、アップグレードの実行に必要な時間を短縮します。

#### アップグレード後に無効なオブジェクトを再コンパイル

このオプションは、アップグレード完了後にすべての無効なPL/SQLモジュールを再コンパイルします。アップグレード後フェーズでDBUAによる無効なオブジェクトの再コンパイルを行わない場合は、データベースをアップグレードした後に無効なオブジェクトを手動で再コンパイルする必要があります。

#### タイム・ゾーン・データのアップグレード

このオプションは、このリリースのタイムゾーン・データ・ファイルを更新します。このオプションを選択しない場合は、アップグレード後にタイム・ゾーン構成ファイルを手動で更新する必要があります。

#### 実行するカスタムSQLスクリプトの指定

アップグレードの一部としてカスタムSQLスクリプトを実行する場合は、このオプションを選択します。必要に応じて「アップグレード前」または「アップグレード後」入力フィールドの「参照」をクリックします。カスタムSQLスクリプトがある場所に移

動します。

選択が完了したら、「次へ」をクリックします。

9. リカバリ・オプションの選択ウィンドウが表示されます。アップグレード中にエラーが発生した場合にデータベースをリカバリするには、次のオプションのいずれかを選択します。

- フラッシュバックおよび保証付きリストア・ポイントの使用

新しい保証付きリストア・ポイントを作成するか、既存のものを使用できます。既存のリストア・ポイントを使用する場合は、選択フィールドをクリックして使用するリストア・ポイントを選択します。

### 注意:



アップグレードするデータベースに Oracle Data Guard フィジカル・スタンバイが含まれている場合は、最初に各スタンバイに保証付きリストア・ポイントを作成してから、プライマリ・データベースに保証付きリストア・ポイントを作成する必要があります。最初に各スタンバイにリストア・ポイントを作成しない場合は、保証付きリストア・ポイントを使用してプライマリ・データベースをダウングレードした後、すべてのスタンバイ・データベースを再作成する必要があります。アップグレードが正常に終了した後、スタンバイ上のすべての保証付きリストア・ポイントを手動で削除する必要があります。

- RMANバックアップの使用

新しいオフラインRMANバックアップを作成するか、既存のバックアップを使用できます。「参照」をクリックしてバックアップ用のパスを指定します。

- 最新の使用可能なRMANバックアップの使用

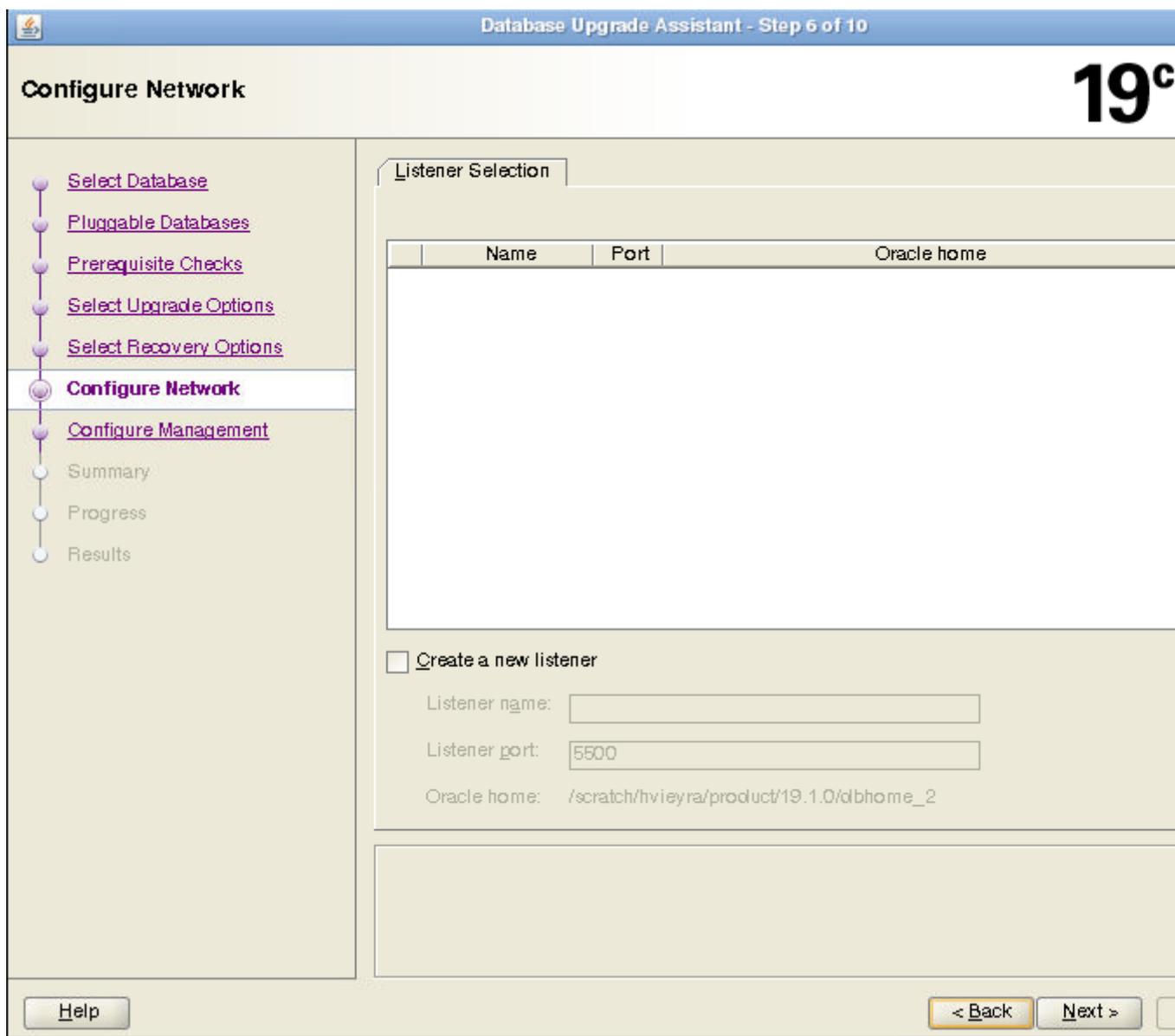
既存のバックアップを使用できます。「リストア・スクリプトの表示/編集」をクリックして、使用するバックアップを選択します。

- 独自のバックアップおよびリストア計画

既存のデータベースにサード・パーティのバックアップ・ソリューションが存在する場合のみ、このオプションを選択します。

選択が完了したら、「次へ」をクリックします。

10. 単一インスタンス・データベースのインストールの場合は、「ネットワークの構成」ウィンドウが開きます。アップグレードした新しいOracleホームに移行するソースOracleホームから1つ以上のリスナーを選択するか、インストール中に新しいリスナーを作成します。



「ネットワーク構成」ウィンドウの「リスナーの選択」領域には、次の列がある表が表示されます。

- 選択列。更新するリスナーを選択します。
- 名前。この列にはリスナーの名前が表示されます。
- ポート。この列にはリスナーが構成されるポートが表示されます。
- Oracleホーム。この列にはリスナーが構成されるOracleホームが表示されます。
- ステータス。この列にはリスナーのステータス(upまたはdown)が表示されます。
- 移行。この列を選択し、移行するには「はい」を、移行しない場合は「いいえ」を選択します。

新しいリスナーの作成を選択することもできます。新しいリスナーを作成する場合はリスナー名、リスナーを配置するOracleホーム、およびリスナーがモニターするポートを指定します。

選択が完了すると、DBUAによって、移行するすべてのリスナーで次のステップが実行されます。

- DBUAによって、選択したリスナーがターゲットOracleホームのlistener.oraファイルに追加され、開始されます。
- DBUAによって、古い(ソース)listener.oraファイルからアップグレードされたデータベースのエントリが削除されます。

- DBUAによって、ソースとターゲット両方のOracle Database環境で、listener.oraファイルが再ロードされます。

**注意:**



同じリスナーに登録されている他のデータベースがある場合、それらの新規クライアント接続リクエストはリスナーの移行時に影響を受ける可能性があります。

選択が完了したら、「次へ」をクリックします。

11. 「管理の構成」ウィンドウが表示されます。管理の構成ウィンドウで、管理オプションを選択します。

- Enterprise Manager (EM) Database Expressの構成

Oracle Enterprise Manager Database Expressは、Oracle Databaseに組み込まれているWebベースのデータベース管理アプリケーションです。EM Expressは、以前のリリースで使用されていたDB Controlコンポーネントを置き換えます。Enterprise Manager Database Expressの構成を選択した場合は、EM Database Expressのポート番号を入力します。たとえば、5500と入力します。また、チェッ

ク・ボックスを選択して、Expressのポートをグローバル・ポートとして構成することもできます。

- Enterprise Manager (EM) Cloud Controlへの登録

Oracle Enterprise Manager Cloud Controlへの登録では、EM Cloud Controlで管理できるターゲットとして、データベースおよび関連するリスナー、Oracle ASMディスク・グループ、Oracle Clusterwareなどのエンティティを追加します。

このオプションを選択した場合は、次の各フィールドに情報を入力する必要があります。

- OMSホスト
- OMSポート
- EM管理ユーザー名
- EM管理パスワード
- DBSNMPユーザー・パスワード
- ASMSNMPユーザー・パスワード

情報の入力が完了したら、「次へ」をクリックします。

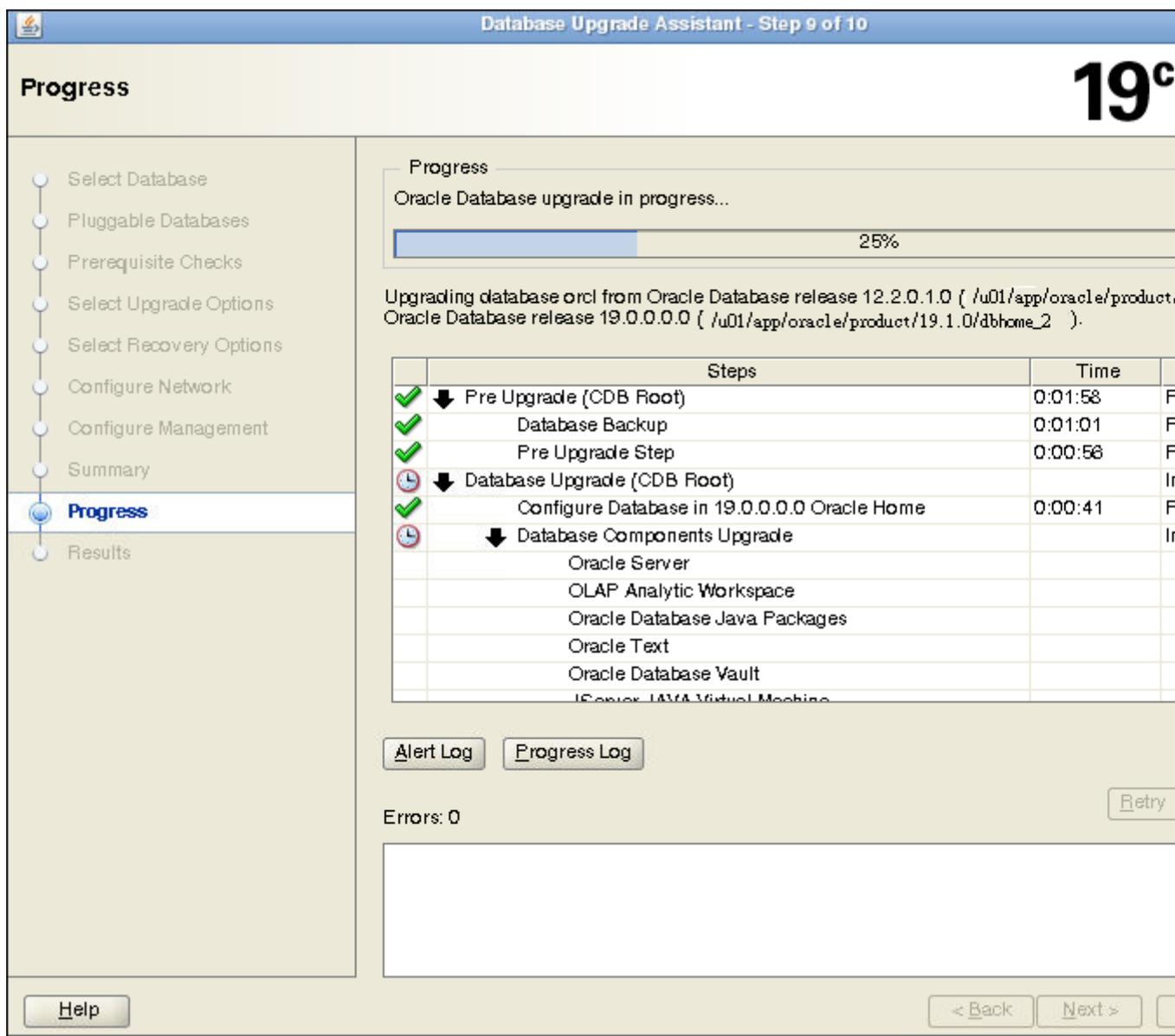
12. 「サマリー」ウィンドウが開きます。「サマリー」ウィンドウに、アップグレードのために指定した情報が表示されます。情報を確認するには、リストを下にスクロールします。サマリーには次のような情報が含まれます。

- ソース・データベース
- ターゲット・データベース
- プラガブル・データベース
- アップグレード前のチェック
- 初期化パラメータの変更
- タイムゾーンのアップグレード

選択を確認します。それから変更する項目のリンクを選択するか、「戻る」をクリックして前のページに移動するか、または「終了」を選択します。

- 「サマリー」ウィンドウに修正が必要な情報が表示された場合は、更新する項目のリンクをクリックするか、または、「戻る」をクリックしてDBUA構成ウィザードを逆に移動します。
- 「サマリー」ウィンドウに表示されている情報が正しい場合、「終了」をクリックします。「終了」をクリックするとアップグレードが開始されます。

DBUAによるアップグレードが開始されると、「進行状況」ウィンドウにプログレス・バーが表示されます。「進行状況」ウィンドウに、アップグレード中にDBUAが完了したステップを含む表が表示されます。この表には、アップグレードの進行につれて、継続時間とアップグレード・ステップのステータスが表示されます。DBUAでは、この時点でアップグレードを取り消す必要がある場合に備えて、「停止」ボタンが提供されています。



CDBルートおよび各PDBシードのアップグレードの終了までアップグレードが進行すると、「進行状況」ウィンドウでは、ステータスが「終了」とマークされます。

- アップグレードが完了すると、「結果」ウィンドウが開きます。「結果」ウィンドウには元のデータベースとアップグレードしたデータベースに関する情報が表示されます。「アップグレード結果」が表示され、DBUAが初期化パラメータに加えた変更も表示されます。マルチテナント・アーキテクチャ・データベースをアップグレードしている場合は、「結果」ウィンドウには、プラグブル・データベースと、アップグレード後にログ・ファイルが格納されるディレクトリも表示されます。アップグレード前チェックの詳細を表示するには、下にスクロールします。アップグレードが正常に終了した場合は、「アップグレード結果」フィールドに結果が表示され、警告メッセージは表示されません。サンプル画像に示されているように、アップグレードが正常に終了しなかった場合は、表示フィールドの右下隅に「データベースのリストア」ボタンが表示されます。このボタンをクリックしてデータベースの復元を開始できます。

Database Upgrade Assistant - Step 10 of 10

## 19c

### Results

- Select Database
- Pluggable Databases
- Prerequisite Checks
- Select Upgrade Options
- Select Recovery Options
- Configure Network
- Configure Management
- Summary
- Progress
- Results**

### Upgrade Results

Database upgrade has been completed successfully, and the database is ready to

|                     | Source Database                        | Target Database           |
|---------------------|----------------------------------------|---------------------------|
| <b>Name:</b>        | orcl                                   | orcl                      |
| <b>Release:</b>     | 12.2.0.1.0                             | 19.0.0.0.0                |
| <b>Oracle Home:</b> | u01/app/oracle/product/12.2.0/dbhome_1 | u01/app/oracle/product/19 |

### Pluggable Databases

| Pluggable Database | Status | Version    |
|--------------------|--------|------------|
| ORCLPDB            | NORMAL | 19.0.0.0.0 |

### Upgrade Details

The following summary lists the steps performed during the upgrade process. Log files for all the a this summary, are available at /u01/app/oracle/cfgtoollogs/dbua/upgrade/2018-11-01\_12-10-14

| Step Name | Log File Name | Status |
|-----------|---------------|--------|
|-----------|---------------|--------|

14. オプション: アップグレード処理の詳細を確認するには、このログ・ファイルを確認します。Oracleベース環境変数が設定されている場合は、DBUAログ・ファイルはパス  
 /ORACLE\_BASE/cfgtoollogs/dbua/upgradesession\_timestamp/SIDにあります。Oracleベースが設定されていない場合は、DBUAログ・ファイルは、パス/ORACLE\_HOME/cfgtoollogs/dbua/upgradesession\_timestamp/SID  
 にあります

#### 注意:



「アップグレード結果」ウィンドウの HTML バージョンも、ログ・ファイル・ディレクトリに保存されます。この HTML ウィンドウのリンクをクリックすると、ブラウザにログ・ウィンドウが表示されます。

アップグレード結果に問題がない場合は、「閉じる」をクリックしてDBUAを終了します。

15. アップグレードが完了したら、このマニュアルに記載されているアップグレード後の手順を実行します。アップグレード後の手順が完了したら、アップグレードしたデータベースをいつでも使用できます。

## 注意:



データベースの不正な使用を防止するために、データベースをアップグレードした直後にすべてのユーザー・パスワードを変更することをお勧めします。

Oracle Database 12c 以降のデフォルトのセキュリティ設定が設定されている場合、パスワードの文字数は 8 文字以上である必要があります。welcome や oracle などのパスワードは使用できません。

親トピック: [Oracle Databaseのアップグレード](#)

# 非CDBアーキテクチャのOracle Databaseの手動アップグレード

この手順では、非CDBアーキテクチャのOracle Databaseをアップグレードするためのステップを示します。

## 注意:



Oracle Database 12c リリース 1 (12.1)以降では、非 CDB アーキテクチャは非推奨です。今後のリリースでサポートが終了する可能性があります。

この手順を使用する前に、次のステップを実行してください。

- Oracle Databaseソフトウェアをインストールします
- 新しいOracleホームを準備します。
- アップグレード前情報ツールの実行

ステップは次のとおりです。

1. アップグレード前情報ツールをまだ実行していない場合、実行します。次に進む前に、アップグレード前情報ツールの出力を確認して、出力に記載されているすべての問題を修正します。

たとえば、LinuxまたはUNIXシステムで次のように指定します。

```
$ORACLE_HOME/jdk/bin/java -jar /opt/oracle/product/19.0.0/rdbms/admin/preupgrade.jar FILE TEXT
```

2. 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。
3. 新しいOracleホームをまだ準備していない場合、準備します。
4. (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FALSEに設定します。

```
ALTER SYSTEM SET CLUSTER_DATABASE=FALSE SCOPE=SPFILE;
```

5. データベースを停止します。次に例を示します。

```
SQL> SHUTDOWN IMMEDIATE
```

6. ご使用のオペレーティング・システムがWindowsの場合、次のステップを実行します。

- a. アップグレードするデータベースのOracleサービスOracleServiceSIDを停止します(SIDはインスタンスの名前です)。たとえば、SIDがORCLの場合、コマンド・プロンプトで次のように入力します。

```
C:¥> NET STOP OracleServiceORCL
```

- b. コマンド・プロンプトで、ORADIMを使用してOracleサービスを削除します。ORADIMの構文およびコマンドの完全なリストは、ご使用のプラットフォームのガイドを参照してください。

たとえば、SIDがORCLの場合、次のコマンドを入力します。

```
C:¥> ORADIM -DELETE -SID ORCL
```

- c. コマンド・プロンプトで新しいOracle DatabaseリリースのORADIMコマンドを使用して、新しいリリースのOracle Databaseのサービスを作成します。

次の構文を使用します。SIDはデータベースSID、PASSWORDはシステム・パスワード、USERSは最大ユーザー数に設定する値、ORACLE\_HOMEはOracleホームです。

```
C:¥> ORADIM -NEW -SID SID -SYSPWD PASSWORD -MAXUSERS USERS
-STARTMODE AUTO -PFILE ORACLE_HOME¥DATABASE¥INITSID.ORA
```

ほとんどのOracle Databaseサービスは、Oracleソフトウェア・インストール所有者の権限を使用してシステムにログオンします。サービスは、このユーザーの権限を使用して実行されます。ORADIMコマンドによって、このユーザー・アカウントのパスワードの入力を求められます。ORADIMを使用して、他のオプションを指定できます。

次の例で、SIDがORCL、password (SYSPWD)がTWxy5791、ユーザーの最大数(MAXUSERS)が10、およびOracleホーム・パスがC:¥ORACLE¥PRODUCT¥19.0.0¥DBの場合、次のコマンドを入力します。

```
C:¥> ORADIM -NEW -SID ORCL -SYSPWD TWxy5791 -MAXUSERS 10
-STARTMODE AUTO -PFILE C:¥ORACLE¥PRODUCT¥19.0.0¥DB¥DATABASE¥INITORCL.ORA
```

ORADIMによって、ORACLE\_HOME¥databaseディレクトリにログが書き込まれます。



#### 注意:

Oracle ホームのユーザー・アカウントを使用して Oracle ホームを所有する場合、ORADIM コマンドによって、そのユーザーの名前とパスワードを求められます。

7. ご使用のオペレーティング・システムがLinuxまたはUNIXの場合は、次の内容を確認します。

- a. ORACLE\_SIDが正しく設定されていること。
- b. oratabファイルが新しいOracleホームを指していること
- c. 次の環境変数が新しいOracle Databaseディレクトリを指していること
  - ORACLE\_HOME
  - PATH
- d. クライアントで\$ORACLE\_HOME環境変数を設定するために使用するどのスクリプトも、新しいOracleホームを指している必要があります。

#### 注意:



Oracle Real Application Clusters データベースをアップグレードする場合、Oracle Real Application Clusters データベースでインスタンスが構成されているすべての Oracle Grid Infrastructure ノードでこれらのチェックを実行してください。

8. 新しいOracle DatabaseリリースのOracleインストール所有者としてシステムにログインします。

9. 新しいOracleホーム・ディレクトリのadminディレクトリから、新しいOracleホームでSQL\*Plusを起動します。

次に例を示します。

```
$ cd $ORACLE_HOME/rdbms/admin
$ pwd
/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin
$ sqlplus
```

10. SPFILE.ORAまたはINIT.ORAファイルを古いOracleホームから新しいOracleホームにコピーします。
11. SYSDBA権限を持つアカウントを使用して、アップグレードするデータベースに接続します。

```
SQL> connect / as sysdba
```

12. アップグレード・モードで非CDB Oracle Databaseを起動します。

```
SQL> startup upgrade
```

サポートが終了した初期化パラメータを示すエラーが表示された場合は、そのサポートが終了した初期化パラメータを記録し、アップグレードを継続します。次回、データベースを停止したときに、そのサポートが終了した初期化パラメータを削除します。

### 注意:



UPGRADE モードでデータベースを起動すると、以前の Oracle Database リリースに基づいてデータベースがオープンされます。また、AS SYSDBA セッションへのログインの制限、システム・トリガーの無効化、およびアップグレードの環境を準備するその他の操作も実行されます。

13. SQL\*Plusを終了します。

次に例を示します。

```
SQL> EXIT
```

14. アップグレードに必要なアップグレード・オプションを使用して、パラレル・アップグレード・ユーティリティ (catctl.pl)スクリプトを実行します。

パラレル・アップグレード・ユーティリティは、Oracle\_home/binにあるdbupgradeシェル・コマンドを使用すると、コマンドライン・シェル・コマンドとして実行できます。Oracle\_home/binを含むようにPATH環境変数を設定した場合、コマンドラインから直接コマンドを実行できます。次に例を示します。

```
$ dbupgrade -d /u01/app/oracle/19.1.0/dbhome_1
```

### 注意:



- パラレル・アップグレード・ユーティリティ・コマンドを実行する場合は、-d オプションを使用してターゲット Oracle ホームのファイル・パスを指定します。-l オプションを使用して、スプール・ログ・ファイルに使用するディレクトリを指定します。

15. アップグレードが成功した後、データベースは停止します。インスタンスを再起動して、通常の操作用にシステ

ム・パラメータを再初期化します。次に例を示します。

```
SQL> STARTUP
```

このようにデータベースが停止してから再起動することで、すべてのキャッシュがフラッシュされ、バッファが消去されて、その他のハウスキeping・アクティビティが実行されます。これらの操作は、アップグレードしたOracle Databaseソフトウェアの整合性と一貫性を保証するための重要な最終ステップです。

### 注意:



データベースを起動したときに、サポートが終了した初期化パラメータがリストされているメッセージが表示された場合は、再起動の前にサポートが終了した初期化パラメータをパラメータ・ファイルから削除します。ファイルを編集してパラメータを削除するために、必要に応じて SPFILE を PFILE に変換します。

16. `catcon.pl` を実行して `utlrp.sql` を起動し、残りの無効なオブジェクトをすべて再コンパイルします。

次に例を示します。

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

`-b utlrp` を使用してコマンドを実行しているため、ログ・ファイル `utlrp0.log` はスクリプトが実行される時に生成されます。ログ・ファイルには再コンパイルの結果が含まれています。

17. `postupgrade_fixups.sql` を実行します。次に例を示します。

```
SQL> @postupgrade_fixups.sql
```

### 注意:



スクリプトを別の場所に配置するように指定しなかった場合、スクリプトはデフォルトのパス `Oracle_base/cfgtool/logs/SID/preupgrade` に配置されます。Oracle\_base は Oracle ベースのホーム・パスで、SID は一意のデータベース名です。

18. `utlusts.sql` を実行します。スクリプトは、すべての問題が修正されたことを検証します。

次に例を示します。

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

スクリプトを実行すると、アップグレードの結果を示すログ・ファイル `utlrp0.log` が生成されます。 `upg_summary.log` でアップグレード・レポート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、このガイドのトラブルシューティングに関する項を参照してください。

19. `DBMS_DST PL/SQL` パッケージを使用してタイムゾーン・ファイルをアップグレードし、タイムゾーン・データファイルを最新の状態にします。アップグレード後にタイムゾーン・データ・ファイルを調整することもできます。

## 20. SQL\*Plusを終了します

次に例を示します。

```
SQL> EXIT
```

21. (条件付き) Oracle Real Application Clustersデータベースをアップグレードする場合、次のコマンド構文を使用して、Oracle Clusterwareのデータベース構成をアップグレードします。

```
srvctl upgrade database -db db-unique-name -oraclehome oraclehome
```

この構文例で、db-unique-nameは(インスタンス名ではなく)データベース名であり、oraclehomeは、データベースがアップグレードされるOracleホームの場所です。SRVCTLユーティリティでは、以前のリリースで使用される短いコマンドライン・インタフェース(CLI)オプションに加えて、GNUスタイルの長いオプションがサポートされています。

22. (条件付き) Oracle RAC環境の場合のみ、すべてのノードをアップグレードした後に、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、データベースを起動します(db\_unique\_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。アップグレード後の手順を完了する準備が整いました。

### 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:** [Oracle Databaseのアップグレード](#)

# 10 Oracle Databaseのアップグレード後の作業

Oracle Databaseをアップグレードした後に、必要なアップグレード後のタスクを完了し、新しいリリースの推奨事項を検討してください。

- [アップグレード後の状態ツールを使用したアップグレードの確認](#)  
アップグレードのスプール・ログ・ファイルを確認し、アップグレード後の状態ツール(utlusts.sql)を使用します。
- [Oracleデータ・ディクショナリの現在の状態の表示方法](#)  
3つの方法のうちのいずれかを使用して、アップグレードと移行を診断するためにOracleデータ・ディクショナリの状態を確認します。
- [Oracle Databaseのアップグレード後に必要な作業](#)  
アップグレードの完了後、現在の環境に対して指定されたこれらの必要な作業を確認して完了します。
- [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)  
Oracle Databaseを更新する場合、これらの適切なプラクティス・ガイドラインを完了することをお勧めします。これらのプラクティスは、手動とDBUAの両方のアップグレードにお勧めです。

# アップグレード後の状態ツールを使用したアップグレードの確認

アップグレードのスプール・ログ・ファイルを確認し、アップグレード後の状態ツール(`utlusts.sql`)を使用します。

アップグレード後の状態ツールは、Oracle Databaseに付属するSQLスクリプトです。アップグレード後の状態ツールは、新しいリリースの環境で実行します。アップグレード後の状態ツールは、データベースのアップグレード後にもいつでも実行できます。

**親トピック:** [Oracle Databaseのアップグレード後の作業](#)

# Oracleデータ・ディクショナリの現在の状態の表示方法

3つの方法のうちのいずれかを使用して、アップグレードと移行を診断するためにOracleデータ・ディクショナリの状態を確認します。

dbupgdiag.sqlスクリプトの実行

dbupgdiag.sqlスクリプトによって、データ・ディクショナリの現在の状態に関するアップグレードおよび移行の診断情報を収集できます。

スクリプトは、SYSユーザーとしてSQL\*Plusを使用して、ソース・データベースでアップグレード前に実行することも、アップグレードしたデータベースでアップグレード後に実行することもできます。アップグレードおよびマイグレート診断情報を収集するためのdbupgdiag.sqlスクリプトの使用の詳細は、My Oracle Supportノート556610.1を参照してください。

DBA\_REGISTRYでのSQL問合せの実行

ディクショナリの現在の状態を表示するには、次の例のようなSQL問合せを実行します。

```
SQL> spool /tmp/regInvalid.out
SQL> set echo on
-- query registry
SQL> set lines 80 pages 100
SQL> select substr(comp_id, 1, 15) comp_id, substr(comp_name, 1, 30)
 comp_name, substr(version, 1, 10) version, status
from dba_registry order by modified;
```

無効なオブジェクトをチェックする問合せの実行

無効なオブジェクトを問い合わせるには、次の例のようなSQL問合せを実行します。

```
SQL> select owner, object_name, object_type from dba_invalid_objects order by owner, object_type;
```

データベースをアップグレードし、utlrp.sqlを実行した後は、このビューの問合せで行が返されない必要があります。

## 関連トピック

- <https://support.oracle.com/rs?type=doc&id=556610.1>

親トピック: [Oracle Databaseのアップグレード後の作業](#)

# Oracle Databaseのアップグレード後に必要な作業

アップグレードの完了後、現在の環境に対して指定されたこれらの必要な作業を確認して完了します。

Oracle Databaseをアップグレードした後で、次のアップグレード後タスクを実行する必要があります。これらのタスクは手動でアップグレードを実行した場合も、Database Upgrade Assistant (DBUA)を使用してアップグレードした場合でも実行する必要があります。

- [手動アップグレード後のLinuxおよびUNIXシステム上での環境変数の設定](#)  
Oracle Databaseの手動アップグレードを実行した場合、必要とされるオペレーティング・システム的环境変数が、新しいOracle Databaseリリースのディレクトリを指していることを確認する必要があります。
- [すべての無効なオブジェクトの再コンパイル](#)  
データベースのインストール、パッチ適用またはアップグレードの後にutlrp.sqlスクリプトを実行し、無効なオブジェクトを特定して再コンパイルすることをお勧めします。
- [無効なオブジェクトの再コンパイルの進行状況の追跡](#)  
これらのSQL問合せを使用して、utlrp.sqlスクリプトによる無効なオブジェクトの再コンパイルの進行状況を追跡します。
- [Oracle Databaseのアップグレード後のOPatchコマンドの実行](#)  
Oracle Databaseをアップグレードした後で、新しいOracleホームからOPatchコマンドを実行する必要があります。
- [Oracle Databaseのアップグレード後にoratabおよびスクリプトが新しいOracleの場所を指すようにするための設定](#)  
新しいOracleホームの場所を指すようにスクリプトを設定する必要があります。
- [PL/SQLパッケージおよび依存プロシージャの確認](#)  
以前のリリースのOracle Databaseにインストールしたパッケージは新しいリリースでは使用できない可能性があり、このことがアプリケーションに影響する場合があります。
- [Oracle管理タイプに依存する表のアップグレード](#)  
Oracle Database 12cリリース2 (12.2)以降では、Oracle管理タイプに依存するユーザー表を手動でアップグレードする必要があります。
- [新しい拡張データ型機能の有効化](#)  
システムで新しい拡張データ型を利用できるようにするには、特定のアップグレード操作が必要です。
- [パラレル実行サーバーの最大値および最小値の調整](#)  
環境に応じて、**PARALLEL\_MIN\_SERVERS**パラメータのデフォルト設定を減らすことができます。
- [Oracle Databaseのアップグレード後のリカバリ・カタログ・アップグレードの概要](#)  
RMANクライアントで要求されるバージョンより古いリカバリ・カタログ・スキーマを使用している場合、それをアップグレードする必要があります。
- [Oracle Databaseのアップグレード後のタイムゾーン・ファイルのバージョンのアップグレード](#)  
データベースのアップグレードを完了した後、アップグレード前情報ツールによってタイムゾーン・ファイルをアップグレードするよう指示された場合は、DBMS\_DST PL/SQLパッケージを使用してタイムゾーン・ファイルをアップグレードします。
- [Oracle Databaseのアップグレード後のDBMS\\_STATSパッケージで作成された統計表のアップグレード](#)  
DBMS\_STATS.CREATE\_STAT\_TABLEプロシージャを使用して統計表を作成した場合、DBMS\_STATS.UPGRADE\_STAT\_TABLEを実行してそれらの表をアップグレードします。
- [Oracle Databaseのアップグレード後の外部認証されたSSLユーザーのアップグレード](#)  
Oracle9iリリース2 (9.2)またはOracle Database 10gリリース1 (10.1)からのアップグレード時に、外部認証されたSSLユーザーを使用している場合は、SSL外部ユーザー変換(extsusupgrade)スクリプトを実行してそれらのユーザーをアップグレードする必要があります。
- [Oracle XML DBに対するFTPとHTTPのポートおよびHTTP認証の構成](#)

Oracle Database Configuration Assistant (DBCA)では、Oracle Database 12cのOracle XML DBのポートは構成されず、それ以降のリリースのアップグレードではダイジェスト認証が使用されます。

- [Oracle Databaseのアップグレード後のOracle Textが提供するナレッジ・ベースのインストール](#)  
Oracle Databaseのアップグレードの後、Oracle Textナレッジ・ベースに対するすべてのユーザー拡張を再生成する必要があります。
- [Oracle Databaseのアップグレード後のOracle Application Express構成の更新](#)  
Oracle Application Expressのリリースとデータベースのインストール・タイプによっては、Oracle Application Expressがアップグレード手順に影響します。
- [外部ネットワーク・サービスへのアクセス制御リスト\(ACL\)の構成](#)  
Oracle Database 12c以降のリリースには、UTL\_TCP、UTL\_SMTP、UTL\_MAIL、UTL\_HTTPまたはUTL\_INADDRパッケージに対するファイナグレイン・アクセス制御が含まれています。
- [SQLNET.ALLOWED\\_LOGON\\_VERSIONパラメータの動作の確認](#)  
10gより前のリリースのクライアントからのOracle Databaseに対する接続は、ORA-28040: 「一致する認証プロトコルがありません」というエラーによって失敗します。

**親トピック:** [Oracle Databaseのアップグレード後の作業](#)

# 手動アップグレード後のLinuxおよびUNIXシステム上での環境変数の設定

Oracle Databaseの手動アップグレードを実行した場合、必要とされるオペレーティング・システムの環境変数が、新しいOracle Databaseリリースのディレクトリを指していることを確認する必要があります。

次の環境変数が新しいOracleホームのディレクトリを指していることを確認します。

- ORACLE\_HOME
- PATH

注意:



DBUA では、自動的に Oracle 環境変数に対して必要な変更が行われます。

## 関連トピック

- [Oracle Database管理者ガイド](#)

親トピック: [Oracle Databaseのアップグレード後に必要な作業](#)

# すべての無効なオブジェクトの再コンパイル

データベースのインストール、パッチ適用またはアップグレードの後にutlrp.sqlスクリプトを実行し、無効なオブジェクトを特定して再コンパイルすることをお勧めします。

utlrp.sqlスクリプトは、すべての無効なオブジェクトを再コンパイルします。インストールの直後にスクリプトを実行して、ユーザーが無効なオブジェクトにアクセスしないようにしてください。

1. SQL\*Plusを起動します。

```
sqlplus "/ AS SYSDBA"
```

2. utlrp.sqlスクリプトを実行します。Oracle\_homeはOracleホームのパスです。

```
SQL> @Oracle_home/rdbms/admin/utlrp.sql
```

utlrp.sqlスクリプトは、無効なオブジェクトの数と使用可能なCPUの数の両方に基づいて、シリアル再コンパイルまたはパラレル再コンパイルで無効なオブジェクトを自動的に再コンパイルします。CPUは、CPUの数(**cpu\_count**)にCPUごとのスレッドの数(**parallel\_threads\_per\_cpu**)を乗じて計算されます。Oracle Real Application Clusters (Oracle RAC)では、すべてのOracle RACノード全体でこの数が追加されます。

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# 無効なオブジェクトの再コンパイルの進行状況の追跡

これらのSQL問合せを使用して、utlrp.sqlスクリプトによる無効なオブジェクトの再コンパイルの進行状況を追跡します。

アップグレード後にutlrp.sqlスクリプトを実行して無効なオブジェクトを再コンパイルすることをお勧めします。SQL問合せを実行してスクリプトを監視できます。

## 例10-1 残っている無効なオブジェクトの数

次の問合せを入力すると、残っている無効なオブジェクトの数が返されます。この数は、utlrp.sqlスクリプトの実行につれて時間とともに減少します。

```
SELECT COUNT(*) FROM obj$ WHERE status IN (4, 5, 6);
```

## 例10-2 再コンパイルされたオブジェクトの数

次の問合せを入力すると、utlrp.sqlによってコンパイルされたオブジェクトの数が返されます。この数は、スクリプトの実行につれて時間とともに増加します。

```
SELECT COUNT(*) FROM UTL_RECOMP_COMPILED;
```

## 例10-3 エラー付きで再コンパイルされたオブジェクトの数

次の問合せを入力すると、utlrp.sqlによってエラー付きでコンパイルされたオブジェクトの数が返されます。

```
select COUNT (DISTINCT(obj#)) "OBJECTS WITH ERRORS" from utl_recomp_errors;
```

この数が予想より多い場合、各オブジェクトでレポートされたエラー・メッセージを確認してください。エラーの原因がシステムの間違った構成やリソース制約にある場合、それらのエラーの原因を修正してutlrp.sqlを再度実行します。

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のOPatchコマンドの実行

Oracle Databaseをアップグレードした後で、新しいOracleホームからOPatchコマンドを実行する必要があります。

OPatchは、Oracle Universal Installerを使用してインストールするJavaベースのユーティリティです。Opatchはプラットフォームに依存します。サポートされているすべてのオペレーティング・システム上で実行されます。スタンドアロンOPatchと呼ばれる別のバージョンのOPatchも使用できます。これは、Oracle Universal Installerを使用せずにOracleホームで実行されません。

パッチとは、既存のインストールにコピーされる小型のファイル・コレクションです。パッチはOracle製品の特定のバージョンに関連付けられています。インストール済の対応バージョンの製品にパッチを適用すると、製品のバージョンがアップグレードされます。

### Oracle Databaseインベントリを確認するためのOPatchの実行

Oracleインストール所有者としてログインし、新しいOracleホームからlsinventoryコマンドを実行します。このコマンドは、使用しているシステムにインストールされているOracleソフトウェアの正確かつ完全なインベントリを生成します。

```
opatch lsinventory -patch
```

My Oracle Supportのノート756671.1を参照して、リリース更新(Updates)およびリリース更新リビジョン(Revisions)に関する最新の推奨事項を定期的に入手してください。

### 関連トピック

- <https://support.oracle.com/rs?type=doc&id=756671.1>

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後にoratabおよびスクリプトが新しいOracleの場所を指すようにするための設定

新しいOracleホームの場所を指すようにスクリプトを設定する必要があります。

Oracle Databaseを新しいリリースにアップグレードした後に、oratabファイルおよびORACLE\_HOME値を設定するすべてのクライアント・スクリプトが、新しいOracle Databaseリリース用に作成された新しいOracleホームを指していることを確認する必要があります。DBUAでは、oratabファイルは自動的に新しいOracleホームを指します。ただし、クライアント・スクリプトは、アップグレードに使用する方法にかかわらず確認する必要があります。

データベースを手動でアップグレードする場合、新しいOracle DatabaseリリースのOracleインストール所有者としてログインし、oratabファイルを手動で更新する必要があります。oratabファイルの場所は、使用しているオペレーティング・システムによって異なります。

## 関連項目:

オペレーティング・システムの環境変数の設定の詳細は、[『Oracle Database管理者ガイド』](#)を参照してください

My Oracle Support: oratabファイルの検索または作成(ドキュメントID 394251.1)

<https://support.oracle.com/rs?type=doc&id=394251.1>

親トピック: [Oracle Databaseのアップグレード後に必要な作業](#)

# PL/SQLパッケージおよび依存プロシージャの確認

以前のリリースのOracle Databaseにインストールしたパッケージは新しいリリースでは使用できない可能性があり、このことがアプリケーションに影響する場合があります。

アップグレード後に、独自のスクリプトで使用していたパッケージまたは独自のスクリプトからコールしていたパッケージがすべて新しいリリースで使用できることを確認してください。パッケージに依存するプロシージャのテストは、アップグレード計画に含まれる必要があります。

データベース・アプリケーションのコードは、接続先データベースのオブジェクトを参照できます。たとえば、Oracle Call Interface(OCI)およびプリコンパイラ・アプリケーションは無名PL/SQLブロックを発行できます。Oracle Formsアプリケーションのトリガーは、スキーマ・オブジェクトを参照できます。これらのアプリケーションは、参照しているスキーマ・オブジェクトに依存しています。依存性管理の方法は開発環境によって異なります。Oracle Databaseでは、アプリケーションの依存性が自動的に追跡されることはありません。

## 関連トピック

- [Oracle Database管理者ガイド](#)

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle管理タイプに依存する表のアップグレード

Oracle Database 12cリリース2 (12.2)以降では、Oracle管理タイプに依存するユーザー表を手動でアップグレードする必要があります。

Oracle管理タイプに依存するユーザー表(AQキュー表など)がデータベースに含まれる場合、アップグレード後に `utluptabdata.sql` コマンドを実行して、Oracle管理タイプの変更の影響を受けるすべてのユーザー表に対して `ALTER TABLE UPGRADE` を実行します。この動作変更によって、ユーザー表は、アップグレード中に `READ ONLY` 状態のままになります。ユーザーは、`SYSDBA` 権限 (`AS SYSDBA`) を使用してアプリケーションにログインし、Oracle管理タイプに依存するアプリケーション表を変更できなくなります。

データベース・アップグレードの完了後にアップグレードする必要がある表を特定するには、データベースに `AS SYSDBA` で接続し、次の問合せを実行します。

```
COLUMN owner FORMAT A30
COLUMN table_name FORMAT A30
SELECT DISTINCT owner, table_name
FROM dba_tab_cols
WHERE data_upgraded = 'NO'
ORDER BY 1,2;
```

この問合せによって、`UPGRADED` としてリストされないすべての表が表示されます。ただし、`utluptabdata.sql` スクリプトは、Oracle管理タイプに依存する表のみをアップグレードします。問合せによって表がリストされた場合、`utluptabdata.sql` スクリプトを実行して依存ユーザー表に `ALTER TABLE UPGRADE` コマンドを実行し、これらのOracle管理タイプをそのタイプの最新バージョンにアップグレードします。

`utluptabdata.sql` スクリプトは、Oracle管理タイプに依存するすべての表に対する `ALTER` 権限を持つユーザー・アカウントか、または `AS SYSDBA` でログインしている `SYSDBA` システム権限を付与されたユーザーを使用して実行する必要があります。

パラメータ `SERVEROUTPUT` が `ON` に設定されている場合、`utluptabdata.sql` スクリプトによって、アップグレードされたすべての表の名前が表示され、表のアップグレード中に発生したエラーがリストされます。サーバー出力を `ON` に設定するには、次のコマンドを実行します。

```
SET SERVEROUTPUT ON
@utluptabdata.sql
```

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# 新しい拡張データ型機能の有効化

システムで新しい拡張データ型を利用できるようにするには、特定のアップグレード操作が必要です。

Oracle Database 12cでは、SQLのVARCHAR2、NVARCHAR2およびRAWデータ型の最大サイズを制御するMAX\_STRING\_SIZEが導入されました。MAX\_STRING\_SIZE = EXTENDEDを設定することで、Oracle Database 12cで導入された32767バイト制限が有効になります。

MAX\_STRING\_SIZE = EXTENDEDを設定するには、COMPATIBLE初期化パラメータを12.0.0.0以上に設定する必要があります。また、データベースがアップグレード・モードでオープンしているときにutl32k.sqlスクリプトを実行することによって、データ型サイズの変更の影響を受けるオブジェクトを無効にして再コンパイルする必要があります。次に例を示します。

```
CONNECT SYS / AS SYSDBA
SHUTDOWN IMMEDIATE;
STARTUP UPGRADE;
ALTER SYSTEM SET max_string_size=extended;
START $ORACLE_HOME/rdbms/admin/utl32k.sql
SHUTDOWN IMMEDIATE;
STARTUP;
```

## 注意:



MAX\_STRING\_SIZE の値は STANDARD から EXTENDED に変更できます。ただし、MAX\_STRING\_SIZE の値を EXTENDED から STANDARD には変更できません。MAX\_STRING\_SIZE = EXTENDED を設定することは、データベースでアプリケーションの非互換性が発生する可能性がある設定を明示的に指定することになります。

## 関連項目:

MAX\_STRING\_SIZEの推奨事項および手順などの詳細は、[Oracle Databaseリファレンス](#)を参照してください

親トピック: [Oracle Databaseのアップグレード後に必要な作業](#)

# パラレル実行サーバーの最大値および最小値の調整

環境に応じて、**PARALLEL\_MIN\_SERVERS**パラメータのデフォルト設定を減らすことができます。

Oracle Database 12cでは、**PARALLEL\_MIN\_SERVERS**のデフォルトは、0からハードウェア・プラットフォームに基づいて提供された値に変更されました。この変更は、パラレル実行に十分な最低限のサポートを提供するために加えられました。新しいデフォルト設定が高すぎる場合は、要件に応じて設定を調整します。**PARALLEL\_MAX\_SERVERS**のデフォルトは変更されていません。以前の環境のデフォルトのまま変更されていない場合、追加の操作を実行する必要はありません。

## 関連項目:

**PARALLEL\_MIN\_SERVERS**の詳細は、[Oracle Databaseリファレンス](#)を参照してください

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のリカバリ・カタログ・アップグレードの概要

RMANクライアントで要求されるバージョンより古いリカバリ・カタログ・スキーマを使用している場合、そのカタログをアップグレードする必要があります。

## 関連項目:

- RMANリカバリ・カタログの管理の詳細は、[『Oracle Databaseバックアップおよびリカバリ・ユーザーズ・ガイド』](#)を参照してください。
- リカバリ・カタログのアップグレードおよびUPGRADE CATALOGコマンドの詳細は、[『Oracle Databaseバックアップおよびリカバリ・ユーザーズ・ガイド』](#)を参照してください

親トピック: [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のタイムゾーン・ファイルのバージョンのアップグレード

データベースのアップグレードを完了した後、アップグレード前情報ツールによってタイムゾーン・ファイルをアップグレードするよう指示された場合は、DBMS\_DST PL/SQLパッケージを使用してタイムゾーン・ファイルをアップグレードします。

Oracle Databaseでは、複数のバージョンのタイムゾーン・ファイルを提供しています。各タイムゾーン・ファイルに関連付けられた2つのタイプのファイルがあり、1つはデータベースに定義されたすべてのタイムゾーンを含む大きいファイルで、1つは最も一般的に使用されるタイムゾーンのみを含む小さいファイルです。大きいバージョンは、timezlg\_version\_number.datという名前です。小さいバージョンは、timezone\_version\_number.datという名前です。ファイルは、Oracle Databaseホーム・ディレクトリ下のoracore/zoneinfoサブディレクトリにあります。

## 関連トピック

- [Oracle Databaseグローバルバージョン・サポート・ガイド](#)
- <https://support.oracle.com/rs?type=doc&id=1585343.1>

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のDBMS\_STATS パッケージで作成された統計表のアップグレード

DBMS\_STATS.CREATE\_STAT\_TABLEプロシージャを使用して統計表を作成した場合、DBMS\_STATS.UPGRADE\_STAT\_TABLEを実行してそれらの表をアップグレードします。

次の例で、greenは統計表の所有者で、STAT\_TABLEは統計表の名前です。

```
EXECUTE DBMS_STATS.UPGRADE_STAT_TABLE('green', 'stat_table');
```

各統計表にこのプロシージャを実行します。

## 関連項目:

DBMS\_STATSパッケージの詳細は、[『Oracle Database PL/SQLパッケージおよびタイプ・リファレンス』](#)を参照してください。

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後の外部認証されたSSLユーザーのアップグレード

Oracle9iリリース2 (9.2)またはOracle Database 10gリリース1 (10.1)からのアップグレード時に、外部認証されたSSLユーザーを使用している場合は、SSL外部ユーザー変換(extusrupgrade)スクリプトを実行してそれらのユーザーをアップグレードする必要があります。

extusrupgradeスクリプトの構文は次のとおりで、ORACLE\_HOMEはOracle Databaseホーム、hostnameはデータベースが実行されているホストの名前、port\_noはリスナーのポート番号、sidはデータベース・インスタンスのシステム識別子、db\_adminはユーザー・アカウントを変更する権限を持つデータベース管理ユーザーです。

```
ORACLE_HOME/rdbms/bin/extusrupgrade --dbconnectstring
hostname:port_no:sid --dbuser db_admin --dbuserpassword
password -a
```

次に例を示します。

```
extusrupgrade --dbconnectstring dlsun88:1521:10gR2 --dbuser system --dbuserpassword manager -a
```

## 注意:



Oracle Database 10g リリース 2 (10.2)以上のリリースからアップグレードしている場合は、extusrupgrade スクリプトを実行する必要はありません。

## 関連項目:

extusrupgradeスクリプトの詳細は、[Oracle Databaseエンタープライズ・ユーザー・セキュリティ管理者ガイド](#)を参照してください

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle XML DBに対するFTPとHTTPのポートおよび HTTP認証の構成

Oracle Database Configuration Assistant (DBCA)では、Oracle Database 12cのOracle XML DBのポートは構成されず、それ以降のリリースのアップグレードではダイジェスト認証が使用されます。

ポートを構成する場合、改善されたセキュリティ機能を利用するために、Oracle XML DB Repositoryへのアクセス用にHTTPの認証も構成することをお勧めします。

Oracle Database 12c以降では、ダイジェスト認証のサポートを提供することによって、データベースのセキュリティが向上しました。ダイジェスト認証は、HTTPプロトコルで一般に使用され、ほとんどのHTTPクライアントでサポートされている業界標準のプロトコルです。ほとんどのHTTPクライアントではこれがサポートされています。ダイジェスト認証により、暗号化(HTTPS)接続が使用されない場合でも、パスワードが常にセキュアな方法で送信されます。ダイジェスト認証をサポートすることによって、組織では、パスワードの漏えいを心配することなくOracle XML DB HTTPを使用するアプリケーションをデプロイできます。Oracle XML DBでのダイジェスト認証のサポートでは、Oracle XML DB HTTPサーバーとMicrosoft Web Folders WebDAVクライアントとの互換性も引き続き維持されます。

新しいリリースのインストールまたはアップグレード後に、次のようにOracle XML DBのFTPおよびHTTPポートを手動で構成する必要があります。

1. DBMS\_XDB\_CONFIG.setHTTPPort(HTTP\_port\_number)を使用して、Oracle XML DBのHTTPポートを設定します。

```
SQL> exec DBMS_XDB_CONFIG.setHTTPPort(port_number);
```

2. DBMS\_XDB\_CONFIG.setFTPPort(FTP\_port\_number)を使用して、Oracle XML DBのFTPポートを設定します。

```
SQL> exec DBMS_XDB_CONFIG.setFTPPort(FTP_port_number);
```

## 注意:



手順内のFTPおよびHTTPで使用するポート番号は、DBMS\_XDB\_CONFIG.getFTPPortおよびDBMS\_XDB\_CONFIG.getHTTPPortをそれぞれ使用することによって問い合わせることができます。

3. 使用されているすべてのポート番号を確認するには、DBMS\_XDB\_CONFIG.usedportを問い合わせます。

## 関連項目:

FTP、HTTP、HTTPSおよびWebDAVプロトコルを使用したOracle XML DB Repositoryのデータへのアクセスの詳細は、[『Oracle XML DB開発者ガイド』](#)を参照してください。

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のOracle Textが提供するナレッジ・ベースのインストール

Oracle Databaseのアップグレードの後、Oracle Textナレッジ・ベースに対するすべてのユーザー拡張を再生成する必要があります。

ユーザー拡張を再生成すると、指定したOracleホームにインストールされているすべてのデータベースに影響を与えます。

新しいOracle Databaseの付属製品の一部である、Oracle Textが提供するナレッジ・ベースは、アップグレードの直後は使用できません。アップグレード前には使用可能であったナレッジ・ベースに依存するOracle Textの機能は、アップグレード後には機能しなくなります。これらの機能を再度使用可能にするには、Oracle Textナレッジ・ベースを新しいOracle Databaseリリースのインストール・メディアからインストールする必要があります。

## 関連項目:

- Oracle Textナレッジ・ベースの詳細は、[Oracle Textアプリケーション開発者ガイド](#)を参照してください
- 付属製品については、[Oracle Databaseインストール・ガイド](#)を参照してください

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後のOracle Application Express構成の更新

Oracle Application Expressのリリースとデータベースのインストール・タイプによっては、Oracle Application Expressがアップグレード手順に影響します。

アップグレードするOracle DatabaseリリースにOracle Application Expressリリース3.2以降が含まれる場合、新しいOracle Databaseリリースへのアップグレード後に追加構成を実行する必要はありません。ただし、Oracle Application Expressがレジストリに存在し、Oracle Application Expressがアップグレードに含まれる場合、**open\_cursors**パラメータを200以上に設定します。

アップグレードするOracle DatabaseがOracle Express Editionデータベースの場合、Oracle Express Edition環境に応じて調整されたOracle Application Expressの以前のリリースが含まれています。アップグレード中に、最新のOracle Application Expressリリースが自動的にインストールされます。新しいOracle Databaseリリースと併用するには、インストール後の一連のステップを実行して、Oracle Application Expressを構成する必要があります。

## 関連項目:

- Oracle Application Expressのインストール後の作業の詳細は、[Oracle Application Expressインストール・ガイド](#)を参照してください
- <http://www.oracle.com/technetwork/developer-tools/apex/overview/index.html>

親トピック: [Oracle Databaseのアップグレード後に必要な作業](#)

# 外部ネットワーク・サービスへのアクセス制御リスト(ACL)の構成

Oracle Database 12c以降のリリースには、UTL\_TCP、UTL\_SMTP、UTL\_MAIL、UTL\_HTTPまたはUTL\_INADDRパッケージに対するファイアウォール・アクセス制御が含まれています。

これらのパッケージを使用するアプリケーションがある場合、Oracle Databaseのアップグレード後に、データベースのネットワーク・アクセス制御リスト(ACL)を構成してから、影響を受けるパッケージを前のリリースと同様に動作させる必要があります。ACLがない場合、エラー「ORA-24247: アクセス制御リスト(ACL)によりネットワーク・アクセスが拒否されました」でアプリケーションが失敗する可能性があります。

## 関連項目:

一部のユーザーはホストAに接続し、別のユーザーはホストBに接続するなど、より複雑な状況については、[Oracle Database セキュリティ・ガイド](#)を参照してください

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# SQLNET.ALLOWED\_LOGON\_VERSIONパラメータの動作の確認

10gより前のリリースのクライアントからのOracle Databaseに対する接続は、ORA-28040: 「一致する認証プロトコルがありません」というエラーによって失敗します。

Oracle Database 18c以降では、SQLNET.ALLOWED\_LOGON\_VERSIONパラメータのデフォルト値が、11 (Oracle Database 12c (12.2))から12 (Oracle Database 18c以降のリリース)に変更されています。このパラメータの使用は非推奨になりました。

SQLNET.ALLOWED\_LOGON\_VERSIONは、現在、SQLNET.ALLOWED\_LOGON\_VERSION\_SERVERおよびSQLNET.ALLOWED\_LOGON\_VERSION\_CLIENTパラメータに置き換えられています。アップグレードしたデータベースでSQLNET.ALLOWED\_LOGON\_VERSION\_SERVERパラメータを明示的に設定していない場合、10gより前のリリースのクライアントからの接続は、ORA-28040: 「一致する認証プロトコルがありません」というエラーによって失敗します。セキュリティ強化のため、データベース・ユーザーのパスワード検証機能を確認し、SQLNET.ALLOWED\_LOGON\_VERSION\_SERVERおよびSQLNET.ALLOWED\_LOGON\_VERSION\_CLIENTパラメータを設定して正しいパスワード検証機能を使用するようにデータベースを構成します。

既存のデータベースにパスワードで保護されたロール(セキュア・ロール)があり、デフォルトのSQLNET.ALLOWED\_LOGON\_VERSION\_SERVER設定である12を使用してOracle Database 18c以降のリリースにアップグレードする場合、そのセキュア・ロールに含まれるのはリリース10gの検証機能のみであるため、アップグレード後もセキュア・ロールが使用可能な状態になるように、管理者は各セキュア・ロールのパスワードをリセットする必要があります。

## 関連項目:

- パスワードのセキュリティの脅威から守る方法の詳細は、[『Oracle Databaseセキュリティ・ガイド』](#)を参照してください。
- ユーザーのパスワード・バージョンの設定の詳細は、[『Oracle Databaseセキュリティ・ガイド』](#)[『Oracle Databaseセキュリティ・ガイド』](#)を参照してください。

**親トピック:** [Oracle Databaseのアップグレード後に必要な作業](#)

# Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス

Oracle Databaseを更新する場合、これらの適切なプラクティス・ガイドラインを完了することをお勧めします。これらのプラクティスは、手動とDBUAの両方のアップグレードにお勧めです。

- [データベースのバックアップ](#)  
本番データベースの全体バックアップを実行します。
- [postupgrade\\_fixups.sqlスクリプトを実行する非CDBのシナリオ](#)  
非CDBデータベースに対してpostupgrade\_fixups.sqlスクリプトを使用する方法を理解するには、この手順を確認します。
- [アップグレード後のディクショナリ統計の収集](#)  
アップグレードを完了した後に高いパフォーマンスを保証するためには、この手順を使用してディクショナリ統計を収集します。
- [DBMS\\_STATSを使用した固定オブジェクトの統計の再収集](#)  
アップグレードの後に、または他のデータベース構成の変更後に、Oracle Databaseに対して代表的なワークロードを実行してから固定オブジェクトの統計を再収集することをお勧めします。
- [パスワードのリセットによる大/小文字区別の強制](#)  
アップグレードしたデータベースで、デフォルト・ユーザー・アカウントおよびユーザー・アカウントの大/小文字区別のあるパスワードを使用してセキュリティを強化します。
- [新機能の適宜追加](#)  
データベース・アップグレード計画の一環として新機能を確認します。
- [必要な新しい管理手順の作成](#)  
スクリプトおよびプロシージャの再確認を計画し、必要に応じて変更します。
- [表領域アラートのしきい値の設定](#)  
アップグレード後、Oracle Databaseのアップグレードされた表領域アラートのしきい値はNULLに設定され、アラートは無効化されます。
- [ロールバック・セグメントから自動UNDOモードへの移行](#)  
データベース・リリースがOracle Database 11gより前の場合は、ロールバック・セグメント(手動UNDO管理)を使用してアップグレードしているデータベースを自動UNDO管理に移行する必要があります。
- [Oracle Data Guard Brokerの構成](#)  
**InitialConnectIdentifier**は**DGConnectIdentifier**に置き換えられたため、Oracle Database 10gからのアップグレードに影響します。
- [LONGデータ型からLOBデータ型への表の移行](#)  
ALTER TABLE文を使用して、LONGデータ型の列をCLOBに、LONG RAWデータ型の列をBLOBに変更できます。
- [再構築対象のOracle Text索引の識別](#)  
新しいOracle Databaseリリースへのアップグレード後にトークン表の再構築が推奨されるOracle Text索引の識別に役立つスクリプトを実行できます。
- [アップグレードした本番Oracle Databaseのテストの概要](#)  
アプリケーションが期待どおり動作することを確認するため、テスト・データベースに実行したテストを本番データベースに対して繰り返します。

**親トピック:** [Oracle Databaseのアップグレード後の作業](#)

# データベースのバックアップ

本番データベースの全体バックアップを実行します。

このステップは必須ではありませんが、本番データベースをバックアップすることを強くお勧めします。

## 関連項目:

RMANを使用したデータベースのバックアップの詳細は、[『Oracle Databaseバックアップおよびリカバリ・ユーザズ・ガイド』](#)を参照してください。

親トピック: [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# postupgrade\_fixups.sqlスクリプトを実行する非CDBのシナリオ

非CDBデータベースに対してpostupgrade\_fixups.sqlスクリプトを使用する方法を理解するには、この手順を確認します。

アップグレード後修正スクリプトは、アップグレード前情報ツール(preupgrade.jar)を実行したときに生成されます。アップグレード後スクリプトは、アップグレード完了後の任意の時点で実行します。非CDBデータベースの場合、アップグレード後修正スクリプトでは、一般的な警告、エラーおよび情報としての推奨事項が提供されます。

catcon.plユーティリティまたはSQL\*Plusを使用して、このスクリプトを実行できます。

アップグレード後SQLスクリプトとログ・ファイルの場所は、出力フォルダの設定またはOracleベース環境変数の定義によって異なります。アップグレード後修正スクリプトは、アップグレード前修正スクリプトと同じディレクトリ・パスに配置されます。

アップグレード前情報ツールでdirオプションを使用して出力ディレクトリを指定した場合、出力ログおよびファイルはファイル・パス/cfgtoollogs/dbunique\_name/preupgradeの該当ディレクトリの下に配置されます。ここで、dbunique\_nameは、ソースOracle Databaseの名前です。アップグレード前情報ツールの実行時に出力ディレクトリを指定しなかった場合は、次のデフォルトの場所のいずれかに出力されます。

- DIRで出力ディレクトリを指定してなくても、Oracleベース環境変数を設定した場合は、生成されるスクリプトとログ・ファイルは次のファイル・パスに作成されます。

Oracle-base/cfgtoollogs/dbunique\_name/preupgrade

- 出力ディレクトリを指定せず、Oracleベース環境変数も定義していない場合は、生成されるスクリプトとログ・ファイルは次のファイル・パスに作成されます。

Oracle-home/cfgtoollogs/dbunique\_name/preupgrade

アップグレード前情報ツールによって作成されるアップグレード後修正スクリプトは、ソース・データベースが非CDBデータベースとCDBデータベースのどちらであるかによって異なります。

- 非CDB: postupgrade\_fixups.sql

## 例10-4 非CDB Oracle Databaseのアップグレード後修正結果のスピーニングの例

出力を確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。ただし、adminディレクトリにはスプールしないでください。

```
SQL> SPPOOL postupgrade.log
SQL> @postupgrade_fixups.sql
SQL> SPPOOL OFF
```

スクリプト結果のログ・ファイルへのスピーニングをオフにします。

```
SPPOOL OFF
```

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# アップグレード後のディクショナリ統計の収集

アップグレードを完了した後に高いパフォーマンスを保証するためには、次の手順を使用してディクショナリ統計を収集します。

データ・ディクショナリ表はアップグレード中に変更され、作成されるため、データベースのアップグレードの前後でディクショナリ統計を収集することをお勧めします。Oracle Database 12cリリース2 (12.2)以上では、アップグレード後にデータベースを通常モードで起動したときに、統計を手動で収集するようになりました。

- 非CDBのOracle Databaseの場合: DBMS\_STATS.GATHER\_DICTIONARY\_STATSプロシージャを使用して、統計を収集することをお勧めします。たとえば、次のSQL文を入力します。

```
SQL> EXEC DBMS_STATS.GATHER_DICTIONARY_STATS;
```

- CDBの場合: catconを使用して、マルチテナント・アーキテクチャ全体のデータ・ディクショナリ統計を収集することをお勧めします

コンテナ・データベースのすべてのPDBのディクショナリ統計を収集するには、次の構文を使用します。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -l /tmp -b gatherstats -- --x"exec dbms_stats.gather_dictionary_stats"
```

特定のPDBに関するディクショナリ統計を収集するには、次のような構文を使用します。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -l /tmp -c 'SALES1' -b gatherstats -- --x"exec dbms_stats.gather_dictionary_stats"
```

前述の例では、SALES1というデータベースを指定することにより、実行するコマンドのPDB包含リストを-c SALES1オプションで指定しています。オプション-b gatherstats では、ログの基底名を指定します。オプション--xは、実行するSQLコマンドを指定します。SQLコマンドそのものは、引用符で囲みます。

## 関連トピック

- [Oracle Database PL/SQLパッケージ・プロシージャおよびタイプ・リファレンス](#)

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# DBMS\_STATSを使用した固定オブジェクトの統計の再収集

アップグレードの後に、または他のデータベース構成の変更後に、Oracle Databaseに対して代表的なワークロードを実行してから固定オブジェクトの統計を再収集することを強くお勧めします。

固定オブジェクトは、X\$表とその索引です。V\$パフォーマンス・ビューは、X\$表を通じて定義されます。固定オブジェクトの統計の収集は、データベース・パフォーマンスにとって有益で(それらの統計はオプティマイザが適切な実行計画を生成する際に役立つため)、データベース・パフォーマンスが向上する可能性があります。代表的な統計を取得しないと、実行計画が最適ではなくなる可能性があり、深刻なパフォーマンス問題が発生する場合があります。

固定オブジェクトの統計を収集するには、DBMS\_STATS.GATHER\_FIXED\_OBJECTS\_STATS PL/SQLプロシージャを使用します。DBMS\_STATS.GATHER\_FIXED\_OBJECTS\_STATSでは、INIT.ORAまたはSPFILEからすべての非表示またはアンダースコアのパラメータおよびイベントを削除するための推奨事項も表示されます。

X\$表は一時的な性質があるため、システムで代表的なワークロードが発生したときに固定オブジェクトの統計を収集する必要があります。ピーク負荷時に固定オブジェクトの統計を収集できない場合、システムが実行時の状態で、最も重要なタイプの固定オブジェクト表が移入された後に収集することをお勧めします。

固定オブジェクトの統計を収集するには、次のPL/SQLプロシージャを実行します。

```
SQL> execute dbms_stats.gather_fixed_objects_stats;
```

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# パスワードのリセットによる大/小文字区別の強制

アップグレードしたデータベースで、デフォルト・ユーザー・アカウントおよびユーザー・アカウントの大/小文字区別のあるパスワードを使用してセキュリティを強化します。

セキュリティを強化するために、パスワードで大/小文字の区別を有効にすることをお勧めします。大/小文字を区別すると、ユーザーは、正しいパスワード文字列を入力し、さらにその文字列の各文字の大/小文字も正しく区別する必要があるため、パスワードのセキュリティを強化できます。たとえば、パスワードがhPP5620qrの場合、hpp5620QRまたはhPp5620Qrと入力すると失敗します。

データベースを保護するため、パスワードはセキュアな方法で作成します。データベースのデフォルト・パスワードを使用している場合、それらのパスワードは変更してください。デフォルトでは、パスワードを変更すると、大/小文字区別が強制されます。事前定義されたユーザー・アカウントのパスワードを含め、すべてのパスワードは、Oracle推奨のパスワード要件を満たしている必要があります。

アップグレード後に新しく作成されたデータベースでは、追加の作業や追加の管理要件はありません。

パスワード変更に関する既存のデータベース要件およびガイドライン

- Oracle Database 12cリリース1 (12.1)以上のデフォルトのセキュリティ設定が適用されている場合、パスワードは8文字以上にする必要があり、welcomeやoracleなどのパスワードは使用できません。
- IGNORECASEパラメータは非推奨です。このパラメータは使用しないでください。
- 既存のデータベースでパスワードの大/小文字区別を利用するには、データベースのアップグレード作業中に既存のユーザーのパスワードをリセットする必要があります。既存のデータベース・ユーザーごとに、ALTER USER文を使用してパスワードをリセットします。
- DBA\_USERSのPASSWORD\_VERSIONS列を問い合わせ、10Gのパスワード・バージョンのみを持ち、11Gや12Cのパスワード・バージョンを持たないアカウントのUSERNAMEを見つけます。10Gのパスワード・バージョンのみを持つすべてのアカウントのパスワードをリセットします。
- [10Gパスワード・バージョンを使用するユーザー・パスワードの確認と再設定](#)  
よりセキュアなパスワード認証を行うためには、10Gパスワード・バージョンを使用するユーザー・アカウントのパスワードを確認して再設定し、より新しい、よりセキュアなパスワード・バージョンを使用するようにします。

## 関連項目:

- パスワードの大/小文字区別の詳細は、[『Oracle Databaseセキュリティ・ガイド』](#)を参照してください
- パスワードの強度の詳細は、[『Oracle Databaseセキュリティ・ガイド』](#)を参照してください

親トピック: [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# 10Gパスワード・バージョンを使用するユーザーのパスワードの確認と再設定

セキュリティを向上するため、10Gバージョンのパスワードを使用しているユーザー・アカウントを確認しパスワードをリセットして、より安全なバージョンのパスワードが今後使用されるようにします。

現行ユーザーのすべてのパスワード・バージョンの確認

DBA\_USERSデータ・ディクショナリ・ビューを問い合せて、ユーザー・アカウント用に構成されたすべてのパスワード・バージョンのリストを確認できます。

次に例を示します。

```
SELECT USERNAME, PASSWORD_VERSIONS FROM DBA_USERS;
```

| USERNAME | PASSWORD_VERSIONS |
|----------|-------------------|
| JONES    | 10G 11G 12C       |
| ADAMS    | 10G 11G           |
| CLARK    | 10G 11G           |
| PRESTON  | 11G               |
| BLAKE    | 10G               |

PASSWORD\_VERSIONS列は、アカウントに存在するパスワード・バージョンのリストを示しています。10Gは以前の大/小文字を区別しないOracleパスワード・バージョン、11GはSHA-1ベースのパスワード・バージョン、12CはSHA-2ベースのSHA-512パスワード・バージョンを表します。

- ユーザー-jones: このユーザーのパスワードは、SQLNET.ALLOWED\_LOGON\_VERSION\_SERVERパラメータ設定が8であった場合にOracle Database 12cリリース12.1では再設定されていました。これにより、3つのすべてのパスワード・バージョンを作成できます。
- ユーザー-adamsおよびclark: これらのアカウントのパスワードは最初にOracle Database 10gで作成され、Oracle Database 11gで再設定されました。Oracle Database 11gソフトウェアは、その時点でSQLNET.ALLOWED\_LOGON\_VERSIONのデフォルト設定8を使用していました。大/小文字の区別がデフォルトで有効になっているため、これらのパスワードは、prestonのパスワードと同様に大/小文字が区別されます。
- ユーザー-preston: このアカウントは、排他モード(SQLNET.ALLOWED\_LOGON\_VERSION = 12)で実行されていたOracle Database 11gデータベースからインポートされました。
- ユーザー-blake: このアカウントは、Oracle Database 10gパスワード・バージョンをまだ使用しています。この段階では、ユーザー-blakeはログインできません。

## 10Gパスワード・バージョンを使用するユーザーのパスワードの再設定

セキュリティを強化するには、すべてのユーザーのアカウントから10Gパスワード・バージョンを削除します。次の手順では、10Gパスワード・バージョンを使用しているユーザーのパスワードを再設定するために、クライアントのログインの許可に必要な機能レベルを制御するSQLNET.ALLOWED\_LOGON\_VERSION\_SERVER設定を一時的に緩和する必要があります。設定を緩和することで、これらのユーザーがログインしてパスワードを変更できるようになるため、10Gパスワード・バージョンに加えて、新しいパスワード・バージョンを生成できます。後で、排他モードを使用するようにデータベースを設定し、クライアントがO5L\_NP機能を使用できることを確認できます。その後、ユーザーはパスワードを再設定して、パスワード・バージョンに10Gを含めずに、より安全な11Gと12Cのパスワード・バージョンのみを含めるようにすることができます。

1. DBA\_USERSビューを問い合せて、10Gパスワード・バージョンのみを使用しているユーザーを特定します。

```
SELECT USERNAME FROM DBA_USERS
WHERE (PASSWORD_VERSIONS = '10G '
OR PASSWORD_VERSIONS = '10G HTTP ')
AND USERNAME <> 'ANONYMOUS' ;
```

2. データベースを排他モードで実行しないように、次のように構成します。
  - a. sqlnet.oraファイルのSQLNET.ALLOWED\_LOGON\_VERSION\_SERVER設定を編集して、デフォルトより緩やかな設定にします。次に例を示します。

```
SQLNET.ALLOWED_LOGON_VERSION_SERVER=11
```

- b. データベースを再起動します。
3. DBA\_USERSビューに問い合わせて、10Gパスワード・バージョンのみを使用するユーザーを期限切れにします。

10Gパスワード・バージョンのみを使用し、11Gまたは12Cパスワード・バージョンのいずれかまたは両方を使用していないユーザーを期限切れにする必要があります。

次に例を示します。

```
ALTER USER username PASSWORD EXPIRE;
```

4. パスワードを期限切れにしたユーザーにログインするよう依頼します。

ユーザーがログインすると、パスワードを変更するよう求められます。データベースは、10Gパスワード・バージョンに加えて、アカウントで欠けている11Gおよび12Cパスワード・バージョンを生成します。データベースは許可モードで実行されているため、10Gパスワード・バージョンはそのまま存在します。

5. ユーザーが接続しているクライアント・ソフトウェアに05L\_NP機能があることを確認します。

Oracle Databaseリリース11.2.0.3以上のすべてのクライアントに05L\_NP機能があります。以前のOracle Databaseクライアントがある場合は、CPUOct2012パッチをインストールする必要があります。

6. すべてのクライアントに05L\_NP機能があれば、次のようにサーバーのセキュリティを排他モードに設定しなおします。

- a. インスタンス初期化ファイルからSEC\_CASE\_SENSITIVE\_LOGONパラメータ設定を削除するか、SEC\_CASE\_SENSITIVE\_LOGONをTRUEに設定します。

```
SEC_CASE_SENSITIVE_LOGON = TRUE
```

- b. サーバーsqlnet.oraファイルからSQLNET.ALLOWED\_LOGON\_VERSION\_SERVERパラメータを削除するか、または、サーバーsqlnet.oraファイルのSQLNET.ALLOWED\_LOGON\_VERSION\_SERVERの値を元の12に設定して、排他モードに設定します。

```
SQLNET.ALLOWED_LOGON_VERSION_SERVER = 12
```

- c. データベースを再起動します。

7. まだ10Gパスワード・バージョンを使用しているアカウントを特定します。

```
SELECT USERNAME FROM DBA_USERS
WHERE PASSWORD_VERSIONS LIKE '%10G%'
AND USERNAME <> 'ANONYMOUS' ;
```

8. まだ10Gパスワード・バージョンを使用しているアカウントを期限切れにします。

```
ALTER USER username PASSWORD EXPIRE;
```

9. これらのユーザーに自分のアカウントにログインするよう依頼します。

ユーザーがログインすると、パスワードを再設定するよう求められます。次に、データベースは、アカウントに対して11Gおよ

び12Gパスワード・バージョンのみを生成します。データベースは排他モードで実行されているため、10Gパスワード・バージョンは生成されません。

10. 次の問合せを再実行します。

```
SELECT USERNAME FROM DBA_USERS
WHERE PASSWORD_VERSIONS LIKE '%10G%'
AND USERNAME <> 'ANONYMOUS';
```

この問合せに何も応答がなければ、10Gパスワード・バージョンを使用しているアカウントはありません。これで、データベースは以前のバージョンよりも安全なモードで稼働しています。

**親トピック:** [パスワードのリセットによる大/小文字区別の強制](#)

# 新機能の適宜追加

データベース・アップグレード計画の一環として新機能を確認してください。

『Oracle Database新機能ガイド』では、新しいOracle Databaseリリースで使用可能な多くの新機能について説明されています。これらのどの新機能がデータベースおよびアプリケーションに有効かを判断してください。その後、これらの機能を使用する計画を立てることができます。

新しいOracle Databaseソフトウェアを使用するためにすぐに変更する必要はありません。データベースおよびアプリケーションに、新機能の拡張機能を徐々に取り入れることもできます。

## 関連項目:

[『Oracle Database新機能ガイド』](#)

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# 必要な新しい管理手順の作成

スクリプトおよびプロセスの再確認を計画し、必要に応じて変更します。

Oracle Databaseの新しいリリースの機能をよく理解したうえで、データベース管理用のスクリプトおよびプロセスを再確認し、変更が必要かどうかを判断します。

それぞれのアプリケーションに必要な変更を、データベースにも行う必要があります。たとえば、データベースで整合性制約を使用可能にした場合、アプリケーションでのデータ・チェックの一部を削除できます。

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# 表領域アラートのしきい値の設定

アップグレード後、Oracle Databaseのアップグレードされた表領域アラートのしきい値はNULLに設定され、アラートは無効化されます。

データベース内で監視対象となる表領域を特定し、それらの表領域に適切なしきい値を設定する必要があります。

Oracle Database18c以降のリリースで新しく作成されたOracle Databaseインストールでは、次の値がデフォルトとして使用されます。

- 警告(表領域の使用率が85%の場合)
- クリティカル(表領域の使用率が97%の場合)

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# ロールバック・セグメントから自動UNDOモードへの移行

データベース・リリースがOracle Database 11gより前の場合は、ロールバック・セグメント(手動UNDO管理)を使用してアップグレードしたデータベースを、自動UNDO管理に移行する必要があります。

自動UNDO管理は、デフォルトのUNDO領域管理モードです。システムで使用するUNDO領域管理モードは、UNDO\_MANAGEMENT初期化パラメータで指定します。

- **UNDO\_MANAGEMENT**がAUTOに設定されている場合(または**UNDO\_MANAGEMENT**が設定されていない場合)、データベース・インスタンスは自動UNDO管理モードで起動します。

**UNDO\_MANAGEMENT**初期化パラメータがNULLの場合のデフォルトは、Oracle Database 11gリリース1(11.1)以降では自動UNDO管理モードです。以前のリリースでは、手動UNDO管理モードがデフォルトです。以前のリリースをアップグレードするときには注意してください。

- **UNDO\_MANAGEMENT**が **MANUAL**に設定されている場合、UNDO領域はロールバック・セグメントとして外部に割り当てられます。

1. **UNDO\_MANAGEMENT**パラメータを **UNDO\_MANAGEMENT=MANUAL**に設定します。
2. インスタンスを再起動して標準的なビジネス・サイクルを一通り実行し、代表的なワークロードを取得します。ワークロードを評価し、自動UNDO管理で必要なUNDO表領域のサイズを計算します。
3. 標準的なビジネス・サイクルを完了したら、次のファンクションを実行してUNDO表領域のサイズを収集すると、UNDO表領域のサイズ変更役に立ちます。このファンクションを実行するにはSYSDBA権限が必要です。

```
DECLARE
 utbsiz_in_MB NUMBER;
BEGIN
 utbsiz_in_MB := DBMS_UNDO_ADV.RBU_MIGRATION;
end;
/
```

このファンクションではPL/SQLプロシージャが実行され、システム構成およびシステムのロールバック・セグメントの使用状況を基にして、新しいUNDO表領域のサイズを求める方法に関する情報が提供されます。このファンクションはサイズ変更に関する情報を直接戻します。

4. 必要なサイズのUNDO表領域を作成し、**UNDO\_MANAGEMENT=AUTO**に設定するかパラメータを削除して、自動UNDO管理を有効にします。
5. Oracle RAC構成の場合は、すべてのインスタンスでこれらのステップを繰り返します。

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# Oracle Data Guard Brokerの構成

**InitialConnectIdentifier**は**DGConnectIdentifier**に置き換えられたため、Oracle Database 10gからのアップグレードに影響します。

DGConnectIdentifierの値は、常時、すべてのData Guardネットワーク・トラフィック用に使用されます。Oracle Databaseリリース10gの構成をアップグレードする場合は、最初にOracle Database 11gにアップグレードする必要があり、InitialConnectIdentifierの値は、そのデータベースのDGConnectIdentifierの新しい値として保持されます。Oracle Real Application Clusters (Oracle RAC)データベースをアップグレードする場合、データベース管理者は、InitialConnectIdentifierプロパティの値がすべてのインスタンスに適用されることを確認する必要があります。

移行では、スタンバイ・データベースに対してのみこのステップを実行する必要があります。

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# LONGデータ型からLOBデータ型への表の移行

ALTER TABLE文を使用して、LONGデータ型の列をCLOBに、LONG RAWデータ型の列をBLOBに変更できます。

LOBデータ型(BFILE、BLOB、CLOBおよびNCLOB)には、LONGデータ型よりも多くのメリットがあります。

次の例では、long\_tab表のlong\_colというLONG列が、CLOBデータ型に変更されます。

```
SQL> ALTER TABLE Long_tab MODIFY (long_col CLOB);
```

この方法でLONG列をLOBに変換した後も、表に設定されている既存の制約およびトリガーはすべて使用できます。ただし、表のすべての列で、ドメイン索引およびファンクション索引を含むすべての索引が使用不可となるため、ALTER INDEX ... REBUILD文を使用してすべての索引を再構築する必要があります。また、LONG列上のドメイン索引は、LONG列をLOBに変更する前に削除する必要があります。

## 関連項目:

LOBデータを使用するためのアプリケーションの変更の詳細は、[『Oracle Database SecureFilesおよびラージ・オブジェクト開発者ガイド』](#)を参照してください。

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# 再構築対象のOracle Text索引の識別

新しいOracle Databaseリリースへのアップグレード後にトークン表の再構築が推奨されるOracle Text索引の識別に役立つスクリプトを実行できます。

Oracle Database 12cリリース1 (12.2.0.1)をOracle Database 18c以降のリリースにアップグレードすると、Oracle Textトークン表(\$I、\$Pなど)が64バイトから255バイトに拡張されます。しかし、索引で使用されている既存のトークン表のサイズ範囲が小さいと、拡張後のトークン列範囲をOracle Text索引で利用できなくなります。255バイトのサイズ範囲を使用するように索引を再構築する必要があります。Oracleは、再構築が推奨される索引の識別に役立つスクリプトを提供しています。

My Oracle Supportからスクリプトを取得してください。

<https://support.oracle.com/rs?type=doc&id=2287094.1>

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)

# アップグレードした本番Oracle Databaseのテストの概要

アプリケーションが期待どおり動作することを確認するため、テスト・データベースに実行したテストを本番データベースに対して繰り返します。

テスト・データベースをOracle Databaseの新しいリリースにアップグレードしてテストした場合、Oracle Databaseの新しいリリースにアップグレードした本番データベースでも同じテストを繰り返すことができます。結果を比較し、相違点を記録します。必要に応じて、アップグレードのテストを繰り返します。

Oracle Databaseの新しいリリースでアプリケーションが適切に動作することを確認するには、新しくアップグレードされた本番データベースを既存のアプリケーションでテストします。また、使用可能なOracle Database機能を追加してテストし、拡張機能をテストすることもできます。ただし、アプリケーションがアップグレードの前と同様に動作するかどうかを最初に確認してください。

**親トピック:** [Oracle Databaseのアップグレード後の推奨作業およびベスト・プラクティス](#)